End-to-End QoS Network Design

Quality of Service for Rich-Media & Cloud Networks

Second Edition

Tim Szigeti
Christina Hattingh
Robert Barton
Kenneth R. Briley, Jr.

ciscopress.com

End-to-End QoS Network Design

Second Edition

Tim Szigeti, CCIE No. 9794
Robert Barton, CCIE No. 6660
Christina Hattingh
Kenneth Briley, Jr., CCIE No. 9754

Cisco Press

800 East 96th Street

Indianapolis, IN 46240

End-to-End QoS Network Design
Quality of Service for Rich-Media & Cloud Networks
Second Edition

Tim Szigeti, CCIE No. 9794
Robert Barton, CCIE No. 6660
Christina Hattingh
Kenneth Briley Jr., CCIE No. 9754

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

Printed in the United States of America

First Printing November 2013

Library of Congress Control Number: 2013950000

ISBN-13: 978-1-58714-369-4

ISBN-10: 1-58714-369-0

Warning and Disclaimer

Trademark Acknowledgments

Corporate and Government Sales

The publisher offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales, which may include electronic versions and/or custom covers and content particular to your business, training goals, marketing focus, and branding interests . For more information, please contact: **U.S. Corporate and Government Sales** 1-800-382-3419 corpsales@pearsontechgroup.com

For sales outside of 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: Paul Boger
Business Operation Manager, Cisco Press: Jan Cornelssen
Associate Publisher: Dave Dusthimer
Executive Editor: Brett Bartow
Senior Development Editor: Christopher Cleveland
Managing Editor: Sandra Schroeder
Copy Editor: Keith Cline
Project Editor: Seth Kerney
Technical Editors: John Johnston, Roland Saville
Editorial Assistant: Vanessa Evans
Proofreader: Jess DeGabriele
Cover Designer: Mark Shirar
Indexer: Christine Karpeles
Composition: Jake McFarland

About the Authors

Tim Szigeti, CCIE No. 9794, is a senior technical leader in the Systems Design Unit at Cisco Systems, where his role is to design network architectures for enterprise mobility solutions. He has specialized in quality of service technologies for the past 15 years, during which time he has authored many technical papers, design guides, and two Cisco Press books: *End-to-End QoS Network Design* (version 1) and *Cisco TelePresence Fundamentals*.

Robert Barton, CCIE No. 6660, is located in Vancouver, where he lives with his wife and two children. He graduated from the University of British Columbia with a degree in engineering physics, and is a registered professional engineer. Rob holds dual CCIEs, in Routing and Switching and Security, and was also the first CCDE in Canada. Rob joined Cisco from ArrowPoint Communications, where he worked as a data center specialist supporting many of the largest corporations in Canada. In the time since ArrowPoint was acquired by Cisco, Rob has worked as a public sector systems engineer, primarily focused on wireless and security architectures. Currently, Rob is working on SmartGrid network technologies, including smart meter and intelligent substation design.

Christina Hattingh spent 13 years as a senior member of the technical staff in Unified Communications (UC) in the Enterprise Networking Routing Group (formerly Services Routing Technology Group or SRTG) at Cisco Systems. The SRTG products, including the Cisco 2900/3900 and 2800/3800 series ISR platforms and their predecessors, were the first Cisco platforms to converge voice, data, and video traffic and services on IP networks by offering TDM gateway interfaces, WAN interfaces, call control, and QoS features. The ISR series of routers often live at smaller remote offices and therefore at the edge of the WAN, where the need for QoS services is most sensitive. In this role, Christina spoke at Cisco Live conferences, trained Cisco sales staff and Cisco resale partners on router-based UC technologies, authored several Cisco Press books, and advised customers on UC network deployment and design, including QoS designs and helping them through the TDM to SIP trunk industry transition.

Kenneth Briley, Jr., CCIE No. 9754 is a technical lead in the Network Operating Systems Technology Group at Cisco Systems. For over 10 years, he has specialized in quality of service design and implementation in customer environments, alignment of QoS features and functions, and the marketing of new products that leverage QoS technologies. During this time, he has written several deployment guides and whitepapers, presented at Cisco Live, and most recently has focused on the convergence of wired and wireless quality of service.

About the Technical Reviewers

John Johnston, previously CCIE No. 5232, is a technical marketing engineer for Cisco Systems. His focus is on mobile security technology and design validation. John has more than 19 years of experience in IP internetworking, including the design and implementation of enterprise networks. Before joining Cisco Systems, John provided network design support for Fortune 500 companies. He holds a BSEE from the UNC-Charlotte.

Roland Saville is a Technical Leader for the Systems Development Unit (SDU) at Cisco, focused on developing best-practice design guides for enterprise network deployments. He has more than 18 years of experience at Cisco as a Systems Engineer, Product Manager, Consulting Systems Engineer, Technical Marketing Engineer, and Technical Leader. During that time, he has focused on a wide range of technology areas, including the integration of voice and video onto network infrastructures, network security, wireless LAN networking, RFID, energy management, Cisco TelePresence, and BYOD. He has also spent time focusing on the retail market segment. Prior to Cisco, he spent eight years as a Communications Analyst for Chevron Corporation. Roland holds a bachelor's of science degree in electrical engineering from the University of Idaho and an MBA degree from Santa Clara University. He co-authored the book *Cisco TelePresence Fundamentals*, is a member of the IEEE, and has 12 U.S. patents.

Dedications

Tim Szigeti:

I find myself in a dedication dilemma.

On the one hand, I already went to great lengths to explain why *not* dedicating the first edition of this book to my wife would have been a fatal mistake. Since then, I've gone on to dedicate my second book to my son, and now I have a beautiful daughter who deserves a dedication too (and whose arrival, incidentally, actually delayed the release of this edition by a couple of months).

So, the question becomes: Are dedications—as their definition implies—*exclusive* to a given book? Or can these be edition-specific? Or perhaps the more important question is: Do I really think it wise to get into a debate over semantics with my wife, who has a double-major in both English and philosophy?

So I'll play the political game and try to weakly rationalize a compromise: The first edition of this book was dedicated to Lella. The second will be to Lella 2.0, or as she's more commonly known, Isla.

Besides, I've already witnessed how much my daughter values my books. For example, over the past few months, she's had two copies of my previous book under her crib, slightly elevating one end to alleviate nighttime gas. Since she wakes up happy and smiling every morning, I'll infer from this her appreciation of the practical benefits of my work. Furthermore, she's always ready to gnaw and drool on my books until they're nice and soggy, and since pure happiness is expressed during the process, I'll attribute this to her esteem of the quality of the authorship.

And so, to my beautiful little girl, I wish to dedicate to you this work. I really don't know how I ever managed to finish it, seeing as how little you let me sleep over the past few months! I know you'll probably never read it, but that's not the point. I just want you to know you were always on my mind and made working on it virtually impossible! And I'm so very happy it's all done with now, so that I can spend more time playing with you and letting you continue wrapping me tightly around your tiny little finger!

Rob Barton:

This book is dedicated to my two wonderful boys, Adrian and Matthew. It's not that I expect you to actually pick up the book and try to become QoS experts, or that I am even trying to encourage you toward a career in network design or engineering, although these are noble pursuits. Rather, the lesson that writing this book has reminded me of is that you only grow as a person when you recognize the space you are in and make the decision to do something new. Oftentimes, we don't know what direction our efforts will take us in, but when you make the mindful choice to do something that is difficult, challenging, and can cause you more than a little pain along the way, you grow. No muscle ever grew without the fibers being damaged through exercise, and so is it too with all aspects of life. My hope is that this book will inspire you throughout your life to look for opportunities for growth—be it artistic, mental, professional, physical, or spiritual. This book is for you.

Christina Hattingh:

To Robert Verkroost and my parents for their unfailing encouragement and support.

Kenneth Briley, Jr.:

As this is my first book, I'd like to heed Tim's advice and dedicate it to my beautiful wife Mirah for fear of the aforementioned transgression. To Mirah, who incidentally read and approved this dedication, and her countless hours devoted to resolving numerous grammatical errors and listening to me drone on about how incredibly interesting QoS is. To our growing family; Lukas, Erik and Max: please don't grow up too fast, and remember that all things are possible.

Acknowledgments

Tim Szigeti:

First, I'd like to thank all the readers of the first edition who made it the success that it has become. There aren't many technology books that are still being steadily purchased nearly 10 years after their release. And a special thanks to the reviewers who have posted comments for it; I cannot express the pride and appreciation I felt when I saw five-star ratings for our book on Amazon. Thank you!

Thanks to my director, Neil Anderson, for long recognizing the critical role of QoS across all our networking systems and solutions and ensuring that it was always properly addressed. Thanks, too, to Greg Edwards in helping to define and articulate various end-to-end QoS strategies.

Thank you Fred Baker for your guidance and direction in both defining and interpreting various QoS standards. Thanks, too, to James Polk for continuing to push the envelope in the IETF to define what tomorrow's QoS models are going to look like.

I'd like to thank the Vancouver Cisco office lab administrator, Mojan Mobasser, for all her diligence in sourcing and arranging access to equipment. Similar thanks are extended to Dawid Brink for letting me use his Nexus boxes—no questions asked!

Farther east, I'd also like to extend thanks to the Toronto Bell Canada team for allowing me extended access to their ASR and CRS labs. Similar thanks, but in the opposite geographic direction, go out to Lim Fung in our Singapore office for providing me access to his labs also.

I'd like to extend sincere thanks to Tim Stevenson for his amazing technical expertise, particularly relating to data center platforms. You really helped demystify a lot of hardware architectural questions I was grappling with. Thanks, Tim!

Also I'd like to thank Lukas Krattiger in Switzerland for hours of research, testing, and correspondence to ensure that we properly wrapped our arms around Nexus 7000 QoS. Thanks for all your insight, patience, and hard work, Lukas!

Additionally, I'd like to thank Lucien Avramov for sharing his work on data center QoS and allowing me to borrow from it. Thank you too, Mike Herbert—wherever you are—for getting the ball rolling in this area.

I'd like to thank also the Cisco product teams that listened to the feedback we offered as a result of producing this book so as to continue to improve our QoS feature sets. This includes Albert Mitchell and the Catalyst 2K/3K team for implementing our latest designs into a new version of AutoQoS. Thanks also to Sarath Subrahmanya and Ramachandra Murthy in India for taking to heart our suggestions on WLC QoS feature enhancements. Kudos also go out to Oz Ben-Rephael and team in Israel for continuing to develop NBAR2 application signatures, including for our own Jabber application.

Thanks to the Cisco Press team. Brett Bartow: Thanks for taking on this project and allowing us to thoroughly update and expand on our original work in a comprehensive manner. We appreciate that you didn't blow a gasket when we exceeded our targeted

page count again, again, and again—to a final tune of target +50%! Thanks also for delaying this publication by a couple of months, letting me focus on my family as my daughter was born.

Thank you Chris Cleveland for making the review process so easy. Your comments and accommodation were very much appreciated and really helped polish this work. Thank you, too, Seth Kerney for coordinating the copy review. And also thanks to Vanessa Evans for ensuring that we always had everything we needed at every step of the way.

I'd like to extend exceptional thanks to our technical editors Roland Saville and John Johnston. Roland: You're one of the smartest persons I've had the pleasure of working with—and in so many fields. I don't know how your brain doesn't explode! You know I like to think of you as a "philosopher engineer," because you can take almost any design recommendation and find the corner-case counterargument where it breaks downs and doesn't apply. That's critically important to the process because by seeing from a distance where things can break you continually save us tremendous amounts of time in the lab, as well as ensuring the overall quality of the final designs. Thank you, too, JJ! You allowed me unfettered access to your massive labs in RTP and helped me along every step of the way. Your attention to detail is so impressive that I'm nearly spooked by your ability to catch the tiniest errors while reviewing hundreds of pages of configurations!

Finally, I owe a great deal of gratitude to my co-authors:

Ken: Thanks for your impressive knowledge and flexibility that you demonstrated by being able to jump right in and seamlessly adapt your research to our work in such an intuitive and cohesive manner. I've enjoyed working with you on many projects for the past decade and look forward to many more collaborations. Thanks again, Ken!

Christina: Thanks so much for coming out of retirement to work on one more project. Even though you're on the road more than Jack Kerouac these days, it was a real pleasure working with you again! Thanks for donning your QoS hat for us once again and bringing all your knowledge and experience to the table to help make this such a solid work.

Rob: Over the past 20 years we've been friends, classmates, roommates, workmates, "second-best" men at each other's weddings, and now co-authors. Your courage and determination are very inspiring. I honestly don't know if I would have taken my CCIE if I hadn't watched you do it. Same goes with running half-marathons (and one-day marathons!) Thanks for all your tremendous work on this project. It certainly was not for the faint-hearted, as every time we turned around we seemed to uncover yet another rabbit hole of technical issues that required yet more research and testing to be done. Thanks for sticking with it and seeing it through, Rob. But then again, that's just the kind of friend you are.

Rob Barton:

To begin, I would like to thank my very forgiving colleagues in the Cisco Vancouver office who have suffered through two years of trying to depend on an attention divided systems engineer who was more interested in solving theoretical QoS problems

than in helping his customers. Special thanks to my Cisco account team partner, Mike MacDonald, for his long-suffering patience, my manager, Ralph Wright, who enthusiastically supported this effort and always offered many words of encouragement, and to my director, Bill Kastelic, who eagerly gave me the flexibility to do this project. None of this would have been possible without the support from you guys.

I would also like to thank my lab administrator, Mojan Mobasser, for helping to get lab gear when I needed it the most. Testing these QoS solutions involved a lot of lab time, and without your support we would not have been able to build and test these solutions.

Special thanks goes out to Ian Procyk and my co-author Ken Briley who helped test some of the more difficult wireless scenarios. As well, I would like to thank Larry Ross for the many hours of emails and phone conversations discussing various wireless QoS solutions with me. Also thanks goes out to Kangwarn Chinthammit for helping with the AVC section review, and Scott Wainer who helped with the GET VPN work. All you guys were like my technical conscious during this project.

I'd also like to thank Bruno Wollmann from Terra-Comm Solutions who, while discussing my presentation at Cisco Live last year, introduced me to the concept of combining DMVPN with GET VPN to solve a real-world performance issue related to VoIP, which I think has made a great addition to the GET VPN chapter.

Chris Cleveland and Brett Bartow, thanks so much for your hard work on this project and supporting us all the way through. This project turned into a much bigger undertaking than any of us had expected, and instead of trying to apply your own QoS mechanism on our output, you let the creative juices flow, and in the end helped support a substantial work of technical literature.

Lastly, I'd like to thank Tim Szigeti. Not only have you been one of my closest friends for more than 20 years, you are also an inspiring engineer. Yes, I said engineer, the word you always tease me with. I can clearly remember the day this project started two and a half years ago; we were rewarding ourselves with a well-earned breakfast at the White Spot after one of our half-marathon training runs. I was complaining that your first edition of the End-to-End QoS book, while being a great book, was hopelessly out of date. Your response to me was unforgettable: "So why don't you help me write a new one?" That day was the start of this project, and although it was a long and difficult undertaking, it has also been an immensely rewarding experience. Thanks, Tim!

Kenneth Briley, Jr.:

First off I'd like to thank Roland Saville, for his guidance and clever insight when we worked through QoS on the Converged Access platforms.

To Stephen Orr, wireless is now awesome, before it was an illusion – thanks for the brilliant and oh so colorful commentary.

Many thanks to Tripti Agarwal, Saravanan Radhakrishnan, Anuj Kumar, and Bhavana Kudipudi without that team we would have never been able to deliver such a versatile platform.

Contents at a Glance

Part V: Data Center QoS Design

Part VI: WAN and Branch QoS Design

Part VII: MPLS VPN QoS Design

Part VIII: IPsec QoS Design

Part XI: Appendixes (Online)

Contents

Introduction

"Aren't we done with QoS yet?"

That's a question I get from time-to-time, which I like to answer along the lines of "As soon as we're done with availability and security, we'll be done with QoS also."

What I'm trying to express—although cheekily—is that although QoS has been around for a while, it is a foundational network infrastructure technology (the same as high-availability technologies and security technologies). And these foundational technologies will always prove to be integral components of any networking system, being present at the platform level, at the place in-the network (PIN) level and ultimately at the end-to-end network level.

Furthermore, such foundational network technologies are constantly evolving and expanding to meet new business and technical requirements. Such has been the case with QoS since the first edition of this work was published nearly 10 years ago.

For example, consider just one QoS-dependent application: video.

In 2004, there were really only two flavors of video traversing most enterprise networks: streaming video (unidirectional flows that benefited from both network- and application-level buffering to offset variations in transmission delays) and video conferencing (bidirectional 384-Kbps or 768-Kbps streams between dedicated hardware-based systems). So, we went into our massive Cisco Validation Labs in Research Triangle Park in North Carolina and hammered out best-practice designs to accommodate these two categories of video. We were done, right?

Wrong.

In the years that followed, codec and hardware advances made video production more cost-efficient and accessible, such that today nearly everyone with a smartphone has the ability to shoot high-definition video anytime and anywhere. Similarly, with the advent of social networking websites, video sharing and distribution suddenly became possible by anyone, anywhere (and that on a global scale!). Finally, video consumption also became possible anytime, anywhere, and on any device—thanks to advances in hardware and in wireless networking technologies.

That being the case, video is now the most dominant type of network traffic on the Internet and is expected to reach 90 percent within in a few years. Furthermore, there are many new forms and variations of video traffic, such as TelePresence, IP video surveillance, desktop video, and digital signage (just to name a few). And each of these types of video has unique service level requirements that must be met to ensure a high quality of experience by the end user. And thus, we circle back to QoS, which represents the enabling technologies to provide this quality of experience.

And that's just one application.

Advances in areas of data center and cloud networking, in addition to wireless networking, all have had corresponding impacts on QoS network designs.

Hence, a new edition of this book.

Another reason behind this second edition is to reflect the evolution of industry standards relating to QoS. Cisco has long advocated following industry standards and recommendations whenever deploying QoS, because this simplifies QoS designs, extends QoS policies beyond an administrative domain, and improves QoS policy effectiveness between administrative domains. Therefore, new standards, RFCs, and proposals have had—and will continue to have—a major impact on current and future strategic QoS designs.

A third key reason behind this new edition is that every network platform detailed in the original book has been replaced or significantly upgraded. So, the latest platforms (at the time of this writing) have been featured in this second version, with over a dozen Cisco product families being represented. In fact, nearly every design chapter features a different Cisco platform that suits the role being discussed, whether the role is a data center virtual switch, a branch router, a wireless LAN controller, a campus distribution switch, a WAN aggregator, a service provider core router, or so on.

And finally, QoS is a comprehensive and complex subject, one that entails a significant amount of fundamental technological concepts as well as platform-specific implementation detail. Therefore, it is often valuable for network administrators to have a single common reference on the subject, such as this book, which overviews all the relevant tools, presents various end-to-end strategies, and details platform-specific design recommendations for every major shipping Cisco platform.

And no, we're not done with QoS yet!

Objectives of This Book

The main objective of this book is to present—in a comprehensive and cohesive manner—the many aspects of quality of service design, including an overview of the tools, strategic and tactical design recommendations, and platform-specific configuration details. Therefore, novice to advanced network administrators alike can benefit from this volume as a single handy reference on this topic.

In addition, this exercise has produced multiple platform-specific configurations that can be viewed as QoS templates. As such, these templates can be considered roughly 80 percent of a generic enterprise or service provider QoS solution (borrowing from Pareto's 80/20 rule), to which another 20 percent of customizing and tailoring can be done to reach a final customer-specific solution. Considerations and rationales behind the presented designs are all explained so that administrators are fully informed of the rationale behind the designs and therefore can confidently modify these to meet their own specific requirements and constraints.

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

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

Finally, each design section has a case study chapter at the end that ties together many of the strategic principles, tactical recommendations, and platform-specific considerations that have been presented within the section. These case studies illustrate how to take generic and abstract design concepts and mold them to meet specific customer requirements. These case studies are indicative of what can be expected in real-life production environments. Each of these case study examples spans multiple devices, thus highlighting critical interrelationships. Furthermore, all case study chapters form respective parts of a single integrated end-to-end QoS network design.

Who Should Read This Book?

The primary reader of this book is the network administrator tasked with deploying QoS technologies. By extension, this group may also include other related IT professionals, such as systems administrators, audio/video specialists, VoIP specialists, and operations staff.

In addition, some readers may include technical decision makers tasked with evaluating the strategy and feasibility of QoS deployments, in addition to the drafting of implementation plans and phases toward these goals.

Yet another group of readers includes system engineers, partners, trainers, and other networking professionals who need to ramp-up technically on QoS technologies and designs, both for practical deployment purposes and to achieve various Cisco certifications.

Prerequisites are minimal, as the opening section of this book covers QoS technologies in high-to-mid-level technical detail, including protocols, tools, and relevant standards. In addition, each chapter includes extensive references for Additional Reading for more detailed information for readers unfamiliar with specific concepts discussed.

Because the content of the book ranges from a high level to a very low level of technical detail, it is suitable for a wide range of audiences, from intermediate to expert.

How This Book Is Organized

This book is organized into 39 chapters distributed across 8 parts, and includes 2 appendixes. Although this book can be read cover to cover, this organization allows readers to easily identify chapters of direct interest, thus facilitating the use of this book as a handy reference work. The eight parts of this book are described below:

Part I, "QoS Design Overview," introduces readers to QoS technologies, presenting a brief history and an architectural framework for these tools. Following this, groups of QoS tools are overviewed, including classification and marking tools, policing and shaping tools, queuing and dropping tools, bandwidth-reservation tools, and advanced tools like Medianet and application visibility and control.

Part II, "QoS Design Strategies," breaks away from a purely technical discussion to take a higher-level view of how business requirements drive QoS design. Application service-level requirements are analyzed, as are strategic QoS design best practices. This section concludes with the first case study chapter, illustrating the considerations that factor into defining an end-to-end QoS design strategy.

Part III, "Campus QoS Design," begins the exercise applying strategic QoS models to a tactical place in the network (PIN), which in this case is the enterprise campus. Campus-specific design considerations and recommendations are discussed at length, and subsequent chapters specialize in design recommendations for the access, distribution, and core layers of the campus network. A campus QoS design case study chapter completes the section.

Part IV, "Wireless LAN QoS Design," applies the strategic QoS models to the enterprise wireless LAN. Because WiFi is a unique media, as compared to the rest of the network, additional concepts need to be covered to explain how QoS can be achieved over-the-air. These considerations include the introduction of the Enhanced Distributed Coordination Function as well as IEEE 802.11e/Wireless Multimedia QoS. Following this, QoS design chapters address both the centralized wireless LAN controller deployment model and the new wired-and-wireless converged access deployment model. The section finishes with a WLAN QoS design case study.

Part V, "Data Center QoS Design," continues the application of QoS strategies, but this time to the data center network. Because of the convergence of storage-area networks and local-area networks within the data center, certain protocols require a completely lossless service that traditional QoS tools cannot guarantee. Therefore, data center-specific QoS tools are discussed, including the data center bridging toolset, which can be leveraged to guarantee such a lossless service. Following this, QoS design chapters address the virtual access layer, access and aggregation layers, and the core layer of data center networks. This part closes with a data center QoS design case study.

Part VI, "WAN and Branch QoS Design," expands the scope of discussion beyond the local area and applies strategic QoS principles to the wide-area network. QoS designs are presented for both WAN aggregation routers and for branch routers. This part ends with a WAN QoS design case study.

Part VII, "MPLS VPN QoS Design," continues the wide-area discussion but addresses QoS strategies for MPLS VPN networks, taking the perspectives of both the enterprise customer and the service provider into account in the end-to-end design. Design chapters are presented for the enterprise customer-edge router, the provider-edge router and the provider core routers. This section finishes with a case study on MPLS VPN QoS design.

Part VIII, "IPsec QoS Design," concludes the discussion by applying strategic QoS principles to IPsec VPNs. QoS designs are detailed for both Dynamic Multipoint VPNs and Group Encrypted Transport VPNs.

An overview on each of the 39 chapters (and the 2 appendixes) follows.

- **Chapter 1, "Introduction and Brief History of QoS and QoE":** Provides a brief history lesson on quality of service and quality of experience evolution, introducing fundamental QoS concepts, standards, and the evolutionary changes necessitating a second edition of this book.
- **Chapter 2, "IOS-Based QoS Architectural Framework and Syntax Structure":** Overviews how QoS tools interrelate, and introduces Cisco's IOS-based Modular QoS command-line interface (MQC), the common syntax structure for configuring QoS across most Cisco platforms.
- **Chapter 3, "Classification and Marking Tools":** Describes the various classification options for distinguishing one packet from another, which is the requisite first step in providing differentiated services. Also discussed are various marking options so that packets do not have to be reclassified at every network node.
- **Chapter 4, "Policing, Shaping, and Markdown Tools":** Discusses various tools that can be used to meter and regulate packet flows, including policers (which drop excess traffic), shapers (which delay excess traffic) and markers (which re-mark excess traffic).
- **Chapter 5, "Congestion Management and Avoidance Tools":** Considers options on how to deal with bottlenecks in the network, by addressing both queuing tools (to determine which packets get priority or preferential treatment during congestion), and early-dropping tools (to reduce the probability of congestion).
- **Chapter 6, "Bandwidth-Reservation Tools":** Introduces the concepts of bandwidth reservations and endpoint/infrastructure signaling to communicate how and when such reservations are to be made.
- **Chapter 7, "QoS in IPv6 Networks":** Examines IPv6 packet formats, classification and marking options, and how QoS tools are to be configured in IPv6 networks or in mixed IPv4 and IPv6 networks.
- **Chapter 8, "Medianet":** Gives a brief overview of the Medianet architecture, with particular focus on the aspects of Medianet specific to QoS configuration and monitoring.
- **Chapter 9, "Application Visibility and Control":** Presents deep packet inspection technologies for application identification, classification, and monitoring and how these can be used within the network.
- **Chapter 10, "Business and Application QoS Requirements":** Examines current business trends impacting QoS designs and various application-class QoS requirements.

- **Chapter 11, "QoS Design Principles and Strategies":** Combines the QoS tools and business requirements presented in preceding chapters and formulates these into QoS strategic models to address basic, intermediate, and advanced requirements.
- **Chapter 12, "Strategic QoS Design Case Study":** This first case study in the series introduces a fictional company, Tifosi Software, and discusses the business and technical considerations that come into play when defining an end-to-end QoS strategy.
- **Chapter 13, "Campus QoS Design Considerations and Recommendations":** Overviews various considerations and recommendations relating to campus QoS design, including trust boundaries, per-port versus per-VLAN design options, and EtherChannel QoS considerations.
- **Chapter 14, "Campus Access (Cisco Catalyst 3750) QoS Design":** This first platform-specific design chapter details best practice QoS designs at a configuration level for Cisco Catalyst 3750 series switches in the role of a campus access layer edge switch.
- **Chapter 15, "Campus Distribution (Cisco Catalyst 4500) QoS Design":** This design chapter details configuration recommendations for a Cisco Catalyst 4500 series switch in the role of a campus distribution layer switch. Additional designs include details on how this switch can be configured as a campus access-edge switch also.
- **Chapter 16, "Campus Core (Cisco Catalyst 6500) QoS Design":** This design chapter details configuration recommendations for a Cisco Catalyst 6500 series switch in the role of a campus core layer switch. Additional designs include details on how this switch can be configured as a campus access-edge or distribution layer switch as well.
- **Chapter 17, "Campus QoS Design Case Study":** This case study chapter describes how Tifosi Software has applied their strategic QoS design model to their campus network consisting of Cisco Catalyst 3750, 4500 and 6500 series switches.
- **Chapter 18, "Wireless LAN QoS Considerations and Recommendations":** Overviews various considerations and recommendations relating to wireless LAN QoS design and introduces WLAN QoS tools such as the Enhanced Distributed Coordination Function and Wireless Multimedia QoS.
- **Chapter 19, "Centralized (Cisco 5500 Wireless LAN Controller) QoS Design":** This design chapter details both GUI and CLI configuration recommendations for centralized wireless LAN controller (WLC) deployment models, featuring the Cisco 5500 WLC.
- **Chapter 20, "Converged Access (Cisco Catalyst 3850 and the Cisco 5760 Wireless LAN Controller QoS Design":** This design chapter details configuration recommendations for converged access WLAN deployment models, featuring the Cisco Catalyst 3850 series switch and the Cisco 5760 WLC.

- **Chapter 21, "Converged Access QoS Design Case Study":** This case study chapter describes how Tifosi Software has applied their strategic QoS design model to their wired-and-wireless converged access LAN network consisting of Cisco Catalyst 3850 series switches and the Cisco 5760 WLC.
- **Chapter 22, "Data Center QoS Design Considerations and Recommendations":** Overviews various considerations and recommendations relating to data center QoS design and introduces the data center bridging toolset.
- **Chapter 23, "Data Center Virtual Access (Nexus 1000V) QoS Design":** This design chapter details configuration recommendations for a Cisco Nexus 1000V series virtual switch in the role of a data center access layer switch.
- **Chapter 24, "Data Center Access/Aggregation (Nexus 5500/2000) QoS Design":** This design chapter details configuration recommendations for a Cisco Nexus 5500 series switch, which may include Cisco Nexus 2000 series Fabric Extenders, in the role of a data center access/aggregation switch.
- **Chapter 25, "Data Center Core (Nexus 7000) QoS Design":** This design chapter details configuration recommendations for a Cisco Nexus 7000 series switch in the role of a data center core switch. QoS designs for both M-Series and F-Series modules are detailed.
- **Chapter 26, "Data Center QoS Design Case Study":** This case study chapter describes how Tifosi Software has applied their strategic QoS design model to their data center network, consisting of Cisco Nexus 1000V, 5500/2000 and 7000 series switches.
- **Chapter 27, "WAN and Branch QoS Design Considerations and Recommendations":** Overviews various considerations and recommendations relating to WAN QoS design, including hardware versus software considerations, latency and jitter targets, and bandwidth-reservation options.
- **Chapter 28, "WAN Aggregator (Cisco ASR 1000) QoS Design":** This design chapter details configuration recommendations for a Cisco ASR 1000 series router in the role of a WAN aggregation router. WAN media featured includes leased lines, ATM, and Packet-Over-SONET.
- **Chapter 29, "Branch Router (Cisco ISR G2) QoS Design":** This design chapter details configuration recommendations for a Cisco ISR G2 series router in the role of a branch router, featuring Medianet and AVC designs.
- **Chapter 30, "WAN and Branch QoS Design Case Study":** This case study chapter describes how Tifosi Software has applied their strategic QoS design model to their wide-area network, consisting of Cisco ASR 1000 and ISR G2 series routers.
- **Chapter 31, "MPLS VPN QoS Design Considerations and Recommendations":** Overviews various considerations and recommendations relating to MPLS VPN QoS design, both from an enterprise and from a service provider perspective, including enterprise-to-provider mapping models and MPLS DiffServ tunneling modes. In

addition, this design section features carrier Ethernet as a WAN media.

- **Chapter 32, "Enterprise Customer Edge (Cisco ASR 1000 and ISR G2) QoS Design":** This design chapter details configuration recommendations for a Cisco ASR 1000 or ISR G2 series router in the role of an enterprise customer-edge router interfacing with a MPLS VPN service provider.
- **Chapter 33, "Service Provider Edge (Cisco ASR 9000) QoS Design":** This design chapter details configuration recommendations for a Cisco ASR 9000 series router in the role of a service provider edge router.
- **Chapter 34, "Service Provider Core (Cisco CRS) QoS Design":** This design chapter details configuration recommendations for a Cisco CRS-3 series router in the role of a service provider core router.
- **Chapter 35, "MPLS VPN QoS Design Case Study":** This case study chapter describes how Tifosi Software has adapted their strategic eight-class enterprise QoS model to integrate with their service provider's six class-of-service model, featuring Cisco ISR G2, ASR 1000, ASR 9000, and CRS-3 series routers.
- **Chapter 36, "IPsec VPN QoS Considerations and Recommendations":** Overviews various considerations and recommendations relating to IPsec VPN QoS design, including classification of encrypted packets, MTU considerations, and anti-replay implications.
- **Chapter 37, "DMVPN QoS Design":** This design chapter details configuration recommendations for Cisco ASR 1000 and ISR G2 routers in the roles of DMVPN hub-and-spoke routers (respectively).
- **Chapter 38, "GET VPN QoS Design":** This design chapter details configuration recommendations for Cisco ISR G2 routers in the roles of GET VPN routers.
- **Chapter 39, "Home Office VPN QoS Case Study":** This case study chapter describes how Tifosi Software has adapted their strategic QoS model over a DMVPN to provide telecommuting services to employees in their home offices. This case study features Cisco ASR 1002 series routers at the headend and ISR 881 series routers connected behind a broadband modem via Ethernet at the home office.
- **Appendix A, "AutoQoS for Medianet":** This online appendix overviews the latest evolution of the AutoQoS feature, which is based on the same QoS designs presented in this book. Detailed syntax is presented for the first platforms to support this feature, including the Cisco Catalyst 3750 and 4500 series switches.
- **Appendix B, "Control Plane Policing":** This online appendix overviews the control plane policing feature, which applies a QoS function (of policing) to a virtual interface (the control plane) to harden the network infrastructure from denial-of-service or worm attacks. Best-practice recommendations and configurations are presented for this feature.

This page intentionally left blank

Chapter 15

Campus Distribution (Cisco Catalyst 4500) QoS Design

The primary role of quality of service (QoS) in the campus distribution switch is to manage packet loss. Therefore, the distribution switch should trust differentiated services code point (DSCP) markings on ingress (as these have been previously set by access-edge switches) and perform both ingress (if required and supported) and egress queuing, as illustrated in Figure 15-1.

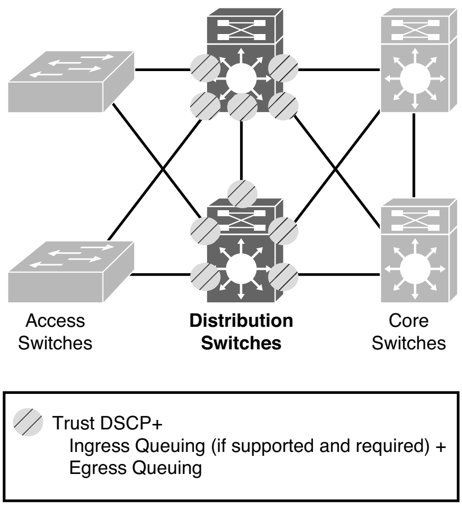

Figure 15-1 *Campus Distribution Switch Port QoS Roles*

The Cisco Catalyst 4500E Supervisor 7-E is a platform well suited to the role of a campus distribution switch and therefore is featured in this design chapter.

Incidentally, the QoS design requirements of a Catalyst 4500E Supervisor 7-E in the role of a distribution switch are generally equivalent to the requirements of a campus core switch.

Cisco Catalyst 4500 QoS Architecture

From a QoS perspective, the Cisco Catalyst 4500-E Supervisor 7-E is nearly identical to the Supervisor 6-E platform and the Catalyst 4500-X, because all of these platforms are Modular QoS command-line interface (MQC) based. However, earlier Catalyst 4500 platforms (such as the Supervisor II-Plus through Supervisor V-10GE) are Multi-Layer Switch (MLS)-QoS-based platforms and are referred to as *Classic Supervisors*.

Note QoS design for these older Classic Supervisors is beyond the scope for this design chapter. However, you can find design guidance for these platforms at http://www.cisco.com/en/US/docs/solutions/Enterprise/WAN_and_MAN/QoS_SRND_40/QoSCampus_40.html#wp1099634.

Figure 15-2 illustrates the QoS architecture for this Catalyst 4500E Supervisor 7-E (hereafter referred to simply as the Catalyst 4500) platform.

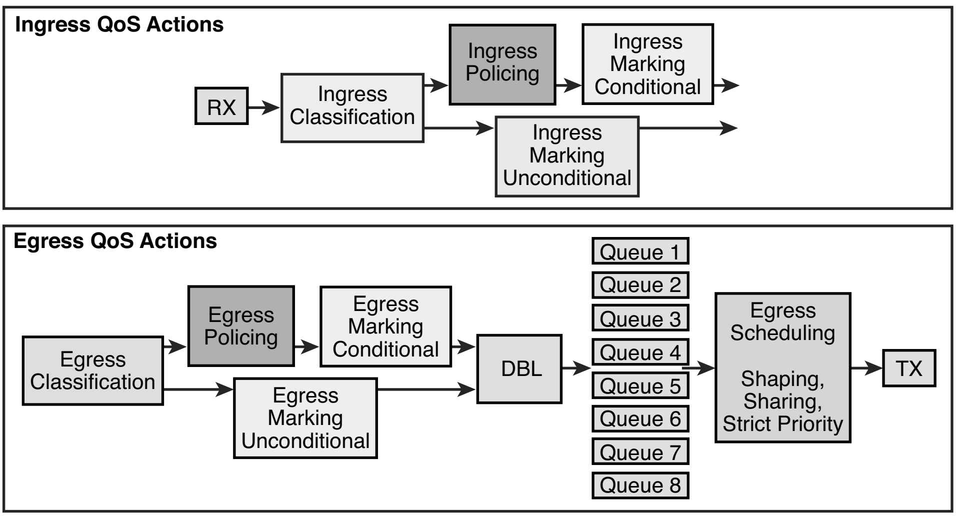

Figure 15-2 *Cisco Catalyst 4500 QoS Architectural Model*

QoS is enabled by default on all MQC-based platforms, which includes the Catalyst 4500. In addition, by default, all ports are set to a trust-DSCP/trust-CoS state.

In the MQC-based Catalyst 4500, QoS policies are applied as follows:

1. The incoming packet is classified (based on different packet fields, receive port, or VLAN) to belong to a traffic class.
2. Depending on the traffic class and configured polices, the packet is policed, which may result in the packet being dropped or re-marked.

3. After the packet has been marked/re-marked, it is looked up for forwarding. This action obtains the transmit port and VLAN to transmit the packet.
4. The packet is classified in the output direction based on the transmit port or VLAN/marking.
5. Depending on the output policies, the packet is policed, and may be dropped or re-marked.
6. The transmit queue for the packet is determined based on the traffic class and the configured egress queuing policies.
7. The transmit queue state is dynamically monitored via Dynamic Buffer Limiting (DBL) and drop threshold configuration to determine whether the packet should be dropped or queued for transmission.
8. If eligible for transmission, the packet is assigned to a transmit queue.

Based on these QoS operations, the design steps for configuring QoS on the Catalyst 4500 in the role of a distribution switch are discussed next.

QoS Design Steps

While there are two explicit QoS policy requirements of a distribution switch (namely to trust DSCP on ingress and queuing policies), because of the default QoS settings on MQC-based platforms there is effectively only a single step to configuring QoS on a Catalyst 4500 in this role:

1. Configure the ingress QoS model—which is recommended to be DSCP trust (and which is enabled by default on all MQC-based platforms).

> **Note** This step may include ingress queuing policies on platforms which support this feature (however, the Catalyst 4500 does not support ingress queuing).

2. Configure egress queuing.

Queuing Models

Ingress queuing is not supported on the Catalyst 4500; only egress queuing is supported.

> **Note** Other ingress QoS policies (including trust, classification, marking, and policing) are all supported; only ingress *queuing* is not supported on this platform.

The Catalyst 4500 supports a strict-priority hardware queue with (up to) seven additional nonpriority hardware queues. In addition, the Catalyst 4500 supports DSCP-to-queue mapping.

At the time of this writing, DSCP-based weighted random early detection (WRED) is not supported on the Catalyst 4500 platform. However, the Catalyst 4500 family uses a platform-specific congestion avoidance algorithm to provide active queue management (AQM), namely Dynamic Buffer Limiting (DBL). DBL tracks the queue length for each traffic flow in the switch. When the queue length of a flow exceeds its limit, DBL drop packets or sets the Explicit Congestion Notification (ECN) bits in the packet headers. The DBL algorithm can identify belligerent flows (that is, unchecked/nonadaptive/inelastic flows) and drop these more aggressively. Belligerent flows can use excessive bandwidth and switch buffers, resulting in poor application performance for well-behaved flows. Therefore, DBL can induce not only random "probabilistic drops" (in a manner similar to WRED), but also "belligerent flow drops," both of which are counted and displayed via the **show policy-map interface** command output on classes where DBL has been enabled (as demonstrated later in Example 15-4).

Therefore, the egress queuing model for the Catalyst 4500 platform can be expressed as 1P7Q1T+DBL.

Note DBL is unique to the Catalyst 4500 platforms. At the time of this writing, there are no tuning options for DBL.

The Catalyst 4500 can be configured to support 4-class, 8-class, or 12-class queuing models, as discussed in the following sections.

Four-Class Egress Queuing Model

In the four-class model (illustrated in Figures 11-3 and 11-4 in Chapter 11, "QoS Design Principles and Strategies"), the application class to queue mappings are as follows:

- Real-time traffic (marked EF) is assigned to the priority queue (which may be optionally policed to 30 percent bandwidth).
- Control traffic (marked CS3) is assigned to a dedicated nonpriority queue with a 10 percent bandwidth allocation.
- Transactional data (marked AF2) is assigned to another dedicated nonpriority queue with a 35 percent bandwidth allocation with DBL enabled.
- Best-effort traffic (marked DF) is assigned to a default queue with 25 percent bandwidth allocation with DBL enabled.

Note DBL is enabled *only* on the transactional data queue and the default queue (because real-time traffic and control traffic should never be early dropped).

Note When the priority queue is configured on one class of a policy map *without* a policer, only **bandwidth remaining percent** is accepted on other classes (guaranteeing a minimum bandwidth for other classes from the remaining bandwidth of what is left after using the priority queue). However, when the priority queue is configured *with* a policer, either **bandwidth percent** or **bandwidth remaining percent** is accepted on the other queuing classes.

Note If queuing policies are to be applied to EtherChannel interfaces, it is recommended not to police the priority queue. This is because two policy maps would be needed in this case: One policy map would be needed to police the priority queue (which would have to be applied to the logical EtherChannel interface in the egress direction), and a second policy map would be needed to define the queuing policy (using bandwidth remaining percent), which would be applied to all EtherChannel physical port-member interfaces in the egress direction. Therefore, to simplify the queuing policy and to increase its portability and modularity, the priority queue is not policed in the queuing design examples in this chapter (which necessitates the use of **bandwidth remaining percent** on nonpriority queues).

Note Although it is true that there will be fractional differences in bandwidth allotments to an application class depending on whether **bandwidth percent** or **bandwidth remaining percent** is used. However, because these differences are relatively minor, the same numeric values are used in these examples for the sake of consistency.

Figure 15-3 illustrates the resulting four-class (1P3Q1T+DBL) egress queuing model for the Catalyst 4500.

Example 15-1 shows the corresponding configuration for four-class (1P3Q1T+DBL) egress queuing on the Catalyst 4500.

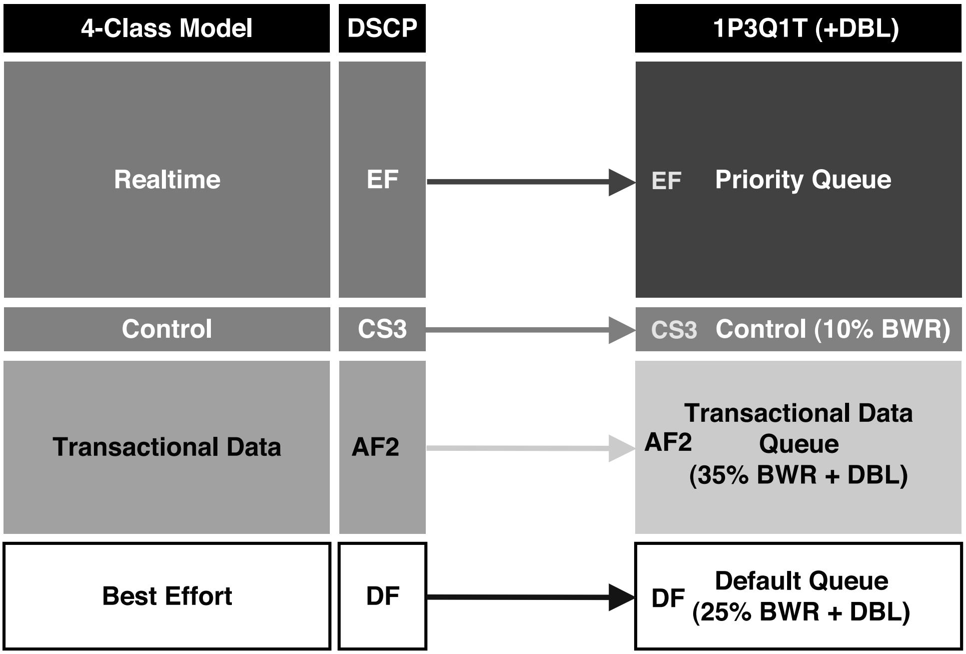

Figure 15-3 *Catalyst 4500 Four-Class Egress Queuing Model*

Example 15-1 *Four-Class (1P3Q1T+DBL) Egress Queuing Configuration Example on a Catalyst 4500*

```
 ! This section configures the class maps for the egress queuing policy
C4500(config)# class-map match-all PRIORITY-QUEUE
C4500(config-cmap)# match dscp ef
 ! VoIP (EF) is mapped to the PQ
C4500(config)# class-map match-all CONTROL-QUEUE
C4500(config-cmap)# match dscp cs3
 ! Signaling (CS3) is mapped to a dedicated queue
C4500(config)# class-map match-all TRANSACTIONAL-DATA-QUEUE
C4500(config-cmap)# match dscp af21 af22 af23
 ! Transactional Data (AF2) is mapped to a dedicated queue


 ! This section configures the four-class egress queuing policy map
C4500(config)# policy-map 1P3Q1T
C4500(config-pmap-c)# class PRIORITY-QUEUE
C4500(config-pmap-c)#  priority
 ! Enables the priority queue
C4500(config-pmap-c)# class CONTROL-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 10
```

```
 ! Defines the control queue with 10% BW remaining
C4500(config-pmap-c)# class TRANSACTIONAL-DATA-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 35
C4500(config-pmap-c)#  dbl
 ! Defines a transactional data queue with 35% BW remaining + DBL
C4500(config-pmap-c)# class class-default
C4500(config-pmap-c)#  bandwidth remaining percent 25
C4500(config-pmap-c)#  dbl
 ! Provisions the default/Best Effort queue with 25% BW remaining + DBL


 ! This section attaches the egress queuing policy to the interface(s)
C4500(config)# interface range TenGigabitEthernet 1/1-2
C4500(config-if-range)# service-policy output 1P3Q1T
```

Note Class maps defined for egress-queuing policies require unique names from any ingress-policy class maps; otherwise, classification errors can occur due to overlapping classification logic

You can verify the configuration in Example 15-1 with the following commands:

- **show class-map**
- **show policy-map**
- **show policy-map interface**

Eight-Class Egress Queuing Model

In the eight-class model (illustrated in Figures 11-5 and 11-6), the application class to queue mappings are as follows:

- Real-time traffic (marked EF) is assigned to the priority queue (which may be optionally policed to 10 percent bandwidth).
- Interactive video (marked AF4) is assigned to a dedicated nonpriority queue with a 23 percent bandwidth allocation with DBL enabled.
- Streaming video (marked AF3) is assigned to a dedicated nonpriority queue with a 10 percent bandwidth allocation with DBL enabled.
- Network control traffic (marked CS6) is assigned to a dedicated nonpriority queue with a 5 percent bandwidth allocation.
- Signaling traffic (marked CS3) is assigned to a dedicated nonpriority queue with a 2 percent bandwidth allocation.

- Transactional data (marked AF2) is assigned to dedicated nonpriority queue with a 24 percent bandwidth allocation with DBL enabled.
- Scavenger traffic (marked CS1) is constrained within a dedicated nonpriority queue with a 1 percent bandwidth allocation.
- Best-effort traffic (marked DF) is assigned to a default queue with 25 percent bandwidth allocation with DBL enabled.

Note As before, DBL is not enabled on the real-time or control traffic classes (because real-time traffic and control traffic should never be early dropped); nor would DBL be required on the scavenger class, because traffic in this class has no "good-faith" guarantee of service to begin with. Enabling DBL on the Interactive Video and Streaming Video classes assumes that the video codecs used for these flows are adaptive/elastic and therefore will adjust transmission rates in the event of congestion.

Figure 15-4 illustrates the resulting eight-class (1P7Q1T+DBL) egress queuing model for the Catalyst 4500.

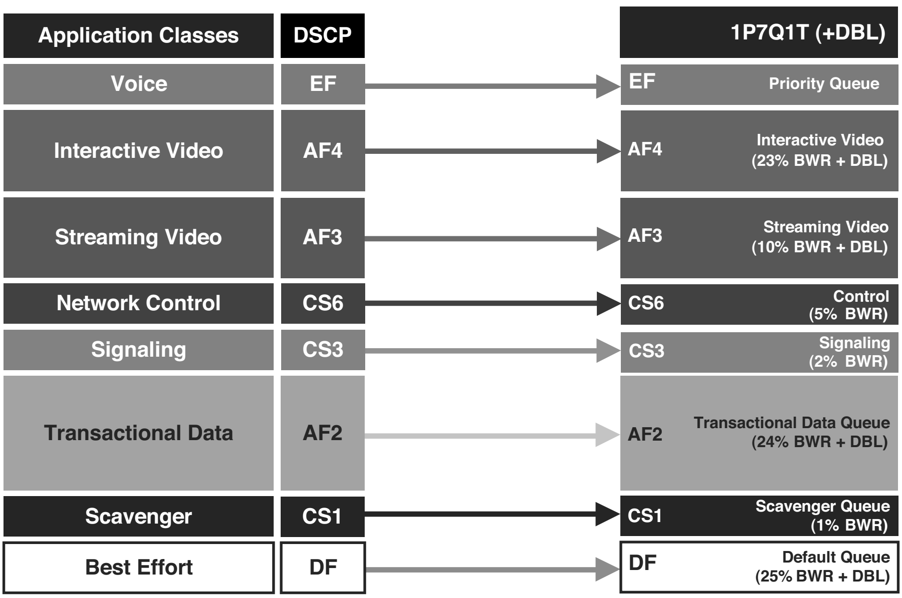

Figure 15-4 *Catalyst 4500 Eight-Class (1P7Q1T+DBL) Egress Queuing Model*

Example 15-2 shows the corresponding configuration for eight-class (1P7Q1T+DBL) egress queuing on the Catalyst 4500.

Example 15-2 *Eight-Class (1P7Q1T+DBL) Egress Queuing Configuration Example on a Catalyst 4500*

```
 ! This section configures the class maps for the egress queuing policy
C4500(config)# class-map match-all PRIORITY-QUEUE
C4500(config-cmap)# match dscp ef
 ! VoIP (EF) is mapped to the PQ
C4500(config)# class-map match-all INTERACTIVE-VIDEO-QUEUE
C4500(config-cmap)# match dscp af41 af42 af43
 ! Interactive-Video (AF4) is assigned a dedicated queue
C4500(config)# class-map match-all STREAMING-VIDEO-QUEUE
C4500(config-cmap)# match dscp af31 af32 af33
 ! Streaming-Video (AF3) is assigned a dedicated queue
C4500(config)# class-map match-all CONTROL-QUEUE
C4500(config-cmap)# match dscp cs6
 ! Network Control (CS6) is mapped to a dedicated queue
C4500(config)# class-map match-all SIGNALING-QUEUE
C4500(config-cmap)# match dscp cs3
 ! Signaling (CS3) is mapped to a dedicated queue
C4500(config)# class-map match-all TRANSACTIONAL-DATA-QUEUE
C4500(config-cmap)# match dscp af21 af22 af23
 ! Transactional Data (AF2) is assigned a dedicated queue
C4500(config)# class-map match-all SCAVENGER-QUEUE
C4500(config-cmap)# match dscp cs1
 ! Scavenger (CS1) is assigned a dedicated queue

 ! This section configures the 1P7Q1T+DBL egress queuing policy map
C4500(config)# policy-map 1P7Q1T
C4500(config-pmap-c)# class PRIORITY-QUEUE
C4500(config-pmap-c)#  priority
 ! Defines a priority queue
C4500(config-pmap-c)# class INTERACTIVE-VIDEO-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 23
C4500(config-pmap-c)#  dbl
 ! Defines a interactive-video queue with 23% BW remaining + DBL
C4500(config-pmap-c)# class STREAMING-VIDEO-QUEUE
C4500(config-pmap-c)# bandwidth remaining percent 10
C4500(config-pmap-c)#  dbl
 ! Defines a streaming-video queue with 10% BW remaining + DBL
C4500(config-pmap-c)# class CONTROL-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 5
 ! Defines a control/management queue with 5% BW remaining
C4500(config-pmap-c)# class SIGNALING-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 2
 ! Defines a signaling queue with 2% BW remaining
```

```
C4500(config-pmap-c)# class TRANSACTIONAL-DATA-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 24
C4500(config-pmap-c)#  dbl
 ! Defines a transactional data queue with 24% BW remaining + DBL
C4500(config-pmap-c)# class SCAVENGER-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 1
 ! Defines a (minimal) scavenger queue with 1% BW remaining/limit
C4500(config-pmap-c)# class class-default
C4500(config-pmap-c)#  bandwidth remaining percent 25
C4500(config-pmap-c)#  dbl
 ! Provisions the default/Best Effort queue with 25% BW remaining + DBL


 ! This section attaches the egress queuing policy to the interface(s)
C4500(config)# interface range TenGigabitEthernet 1/1-2
C4500(config-if-range)# service-policy output 1P7Q1T
```

You can verify the configuration in Example 15-2 with the following commands:

- **show class-map**
- **show policy-map**
- **show policy-map interface**

Twelve-Class Egress Queuing Model

In the 12-class model (illustrated in Figures 11-7 and 11-8), the application class to queue mappings are as follows:

- Voice (marked EF), broadcast video (marked CS5), and real-time interactive traffic (marked CS4) is all assigned to the priority queue (which may be optionally policed to 30 percent bandwidth).
- Multimedia-conferencing traffic (marked AF4) is assigned to a dedicated nonpriority queue with a 10 percent bandwidth allocation with DBL enabled.
- Multimedia-streaming traffic (marked AF3) is assigned to a dedicated nonpriority queue with a 10 percent bandwidth allocation with DBL enabled.
- Network control traffic (marked CS6), signaling traffic (marked CS3) and network management traffic (marked CS2) is all assigned to a dedicated nonpriority queue with a 10 percent bandwidth allocation; optionally, CS7 traffic may also be mapped to this queue.
- Transactional data traffic (marked AF2) is assigned to dedicated nonpriority queue with a 10 percent bandwidth allocation with DBL enabled.

- Bulk data traffic (marked AF1) is assigned to a dedicated nonpriority queue with 4 percent bandwidth allocation with DBL enabled.
- Scavenger traffic (marked CS1) is constrained within a dedicated nonpriority queue with a 1 percent bandwidth allocation.
- Best-effort traffic (marked DF) is assigned to a default queue with 25 percent bandwidth allocation with DBL enabled.

Figure 15-5 illustrates the resulting 12-class (1P7Q1T+DBL) egress queuing model for the Catalyst 4500.

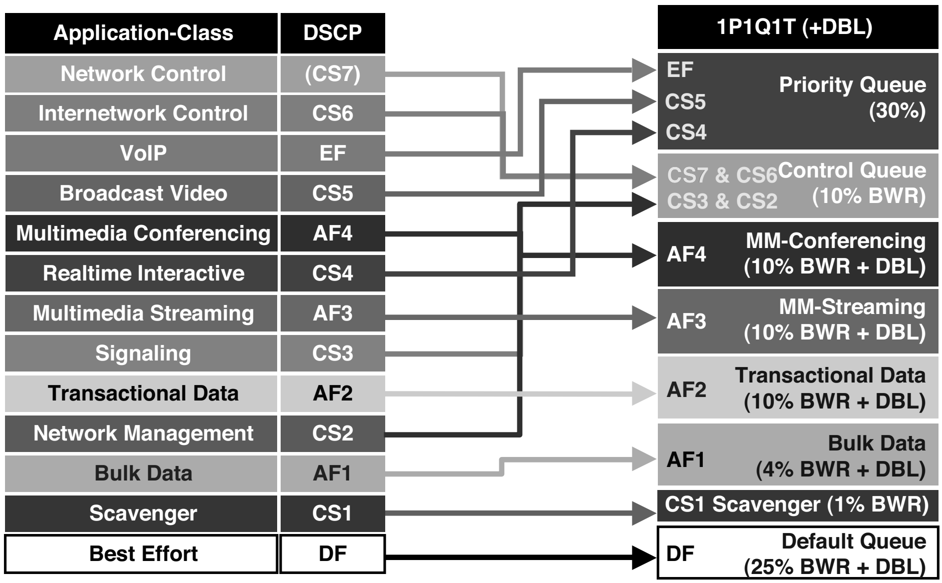

Figure 15-5 *Catalyst 4500 12-Class (1P7Q1T+DBL) Egress Queuing Model*

Example 15-3 shows the corresponding configuration for 12-class (1P7Q1T+DBL) egress queuing on the Catalyst 4500.

Example 15-3 *Twelve-Class (1P7Q1T+DBL) Egress Queuing Configuration Example on a Catalyst 4500*

```
 ! This section configures the class maps for the egress queuing policy
C4500(config)# class-map match-any PRIORITY-QUEUE
C4500(config-cmap)# match dscp ef
C4500(config-cmap)# match dscp cs5
C4500(config-cmap)# match dscp cs4
 ! VoIP (EF), Broadcast Video (CS5) and Realtime Interactive (CS4)
 ! are all mapped to the PQ
```

```
C4500(config)# class-map match-any CONTROL-MGMT-QUEUE
C4500(config-cmap)# match dscp cs7
C4500(config-cmap)# match dscp cs6
C4500(config-cmap)# match dscp cs3
C4500(config-cmap)# match dscp cs2
 ! Network Control (CS7), Internetwork Control (CS6),
 ! Signaling (CS3) and Management (CS2) are mapped
 ! to a Control/Management Queue
C4500(config)# class-map match-all MULTIMEDIA-CONFERENCING-QUEUE
C4500(config-cmap)# match dscp af41 af42 af43
 ! Multimedia Conferencing (AF4) is assigned a dedicated queue
C4500(config)# class-map match-all MULTIMEDIA-STREAMING-QUEUE
C4500(config-cmap)# match dscp af31 af32 af33
 ! Multimedia Streaming (AF3) is assigned a dedicated queue
C4500(config)# class-map match-all TRANSACTIONAL-DATA-QUEUE
C4500(config-cmap)# match dscp af21 af22 af23
 ! Transactional Data (AF2) is assigned a dedicated queue
C4500(config)# class-map match-all BULK-DATA-QUEUE
C4500(config-cmap)# match dscp af11 af12 af13
 ! Bulk Data (AF1) is assigned a dedicated queue
C4500(config)# class-map match-all SCAVENGER-QUEUE
C4500(config-cmap)# match dscp cs1
 ! Scavenger (CS1) is assigned a dedicated queue


 ! This section configures the 1P7Q1T+DBL egress queuing policy map
C4500(config)# policy-map 1P7Q1T
C4500(config-pmap-c)# class PRIORITY-QUEUE
C4500(config-pmap-c)#  priority
 ! Defines a priority queue
C4500(config-pmap-c)# class CONTROL-MGMT-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 10
 ! Defines a control/management queue with 10% BW remaining
C4500(config-pmap-c)# class MULTIMEDIA-CONFERENCING-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 10
C4500(config-pmap-c)#  dbl
 ! Defines a multimedia conferencing queue with 10% BW remaining + DBL
C4500(config-pmap-c)# class MULTIMEDIA-STREAMING-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 10
C4500(config-pmap-c)#  dbl
 ! Defines a multimedia streaming queue with 10% BW remaining + DBL
C4500(config-pmap-c)# class TRANSACTIONAL-DATA-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 10
C4500(config-pmap-c)#  dbl
```

```
! Defines a transactional data queue with 10% BW remaining + DBL
C4500(config-pmap-c)# class BULK-DATA-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 4
C4500(config-pmap-c)#  dbl
 ! Defines a bulk data queue with 10% BW remaining + DBL
C4500(config-pmap-c)# class SCAVENGER-QUEUE
C4500(config-pmap-c)#  bandwidth remaining percent 1
 ! Defines a (minimal) scavenger queue with 1% BW remaining/limit
C4500(config-pmap-c)# class class-default
C4500(config-pmap-c)#  bandwidth remaining percent 25
C4500(config-pmap-c)#  dbl
 ! Provisions the default/Best Effort queue with 25% BW remaining + DBL


 ! This section attaches the egress queuing policy to the interface(s)
C4500(config)# interface range TenGigabitEthernet 1/1-2
C4500(config-if-range)# service-policy output 1P7Q1T
```

You can verify the configuration in Example 15-3 with the following commands:

- **show class-map**
- **show policy-map**
- **show policy-map interface** (as shown in Example 15-4)

Example 15-4 *Verifying Queuing Policies on a Catalyst 4500:* ***show policy-map interface***

```
C4500# show policy-map interface TenGigabitEthernet 1/1
 TenGigabitEthernet1/1
  Service-policy output: 1P7Q1T
    Class-map: PRIORITY-QUEUE (match-any)
      102598 packets
      Match:  dscp ef (46)
        102598 packets
      Match:  dscp cs5 (40)
        0 packets
      Match:  dscp cs4 (32)
        0 packets
      priority queue:
          Transmit: 22782306 Bytes, Queue Full Drops: 0 Packets

    Class-map: CONTROL-MGMT-QUEUE (match-any)
      24847 packets
```

```
      Match:  dscp cs7 (56)
        0 packets
      Match:  dscp cs6 (48)
        0 packets
      Match:  dscp cs3 (24)
        24847 packets
      Match:  dscp cs2 (16)
        0 packets
      bandwidth remaining 10 (%)
          Transmit: 24909844 Bytes, Queue Full Drops: 0 Packets


    Class-map: MULTIMEDIA-CONFERENCING-QUEUE (match-all)
      22280511 packets
      Match:  dscp af41 (34) af42 (36) af43 (38)
      bandwidth remaining 10 (%)
          Transmit: 4002626800 Bytes, Queue Full Drops: 0 Packets
      dbl
          Probabilistic Drops: 0 Packets
          Belligerent Flow Drops: 0 Packets


    Class-map: MULTIMEDIA-STREAMING-QUEUE (match-all)
      0 packets
      Match:  dscp af31 (26) af32 (28) af33 (30)
      bandwidth remaining 10 (%)
          Transmit: 0 Bytes, Queue Full Drops: 0 Packets
      dbl
          Probabilistic Drops: 0 Packets
          Belligerent Flow Drops: 0 Packets


    Class-map: TRANSACTIONAL-DATA-QUEUE (match-all)
      235852 packets
      Match:  dscp af21 (18) af22 (20) af23 (22)
      bandwidth remaining 10 (%)
          Transmit: 247591260 Bytes, Queue Full Drops: 0 Packets
      dbl
          Probabilistic Drops: 0 Packets
          Belligerent Flow Drops: 0 Packets


    Class-map: BULK-DATA-QUEUE (match-all)
      2359020 packets
      Match:  dscp af11 (10) af12 (12) af13 (14)
```

```
      bandwidth remaining 4 (%)
          Transmit: 2476460700 Bytes, Queue Full Drops: 0 Packets
      dbl
          Probabilistic Drops: 0 Packets
          Belligerent Flow Drops: 0 Packets


    Class-map: SCAVENGER-QUEUE (match-all)
      78607323 packets
      Match:  dscp cs1 (8)
      bandwidth remaining 1 (%)
          Transmit: 98144078642 Bytes, Queue Full Drops: 26268 Packets


    Class-map: class-default (match-any)
      12388183 packets
      Match: any
        12388183 packets
      bandwidth remaining 25 (%)
          Transmit: 13001465825 Bytes, Queue Full Drops: 0 Packets
      dbl
          Probabilistic Drops: 0 Packets
          Belligerent Flow Drops: 0 Packets
C4500#
```

Example 15-4 shows various queuing classes and their associated packet and byte counts, including 26,268 queuing drops noted on the scavenger queue.

Additional Platform-Specific QoS Design Options

These designs represent a generic building block for Catalyst 4500 QoS in a campus distribution switch role, but they are by no means the only design options available to you. Additional options and considerations include the following:

- Access-edge design options
- Per-VLAN QoS design
- Per-port/per-VLAN QoS design
- EtherChannel QoS design
- AutoQoS SRND4
- Control plane policing

Each of these additional QoS design options is discussed in turn.

Access-Edge Design Options

This chapter has focused on QoS designs for the Catalyst 4500 in the role of a campus distribution switch (which are generally equivalent to the QoS designs required were it serving in the role of a campus core switch). However, the Catalyst 4500 can also be deployed as a campus access switch. Therefore, a few additional design options would apply in such a role, including the following access-edge models:

- Conditional Trust Model
- Classification and Marking Model
- Classification, Marking, and Policing Model

Each of these access-edge design options will be discussed in turn.

Conditional Trust Model

As previously mentioned, MQC-based platforms trust at Layer 2 and Layer 3 by default and therefore do not require any explicit commands to perform such functions. Therefore, there are no equivalent commands to **mls qos trust cos** or **mls qos trust dscp** (nor are any required).

However, there is a need to provide conditional trust functionality for all switch platforms that may be deployed in the role of an access switch. Hence, there is a corresponding command for conditional trust on the Catalyst 4500 (namely, **qos trust device**).

At the time of this writing, the Catalyst 4500 supports conditional trust for the following devices:

- Cisco IP phone via the **cisco-phone** keyword option
- Cisco TelePresence systems via the **cts** keyword option
- Cisco IP video surveillance cameras systems via the **ip-camera** keyword option
- Cisco Digital Media Players via the **media-player** keyword option

When extending conditional trust to Cisco IP phones, it is important to remember that these can only re-mark class of service (CoS) bits (on PC-generated traffic). Therefore, the Conditional Trust Model on the Catalyst 4500 requires a dynamic conditional trust policy applied to the port in conjunction with a simple MQC policy that explicitly matches CoS 5 (for voice) and CoS 3 (for signaling) and marks the DSCP values of these packets to EF and CS3, respectively (essentially performing a CoS-to-DSCP mapping). Example 15-5 shows this conditional trust model for the Catalyst 4500.

Example 15-5 *Configuring (CoS-Based) Conditional Trust to a Cisco IP Phone on a Catalyst 4500*

```
 ! This section defines the class maps to match Voice and Signaling
C4500(config-cmap)# class-map match-all VOICE
C4500(config-cmap)#  match cos 5
C4500(config-cmap)# class-map match-all SIGNALING
C4500(config-cmap)#  match cos 3


 ! This section defines the CoS-to-DSCP re-marking policy map
C4500(config-cmap)# policy-map CISCO-IPPHONE
C4500(config-pmap)#  class VOICE
C4500(config-pmap-c)# set dscp ef
 ! Maps CoS 5 to DSCP EF
C4500(config-pmap-c)# class SIGNALING
C4500(config-pmap-c)#  set dscp cs3
 ! Maps CoS 3 to DSCP CS3
C4500(config-pmap-c)# class class-default
C4500(config-pmap-c)#  set dscp default
 ! All other traffic is set to DSCP DF


 ! This section applies conditional trust and policy map to the int(s)
C4500(config)# interface GigabitEthernet 3/1
C4500(config-if)# switchport access vlan 10
C4500(config-if)# switchport voice vlan 110
C4500(config-if)# spanning-tree portfast
C4500(config-if)# qos trust device cisco-phone
 ! Applies conditional-trust to the switch port
C4500(config-if)# service-policy input CISCO-IPPHONE
 ! Attaches the CoS-to-DSCP mapping policy map
```

You can verify the configuration in Example 15-5 with the following commands:

- **show qos interface**
- **show class-map**
- **show policy-map**
- **show policy-map interface**

Medianet Metadata Classification Model

Beginning with Cisco IOS Release IOS XE 3.3.0SG and IOS 15.1(1)SG, you can configure a class map with metadata filters. A QoS policy that includes such classes is termed a metadata-based QoS policy. It allows you to classify flows based on user-friendly metadata attributes rather than on access control list (ACL)-based classification criteria (such as source/destination addresses/ports, and so on).

The following restrictions apply to using a metadata-based QoS policy on a Catalyst 4500 series switch:

- They can only be attached to target in input direction.
- They can only be attached to physical ports and EtherChannel port channel interfaces; they cannot be attached to VLANs, port VLANs, and switch virtual interfaces (SVIs).
- A policy can have multiple metadata-based classifiers.
- A class map can have one or more metadata filters with **match-any** or **match-all** semantics.
- Policy actions corresponding to metadata class are applied on aggregate traffic; however, if the metadata filter is configured along with Flexible NetFlow record filter, the policy action (like policer) applies on individual flows.

Note Flow-based QoS policies and Flexible NetFlow (FNF) are discussed further in a following section.

Example 15-6 illustrates a metadata-based QoS policy with two classes using metadata filters.

Example 15-6 *Medianet Metadata Classification Policy Example on a Catalyst 4500*

```
 ! This section configures the medianet metadata class maps
C4500(config-cmap)# class-map match-all REALTIME-INTERACTIVE
C4500(config-cmap)#  match application telepresence-media
 ! Identifies TelePresence media flows via metadata
C4500(config-cmap)# class-map match-any MULTIMEDIA-CONFERENCING
C4500(config-cmap)#  match application webex-video
 ! Identifies WebEx video flows via metadata
C4500(config-cmap)#  match application webex-voice
 ! Identifies WebEx voice flows via metadata
```

You can verify the configuration in Example 15-6 with the following commands:

- **show class-map**
- **show policy-map**
- **show policy-map interface**

Classification and Marking Models

In many scenarios, trust models may not be available or sufficient to distinctly classify all types of traffic required by the end-to-end QoS strategic model. Therefore, explicit classification and marking policies may be needed at the access edge.

Example 15-7 shows a configuration example based on Figure 11-5 (An eight-class QoS model).

Note As previously discussed, not all application classes may be present at the access edge on ingress. For example, streaming video would likely not be present at the access edge on ingress (as these flows are not *sourced* from campus endpoints, but are likely *destined* to them), nor would network control flows be sourced from campus endpoints. Therefore, these classes would not need to be included in the access-edge classification and marking policy map.

Note Referenced access lists are omitted from the policy examples for brevity.

Example 15-7 *Classification and Marking Policy Example on a Catalyst 4500*

```
 ! This section configures the class maps
C4500(config-cmap)# class-map match-all VOICE
C4500(config-cmap)#  match dscp ef
 ! Voice is matched on DSCP EF
C4500(config-cmap)# class-map match-all INTERACTIVE-VIDEO
C4500(config-cmap)#  match access-group name INTERACTIVE-VIDEO
 ! Associates INTERACTIVE-VIDEO access-list with class map
C4500(config-cmap)# class-map match-all SIGNALING
C4500(config-cmap)#  match cs3
 ! Signaling is matched on DSCP CS3
C4500(config-cmap)# class-map match-all TRANSACTIONAL-DATA
C4500(config-cmap)#  match access-group name TRANSACTIONAL-DATA
 ! Associates TRANSACTIONAL-DATA access-list with class map
C4500(config-cmap)# class-map match-all SCAVENGER
C4500(config-cmap)#  match access-group name SCAVENGER
 ! Associates SCAVENGER access-list with class map
```

```
 ! This section configures the Per-Port ingress marking policy map
C4500(config-cmap)# policy-map PER-PORT-MARKING
C4500(config-pmap)#  class VOICE
C4500(config-pmap-c)# set dscp ef
 ! VoIP is marked EF
C4500(config-pmap-c)# class INTERACTIVE-VIDEO
C4500(config-pmap-c)#  set dscp af41
 ! Interactive-Video is marked AF41
C4500(config-pmap-c)# class SIGNALING
C4500(config-pmap-c)#  set dscp cs3
 ! Signaling is marked CS3
C4500(config-pmap-c)# class TRANSACTIONAL-DATA
C4500(config-pmap-c)#  set dscp af21
 ! Transactional Data is marked AF21
C4500(config-pmap-c)# class SCAVENGER
C4500(config-pmap-c)#  set dscp cs1
 ! Scavenger traffic is marked CS1
C4500(config-pmap-c)# class class-default
C4500(config-pmap-c)#  set dscp default
 ! All other traffic is marked DF


 ! This section attaches the service-policy to the interface(s)
C4500(config)# interface range GigabitEthernet 2/1-48
C4500(config-if-range)# switchport access vlan 10
C4500(config-if-range)# switchport voice vlan 110
C4500(config-if-range)# spanning-tree portfast
C4500(config-if-range)# qos trust device cisco-phone
 ! The interface is set to conditionally trust Cisco IP Phones
C4500(config-if-range)# service-policy input PER-PORT-MARKING
 ! Attaches the Per-Port Marking policy to the interface(s)
```

You can verify the configuration in Example 15-7 with the following commands:

- **show qos interface**
- **show class-map**
- **show policy-map**
- **show policy-map interface**

Classification, Marking, and Policing Model

In addition to classification and marking, policing might also be required at the access edge. The Catalyst 4500 can perform single-rate (two-color) policing and three-color policing—via either the RFC 2697 single-rate three-color marker (srTCM) or the RFC 2698 two-rate three-color marker (trTCM). Example 15-8 shows a per-port single-rate policing example for the Catalyst 4500 (based on Figure 13-8), and Example 15-9 shows policy amendments to support a RFC 2698 two-rate three-color marker.

Example 15-8 *(Single-Rate Two-Color) Per-Port Policing Configuration Example on a Catalyst 4500*

```
 ! This section configures the single-rate per-port policing policy map
C4500(config)# policy-map PER-PORT-POLICING
C4500(config-pmap)# class VVLAN-VOIP
C4500(config-pmap-c)#  set dscp ef
C4500(config-pmap-c)#  police 128k bc 8000
C4500(config-pmap-c-police)# conform-action transmit
C4500(config-pmap-c-police)# exceed-action drop
 ! VoIP is marked EF and policed to drop at 128 kbps
C4500(config-pmap)# class VVLAN-SIGNALING
C4500(config-pmap-c)#  set dscp cs3
C4500(config-pmap-c)#  police 32k bc 8000
C4500(config-pmap-c-police)# conform-action transmit
C4500(config-pmap-c-police)# exceed-action drop
 ! (VVLAN) Signaling is marked CS3 and policed to drop at 32 Kbps
C4500(config-pmap)# class MULTIMEDIA-CONFERENCING
C4500(config-pmap-c)#  set dscp af41
C4500(config-pmap-c)#  police 5m bc 8000
C4500(config-pmap-c-police)# conform-action transmit
C4500(config-pmap-c-police)# exceed-action drop
 ! Multimedia-conferencing is marked AF41 and policed to drop at 5 Mbps
C4500(config-pmap)# class SIGNALING
C4500(config-pmap-c)#  set dscp cs3
C4500(config-pmap-c)#  police 32k bc 8000
C4500(config-pmap-c-police)# conform-action transmit
C4500(config-pmap-c-police)# exceed-action drop
 ! (DVLAN) Signaling is marked CS3 and policed to drop at 32 Kbps
C4500(config-pmap)# class TRANSACTIONAL-DATA
C4500(config-pmap-c)#  set dscp af21
C4500(config-pmap-c)#  police 10m bc 8000
C4500(config-pmap-c-police)# conform-action transmit
C4500(config-pmap-c-police)# exceed-action set-dscp-transmit af22
 ! Trans-data is marked AF21 and policed to re-mark (to AF22) at 10 Mbps
C4500(config-pmap)# class BULK-DATA
C4500(config-pmap-c)#  set dscp af11
C4500(config-pmap-c)#  police 10m bc 8000
C4500(config-pmap-c-police)# conform-action transmit
C4500(config-pmap-c-police)# exceed-action set-dscp-transmit af12
```

```
 ! Bulk-data is marked AF11 and policed to re-mark (to AF12) at 10 Mbps
C4500(config-pmap)# class SCAVENGER
C4500(config-pmap-c)#  set dscp cs1
C4500(config-pmap-c)#  police 10m bc 8000
C4500(config-pmap-c-police)# conform-action transmit
C4500(config-pmap-c-police)# exceed-action drop
 ! Scavenger traffic is marked CS1 and policed to drop at 10 Mbps
C4500(config-pmap)# class class-default
C4500(config-pmap-c)#  set dscp default
C4500(config-pmap-c)#  police 10m bc 8000
C4500(config-pmap-c-police)# conform-action transmit
C4500(config-pmap-c-police)# exceed-action set-dscp-transmit cs1
 ! The implicit default class marks all other traffic to DF
 ! and polices all other traffic to re-mark (to CS1) at 10 Mbps


 ! This section attaches the service-policy to the interface(s)
C4500(config)# interface range GigabitEthernet 2/1-48
C4500(config-if-range)# switchport access vlan 10
C4500(config-if-range)# switchport voice vlan 110
C4500(config-if-range)# spanning-tree portfast
C4500(config-if-range)# qos trust device cisco-phone
 ! The interface is set to conditionally trust Cisco IP phones
C4500(config-if-range)# service-policy input PER-PORT-POLICING
 ! Attaches the Per-Port Policing policy to the interface(s)
```

Note The Catalyst 4500 IOS Software allows for policing rates to be entered using the postfixes **k** (for kilobits), **m** (for megabits), and **g** (for gigabits), as shown in Example 15-8. In addition, decimal points are allowed in conjunction with these postfixes. For example, a rate of 10.5 Mbps could be entered with the policy map command **police 10.5m**. These policing rates are converted to their full bits-per-second values within the configuration, but it makes the entering of these rate more user friendly and less error prone (as could easily be the case when having to enter up to 10 zeros to define the policing rate).

You can verify the configuration in Example 15-8 with the following commands:

- **show qos interface**
- **show class-map**
- **show policy-map**
- **show policy-map interface**

To avoid excessive repetition, Example 15-9 amends and expands the policer from a single-rate two-color marker to a two-rate three-color marker only on a single class (the Bulk Data class). However, similar amendments can be made on any Assured Forwarding (AF) class of traffic.

Example 15-9 *(Two-Rate Three-Color) Per-Port Policing Configuration Amendment Example on a Catalyst 4500*

```
 ! This section configures a dual-rate per-port policing policy map
C4500(config)# policy-map TWO-RATE-POLICER

<snip>

C4500(config-pmap)# class BULK-DATA
C4500(config-pmap-c)# set dscp af11
C4500(config-pmap-c)# police 10m bc 8000 pir 15m
  ! Bulk-data is policed to 10 Mbps rate and 15 Mbps peak rate
C4500(config-pmap-c-police)# conform-action set-dscp-transmit af11
 ! Bulk data under 10 Mbps will be marked AF11
C4500(config-pmap-c-police)# exceed-action set-dscp-transmit af12
 ! Bulk data traffic between 10 Mbps and 15 Mbps will be marked AF12
C4500(config-pmap-c-police)# violate-action set-dscp-transmit af13
 ! Bulk data traffic over 15Mbps will be marked AF13
```

You can verify the configuration in Example 15-9 with the following commands:

- **show qos interface**
- **show class-map**
- **show policy-map**
- **show policy-map interface**

Per-VLAN QoS Design

The Catalyst 4500 supports VLAN-based QoS. However, unlike the Catalyst 3750, the Catalyst 4500 does not support the **mls qos vlan-based** interface command. Furthermore, service policies are attached to VLANs via the VLAN configuration mode (instead of the interface configuration mode), as shown in Example 15-10.

Example 15-10 *Per-VLAN Marking Configuration Example on a Catalyst 4500*

```
 ! This section configures the interface(s) for conditional trust,
C4500(config)# interface range GigabitEthernet 2/1-48
C4500(config-if-range)# switchport access vlan 10
```

```
C4500(config-if-range)# switchport voice vlan 110
C4500(config-if-range)# spanning-tree portfast
C4500(config-if-range)# qos trust device cisco-phone
 ! The interface is set to conditionally trust Cisco IP phones


 ! This section attaches a marking policy to the DVLAN
C4500(config)# vlan config 10
C4500(config-vlan-config)# service-policy input DVLAN-MARKING


 ! This section attaches a marking policy to the VVLAN
C4500(config)# vlan config  110
C4500(config-vlan-config)# service-policy input VVLAN-MARKING
```

You can verify the configuration in Example 15-10 with the following commands:

- **show qos interface**
- **show class-map**
- **show policy-map**
- **show policy-map vlan** *vlan-number* (This command is nearly identical to show policy map interface, except that it references a VLAN directly, rather than a VLAN interface.)

Note It is not recommended to deploy policing policies on a per-VLAN basis, as discussed further in the next section.

Per-Port/Per-VLAN QoS

Although it is technically possible to apply a (aggregate) policing policy on a per-VLAN basis, it is not advisable to do so. This is because the number of endpoints in a given VLAN can dynamically vary, yet the policing rates are statically fixed at an aggregate level, resulting in unpredictable bandwidth allotments per endpoint.

However, a more flexible and discrete approach for deploying policing policies exists on the Catalyst 4500 platforms—namely, to deploy these on a per-port/per-VLAN basis. The Catalyst 4500 has a very elegant syntax for deploying per-port/per-VLAN policies, as follows: Within a (trunked) switch port's interface configuration, each VLAN carried over that trunked port can have a separate policy applied to it via an **interface-vlan** configuration mode, as shown in Example 15-11.

Example 15-11 *Per-Port/Per-VLAN Policing Configuration Example on a Catalyst 4500*

```
! This section attaches the policy to the VLANs on a per-port basis
C4500(config)# interface range GigabitEthernet 2/1-48
C4500(config-if-range)# switchport access vlan 10
C4500(config-if-range)# switchport voice vlan 110
C4500(config-if-range)# spanning-tree portfast
C4500(config-if-range)# qos trust device cisco-phone
 ! The interface is set to conditionally trust Cisco IP phones
C4500(config-if-range)# vlan 10
C4500(config-if-vlan-range)# service-policy input DVLAN-POLICERS
 ! Attaches the per-port/per-VLAN DVLAN policing policy to the
 ! DVLAN of the trunked switch port(s)
C4500(config-if-range)# vlan 110
C4500(config-if-vlan-range)# service-policy input VVLAN-POLICERS
 ! Attaches the per-port/per-VLAN VVLAN policing policy to the
 ! VVLAN of the trunked switch port(s)
```

You can verify the configuration in Example 15-11 with the following commands:

- **show qos interface**
- **show class-map**
- **show policy-map**
- **show policy-map interface**
- **show policy-map interface** *interface x/y* **vlan** *vlan-number*

EtherChannel QoS Design

The following rules apply when deploying QoS service policies on Catalyst 4500 EtherChannels:

- Classification, marking, and policing policies (whether ingress or egress) are applied to the logical port channel interfaces.
- Queuing policies are applied to the physical port-member interfaces.

For EtherChannel interfaces configured on Catalyst 4500 switches, *the ingress QoS policies* (including classification, marking, and policing policies) are applied via MQC **service-policy** statements (in the ingress direction using the **input** keyword) configured on the *logical port channel interface*. Trust statements are not required because this MQC-based platform trusts by default.

In addition, the Catalyst 4500 supports *egress QoS policies* (including marking/policing policies) to be similarly applied via MQC **service-policy** statements (in the egress direction using the **output** keyword) on the *logical port channel interface*.

Egress queuing policies, however, are applied via MQC **service-policy** statements (in the egress direction using the **output** keyword) on the *physical port-member interfaces*, as shown in Example 15-12.

Example 15-12 *EtherChannel QoS Design on a Catalyst 4500*

```
! This section configures the logical port channel interface
C4500(config)# interface Port-channel1
C4500(config-if)# description ETHERCHANNEL-LOGICAL-INTERFACE
C4500(config-if)# switchport mode trunk
C4500(config-if)# switchport trunk encapsulation dot1q
C4500(config-if)# switchport trunk allowed vlan 10,110
C4500(config-if)# service-policy input MARKING


 ! This section configures 1P3Q1T+DBL queuing on physical port-member interfaces
C4500(config)# interface range TenGigabitEthernet1/1-2
C4500(config-if-range)# description PORT-CHANNEL1-PORT-MEMBER
C4500(config-if-range)# switchport mode trunk
C4500(config-if-range)# switchport trunk encapsulation dot1q
C4500(config-if-range)# switchport trunk allowed vlan 10,110
C4500(config-if-range)# channel-group 1 mode auto
C4500(config-if-range)# service-policy output 1P7Q1T-QUEUING
 ! Applies 1P7Q1T+DBL-QUEUING queuing policy to physical port member
```

You can verify the configuration in Example 15-12 with the following commands:

- **show class-map**
- **show policy-map**
- **show policy-map interface**

Note As previously stated, the queuing policies will only attach to EtherChannel port-member physical interfaces if the priority queue is not explicitly policed. If policing the priority queue is desired, a separate policy map needs to be constructed to do so and attached to the logical EtherChannel interface in the *egress* direction.

Flow-Based QoS

Flow-based QoS enables microflow policing and marking capability to dynamically learn traffic flows, providing the capability to police every unique flow to an individual rate. Flow-based QoS is available on a Catalyst 4500 series switch with the built-in NetFlow hardware support. It can be applied to ingress traffic on both switched and routed interfaces with flow masks defined using Flexible NetFlow (FNF). Flow-based QoS is typically used in environments where per-user, granular rate limiting is required. Flow-based QoS is also referred to as user-based rate limiting (UBRL).

A *flow* is defined as a stream of packets having the same properties as those defined by the key fields in the FNF flow record. A new flow is created when the value of data in packet's key fields is unique with respect to the flows that already exist.

A flow-based QoS policy is possesses one or more class maps matching on a FNF flow record. Such a class map must be configured as **match-all** to match all the match criteria specified in the class map. When a flow-based QoS policy is attached to a QoS target, ingress traffic on the target is first classified based on the classification rules specified in the class map. If the classifier has an FNF flow record, the key fields specified in the FNF flow record are applied on the classified traffic to create flows provided the flow does not already exist. The corresponding policy actions (policing and marking) are then applied to these individual flows. Flow-based policers (termed microflow policers) rate limit each unique flow. Flows are dynamically created and inactive flows are periodically aged out.

Flow-based QoS policy can be applied on a per-port basis, per-port/per-VLAN basis, or on an EtherChannel port channel interface (but only in the ingress direction). Therefore, flow-based QoS may be deployed at either the access layer or distribution layer (wherever UBRL may be of value).

Note that flow-based policies will apply to all flows matched within a given class. For example, if a flow-based policer is applied to the default class and attached to port or VLAN, *all* flows originating from that port or VLAN (respectively) will be subject to the policer. If this is not to be the intent, additional classification is recommended and the flow-based policer should be more selectively applied.

Example 15-13 shows how to configure a flow-based QoS policy that uses microflow policing in the context of user-based rate limiting. Any and all flows sourced from the subnet 192.168.10.* are microflow policed to 1 Mbps.

Example 15-13 *Configuring Flow-Based QoS (UBRL) on Catalyst 4500*

```
 ! This section defines an ACL to match traffic from subnet
C4500(config)# ip access-list extended USERGROUP-1
C4500(config-ext-nacl)# permit ip 192.168.10.0 0.0.0.255 any
 ! Traffic sourced from the 1922.168.10.x subnet is matched
```

```
 ! This section defines a flow record with source address as key
C4500(config)# flow record FLOW-RECORD-1
C4500(config-flow-record)# match ipv4 source address
 ! Source address is defined as the key tuple


 ! This section defines the class map to match on USERGROUP-1 ACL
 ! and specify FLOW-RECORD-1 definition for flow creation
C4500(config)# class-map match-all USER-GROUP-1
C4500(config-cmap)# match access-group name USERGROUP-1
C4500(config-cmap)# match flow record FLOW-RECORD-1
 ! A "match-all" class map binds the ACL and flow-record
 ! to identify unique flows


 ! This section defines the microflow policer policy map
C4500(config)# policy-map 1MBS-MICROFLOW-POLICER
C4500(config-pmap)# class USER-GROUP-1
C4500(config-pmap-c)# police cir 1m
C4500(config-pmap-c-police)# conform-action transmit
C4500(config-pmap-c-police)# exceed-action drop
 ! Specifies each discrete microflow is to be limited to 1Mbs


 ! This section applies the microflow policer to the interface
C4500(config)# interface gigabitEthernet3/1
C4500(config-if)# service-policy input 1MBS-MICROFLOW-POLICER
```

You can verify the configuration in Example 15-13 with the following commands:

- **show flow record** (demonstrated in Example 15-14)
- **show class-map**
- **show policy-map**
- **show policy-map interface**

Example 15-14 *Verifying Flow-Based QoS Policies on a Catalyst 4500:* ***show flow record***

```
C4500# show flow record
flow record FLOW-RECORD-1:
  Description:        User defined
  No. of users:       1
```

```
Total field space:  4 bytes
Fields:
  match ipv4 source address
```

AutoQoS SRND4

AutoQoS SRND4 is supported on the Cisco Catalyst 4500 beginning with Cisco IOS Release IOS XE 3.3.0SG and IOS 15.1(1)SG and is detailed in Appendix A, "AutoQoS for Medianet."

Control Plane Policing

Control plane policing (CPP) is supported on the Catalyst 4500 and is detailed in Appendix B, "Control Plane Policing."

Summary

This design chapter primarily discussed the best-practice QoS design recommendations for the Cisco Catalyst 4500 (Supervisor 6-E/7-E) series switch in the role of a campus distribution layer switch. (which, incidentally are equivalent to the QoS designs required were it serving in the role of a campus core switch).

Because the Catalyst 4500 is an MQC-based QoS platform, QoS is enabled by default, as is DSCP trust, on all ports. Therefore, there is effectively only a single step to configuring QoS on a Catalyst 4500 performing the role of a distribution switch: to configure an egress queuing policy.

To this end, 4-class, 8-class, and 12-class queuing policies were detailed, along with corresponding configurations and verification examples, leveraging the Catalyst 4500's flexible 1P7Q1T+DBL hardware queuing capabilities.

Additional platform-specific design options and considerations were discussed, including how the Catalyst 4500 could be deployed as an access-edge switch, and how to configure per-VLAN QoS, per-port/per-VLAN QoS, and EtherChannel QoS designs.

AutoQoS SRND4 is supported on the Catalyst 4500 and is covered in Appendix A; similarly, CPP is also supported and is covered in Appendix B.

Further Reading

Cisco Enterprise Medianet Campus QoS Design 4.0: http://www.cisco.com/en/US/docs/solutions/Enterprise/WAN_and_MAN/QoS_SRND_40/QoSCampus_40.html

Medianet Campus QoS Design At-A-Glance: http://www.cisco.com/en/US/docs/solutions/Enterprise/Video/qoscampusaag.html

Medianet Catalyst 4500 QoS Design At-A-Glance: http://www.cisco.com/en/US/docs/solutions/Enterprise/Video/qoscampuscat4500aag.html

Cisco Catalyst 4500 Series Switch Software Configuration Guide, Release IOS XE 3.3.0SG and IOS 15.1(1)SG—QoS Configuration Guide: http://www.cisco.com/en/US/docs/switches/lan/catalyst4500/15.1/XE_330SG/configuration/guide/qos_mrg.html

Index

A

B

D

E

F

G

H

N

O

P

Q

R

S

T

U

V