Practice tests

Video Training

Flash Cards

Review Exercises

Study Planner

Official Cert Guide

Advance your IT career with hands-on learning

CCNP and CCIE Enterprise Core

ENCOR 350-401

BRADLEY EDGEWORTH, CCIE® No. 31574

RAMIRO GARZA RIOS, CCIE® No. 15469

JASON GOOLEY, CCIE® No. 38759

DAVID HUCABY, CCIE® No. 4594

ciscopress.com

CCNP and CCIE Enterprise Core

ENCOR 350-401

Official Cert Guide

BRAD EDGEWORTH, CCIE No. 31574
RAMIRO GARZA RIOS, CCIE No. 15469
DAVID HUCABY, CCIE No. 4594
JASON GOOLEY, CCIE No. 38759

Cisco Press

CCNP and CCIE Enterprise Core ENCOR 350-401 Official Cert Guide

Brad Edgeworth, Ramiro Garza Rios, David Hucaby, Jason Gooley

Published by:

Cisco Press

ScoutAutomatedPrintCode

Library of Congress Control Number: 2019951592

ISBN-13: 978-1-58714-523-0

ISBN-10: 1-58714-523-5

Warning and Disclaimer

This book is designed to provide information about the CCNP and CCIE Enterprise Core Exam. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

The information is provided on an "as is" basis. The authors, Cisco Press, and Cisco Systems, Inc. shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

The opinions expressed in this book belong to the author and are not necessarily those of Cisco Systems, Inc.

Trademark Acknowledgments

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc., cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Special Sales

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact intlcs@pearson.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 email at feedback@ciscopress.com. Please make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Editor-in-Chief: Mark Taub

Alliances Manager, Cisco Press: Arezou Gol

Director, ITP Product Management: Brett Bartow

Managing Editor: Sandra Schroeder

Development Editor: Ellie Bru

Senior Project Editor: Tonya Simpson

Copy Editor: Kitty Wilson

Technical Editor(s): Richard Furr, Denise Fishburne, Dmitry Figol, Patrick Croak

Editorial Assistant: Cindy Teeters

Cover Designer: Chuti Prasertsith

Composition: codeMantra

Indexer: Tim Wright

Proofreader: Abigail Manheim

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**

About the Authors

Brad Edgeworth, CCIE No. 31574 (R&S and SP), is a systems architect at Cisco Systems. Brad is a distinguished speaker at Cisco Live, where he has presented on various topics. Before joining Cisco, Brad worked as a network architect and consultant for various Fortune 500 companies. Brad's expertise is based on enterprise and service provider environments, with an emphasis on architectural and operational simplicity. Brad holds a bachelor of arts degree in computer systems management from St. Edward's University in Austin, Texas. Brad can be found on Twitter as @BradEdgeworth.

Ramiro Garza Rios, CCIE No. 15469 (R&S, SP, and Security), is a solutions architect in the Cisco Customer Experience (CX) organization. His expertise is on enterprise and service provider network environments, with a focus on evolving architectures and next-generation technologies. He is also a Cisco Live distinguished speaker. Ramiro recently concluded a multi-year Cisco ACI project for one of the top three Tier 1 ISPs in the United States.

Before joining Cisco Systems in 2005, he was a network consulting and presales engineer for a Cisco Gold Partner in Mexico, where he planned, designed, and implemented both enterprise and service provider networks.

David Hucaby, CCIE No. 4594 (R&S), CWNE No. 292, is a lead network engineer for the University of Kentucky Healthcare, where he focuses on wireless networks in a large medical environment. David holds bachelor's and master's degrees in electrical engineering from the University of Kentucky. He has been authoring Cisco Press titles for 20 years.

Jason Gooley, CCIE No. 38759 (R&S and SP), is a very enthusiastic and spontaneous person who has more than 20 years of experience in the industry. Currently, Jason works as a technical solutions architect for the Worldwide Enterprise Networking Sales team at Cisco Systems. Jason is very passionate about helping others in the industry succeed. In addition to being a Cisco Press author, Jason is a distinguished speaker at Cisco Live, contributes to the development of the Cisco CCIE and DevNet exams, provides training for Learning@Cisco, is an active CCIE mentor, is a committee member for the Cisco Continuing Education Program (CE), and is a program committee member of the Chicago Network Operators Group (CHI-NOG), www.chinog.org. Jason also hosts a show called MetalDevOps. Jason can be found at www.MetalDevOps.com, @MetalDevOps, and @Jason_Gooley on all social media platforms.

About the Technical Reviewers

Richard Furr, CCIE No. 9173 (R&S and SP), is a technical leader in the Cisco Customer Experience (CX) organization, providing support for customers and TAC teams around the world. Richard has authored and acted as a technical editor for Cisco Press publications. During the past 19 years, Richard has provided support to service provider, enterprise, and data center environments, resolving complex problems with routing protocols, MPLS, IP Multicast, IPv6, and QoS.

Denise "Fish" Fishburne, CCDE No. 2009::0014, CCIE No. 2639 (R&S and SNA), is a solutions architect with Cisco Systems. Fish is a geek who absolutely adores learning and passing it on. Fish has been with Cisco since 1996 and has worn many varying "hats," such as TAC engineer, advanced services engineer, CPOC engineer, and now solutions architect. Fish is heavily involved with Cisco Live, which is a huge passion of hers. Outside of Cisco, you will find her actively sharing and "passing it on" on her blog site, YouTube Channel, and Twitter. Look for Fish swimming in the bits and bytes all around you or just go to www.NetworkingWithFish.com.

Dmitry Figol, CCIE No. 53592 (R&S), is a systems engineer in Cisco Systems Enterprise Sales. He is in charge of design and implementation of software applications and automation systems for Cisco. His main expertise is network programmability and automation. Before joining Cisco Sales, Dmitry worked on the Cisco Technical Assistance Center (TAC) Core Architecture and VPN teams. Dmitry maintains several open-source projects and is a regular speaker at conferences. He also does live streams on Twitch about network programmability and Python. Dmitry holds a bachelor of science degree in telecommunications. Dmitry can be found on Twitter as @dmfigol.

Patrick Croak, CCIE No. 34712 (Wireless), is a systems engineer with a focus on wireless and mobility. He is responsible for designing, implementing, and optimizing enterprise wireless networks. He also works closely with the business unit and account teams for product development and innovation. Prior to this role, he spent several years working on the TAC Support Escalation team, troubleshooting very complex wireless network issues. Patrick has been with Cisco since 2006.

Dedications

Brad Edgeworth:

This book is dedicated to my wife, Tanya, and daughter, Teagan. The successes and achievements I have today are because of Tanya. Whenever I failed an exam, she provided the support and encouragement to dust myself off and try again. She sacrificed years' worth of weekends while I studied for my CCIE certifications. Her motivation has allowed me to overcome a variety of obstacles with great success.

To Teagan, thank you for bringing me joy and the ability to see life through the eyes of an innocent child.

David Hucaby:

As always, my work is dedicated to my wife and my daughters, for their love and support, and to God, who has blessed me with opportunities to learn, write, and work with so many friends.

Jason Gooley:

This book is dedicated to my wife, Jamie, and my children, Kaleigh and Jaxon. Without the support of them, these books would not be possible. To my father and brother, thank you for always supporting me.

Ramiro Garza:

I would like to dedicate this book to my wonderful and beautiful wife, Mariana, and to my four children, Ramiro, Frinee, Felix, and Lucia, for their love, patience, and support as I worked on this project. And to my parents, Ramiro and Blanca D., and my in-laws, Juan A. and Marisela, for their continued support and encouragement. And most important of all, I would like to thank God for all His blessings in my life.

Acknowledgments

Brad Edgeworth:
A debt of gratitude goes to my co-authors, Ramiro, Jason, and David. The late-night calls were kept to a minimum this time. I'm privileged to be able to write a book with David; I read his BCMSN book while I was studying for my CCNP 11 years ago.

To Brett Bartow, thank you for giving me the privilege to write on such an esteemed book. I'm thankful to work with Ellie Bru again, along with the rest of the Pearson team.

To the technical editors—Richard, Denise, Dmitry, and Patrick—thank you for finding our mistakes before everyone else found them.

Many people within Cisco have provided feedback and suggestions to make this a great book, including Craig Smith, Vinit "Count Vinny" Jain, Dustin Schuemann, and Steven "no-redistribution" Allspach.

Ramiro Garza Rios:
I'd like to give a special thank you to Brett Bartow for giving us the opportunity to work on this project and for being our guiding light. I'm also really grateful and honored to have worked with Brad, Jason, and David; they are amazing and great to work with. I'd like to give special recognition to Brad for providing the leadership for this project. A big thank you to the Cisco Press team for all your support, especially to Ellie Bru. I would also like to thank our technical editors—Denise, Richard, Patrick, and Dmitry—for their valuable feedback to ensure that the technical content of this book is top-notch. And most important of all, I would like to thank God for all His blessings in my life.

David Hucaby:
I am very grateful to Brett Bartow for giving me the opportunity to work on this project. Brad, Ramiro, and Jason have been great to work with. Many thanks to Ellie Bru for her hard work editing our many chapters!

Jason Gooley:
Thank you to the rest of the author team for having me on this book. It has been a blast! Thanks to Brett and the whole Cisco Press team for all the support and always being available. This project is near and dear to my heart, as I am extremely passionate about helping others on their certification journey.

Contents at a Glance

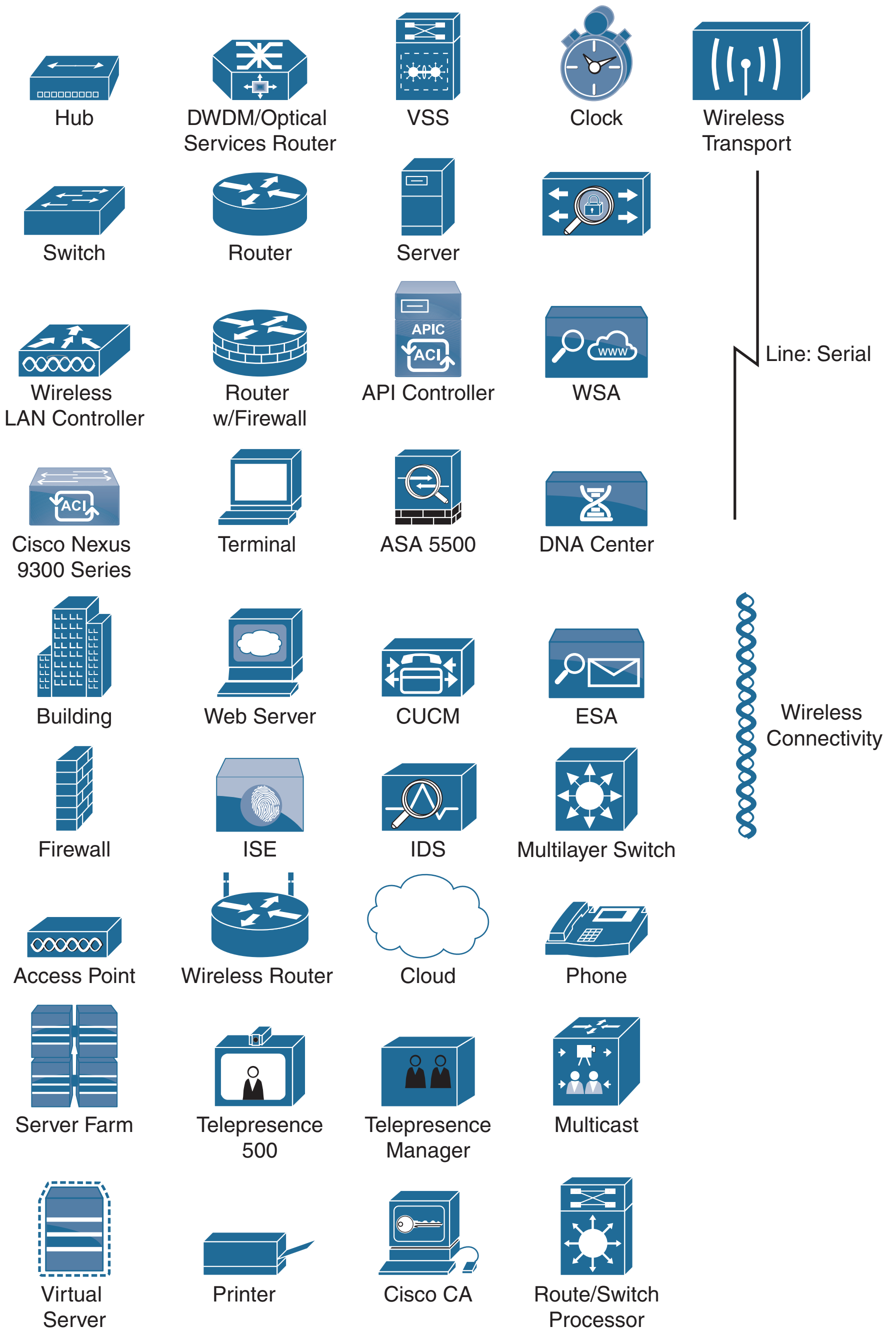
Reader Services

Register your copy at www.ciscopress.com/title/9781587145230 for convenient access to downloads, updates, and corrections as they become available. To start the registration process, go to www.ciscopress.com/register and log in or create an account.* Enter the product ISBN 9781587145230 and click Submit. When the process is complete, you will find any available bonus content under Registered Products.

*Be sure to check the box that you would like to hear from us to receive ex
discounts on future editions of this product.

Contents

Icons Used in This Book



Command Syntax Conventions

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

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

Introduction

Congratulations! If you are reading this Introduction, then you have probably decided to obtain a Cisco certification. Obtaining a Cisco certification will ensure that you have a solid understanding of common industry protocols along with Cisco's device architecture and configuration. Cisco has a high market share of routers and switches, with a global footprint.

Professional certifications have been an important part of the computing industry for many years and will continue to become more important. Many reasons exist for these certifications, but the most popularly cited reason is credibility. All other factors being equal, a certified employee/consultant/job candidate is considered more valuable than one who is not certified.

Cisco provides three primary certifications: Cisco Certified Network Associate (CCNA), Cisco Certified Network Professional (CCNP), and Cisco Certified Internetwork Expert (CCIE). Cisco is making changes to all three certifications, effective February 2020. The following are the most notable of the many changes:

- The exams will include additional topics, such as programming.
- The CCNA certification is not a prerequisite for obtaining the CCNP certification. CCNA specializations will not be offered anymore.
- The exams will test a candidate's ability to configure and troubleshoot network devices in addition to answering multiple-choice questions.
- The CCNP is obtained by taking and passing a Core exam and a Concentration exam.
- The CCIE certification requires candidates to pass the Core written exam before the CCIE lab can be scheduled.

CCNP Enterprise candidates need to take and pass the CCNP and CCIE Enterprise Core ENCOR 350-401 examination. Then they need to take and pass one of the following Concentration exams to obtain their CCNP Enterprise:

- **300-410 ENARSI:** Implementing Cisco Enterprise Advanced Routing and Services (ENARSI)
- **300-415 ENSDWI:** Implementing Cisco SD-WAN Solutions (SDWAN300)
- **300-420 ENSLD:** Designing Cisco Enterprise Networks (ENSLD)
- **300-425 ENWLSD:** Designing Cisco Enterprise Wireless Networks (ENWLSD)
- **300-430 ENWLSI:** Implementing Cisco Enterprise Wireless Networks (ENWLSI)
- **300-435 ENAUTO:** Implementing Automation for Cisco Enterprise Solutions (ENAUI)

Be sure to visit www.cisco.com to find the latest information on CCNP Concentration requirements and to keep up to date on any new Concentration exams that are announced.

CCIE Enterprise candidates need to take and pass the CCNP and CCIE Enterprise Core ENCOR 350-401 examination. Then they need to take and pass the CCIE Enterprise Infrastructure or Enterprise Wireless lab exam.

Goals and Methods

The most important and somewhat obvious goal of this book is to help you pass the CCNP and CCIE Enterprise Core ENCOR 350-401 exam. In fact, if the primary objective of this book were different, then the book's title would be misleading; however, the methods used in this book to help you pass the exam are designed to also make you much more knowledgeable about how to do your job.

One key methodology used in this book is to help you discover the exam topics that you need to review in more depth, to help you fully understand and remember those details, and to help you prove to yourself that you have retained your knowledge of those topics. This book does not try to help you simply memorize; rather, it helps you truly learn and understand the topics. The CCNP and CCIE Enterprise Core exam is just one of the foundation topics in the CCNP certification, and the knowledge contained within is vitally important to being a truly skilled routing/switching engineer or specialist. This book would do you a disservice if it didn't attempt to help you learn the material. To that end, the book will help you pass the CCNP and CCIE Enterprise Core exam by using the following methods:

- Helping you discover which test topics you have not mastered
- Providing explanations and information to fill in your knowledge gaps
- Supplying exercises and scenarios that enhance your ability to recall and deduce the answers to test questions

Who Should Read This Book?

This book is not designed to be a general networking topics book, although it can be used for that purpose. This book is intended to tremendously increase your chances of passing the CCNP and CCIE Enterprise Core exam. Although other objectives can be achieved from using this book, the book is written with one goal in mind: to help you pass the exam.

So why should you want to pass the CCNP and CCIE Enterprise Core ENCOR 350-401 exam? Because it's one of the milestones toward getting the CCNP certification or to being able to schedule the CCIE lab—which is no small feat. What would getting the CCNP or CCIE mean to you? It might translate to a raise, a promotion, and recognition. I would certainly enhance your resume. It would demonstrate that you are serious about continuing the learning process and that you're not content to rest on your laurels. It might please your reseller-employer, who needs more certified employees for a higher discount from Cisco. Or you might have one of many other reasons.

Strategies for Exam Preparation

The strategy you use to prepare for the CCNP and CCIE Enterprise Core ENCOR 350-401 exam might be slightly different from strategies used by other readers, depending on the skills, knowledge, and experience you already have obtained. For instance, if you have attended the CCNP and CCIE Enterprise Core ENCOR 350-401 course, then you might take a different approach than someone who learned switching via on-the-job training.

Regardless of the strategy you use or the background you have, the book is designed to help you get to the point where you can pass the exam with the least amount of time required. For instance, there is no need for you to practice or read about IP addressing and subnetting if you fully understand it already. However, many people like to make sure that they truly know a topic and thus read over material that they already know. Several features of this book will help you gain the confidence that you need to be convinced that you know some material already and to also help you know what topics you need to study more.

The Companion Website for Online Content Review

All the electronic review elements, as well as other electronic components of the book, exist on this book's companion website.

How to Access the Companion Website

To access the companion website, which gives you access to the electronic content with this book, start by establishing a login at www.ciscopress.com and registering your book. To do so, simply go to www.ciscopress.com/register and enter the ISBN of the print book: 9781587145230. After you have registered your book, go to your account page and click the Registered Products tab. From there, click the Access Bonus Content link to get access to the book's companion website.

Note that if you buy the Premium Edition eBook and Practice Test version of this book from Cisco Press, your book will automatically be registered on your account page. Simply go to your account page, click the Registered Products tab, and select Access Bonus Content to access the book's companion website.

How to Access the Pearson Test Prep (PTP) App

You have two options for installing and using the Pearson Test Prep application: a web app and a desktop app. To use the Pearson Test Prep application, start by finding the registration code that comes with the book. You can find the code in these ways:

- **Print book:** Look in the cardboard sleeve in the back of the book for a piece of paper with your book's unique PTP code.
- **Premium Edition:** If you purchase the Premium Edition eBook and Practice Test directly from the Cisco Press website, the code will be populated on your account page after purchase. Just log in at www.ciscopress.com, click Account to see details of your account, and click the digital purchases tab.
- **Amazon Kindle:** For those who purchase a Kindle edition from Amazon, the access code will be supplied directly from Amazon.
- **Other Bookseller E-books:** Note that if you purchase an e-book version from any other source, the practice test is not included because other vendors to date have not chosen to vend the required unique access code.

NOTE Do not lose the activation code because it is the only means with which you can access the QA content with the book.

Once you have the access code, to find instructions about both the PTP web app and the desktop app, follow these steps:

Step 1. Open this book's companion website, as shown earlier in this Introduction under the heading "How to Access the Companion Website."

Step 2. Click the Practice Exams button.

Step 3. Follow the instructions listed there both for installing the desktop app and for using the web app.

Note that if you want to use the web app only at this point, just navigate to www.pearsontestprep.com, establish a free login if you do not already have one, and register this book's practice tests using the registration code you just found. The process should take only a couple of minutes.

NOTE Amazon eBook (Kindle) customers: It is easy to miss Amazon's email that lists your PTP access code. Soon after you purchase the Kindle eBook, Amazon should send an email. However, the email uses very generic text, and makes no specific mention of PTP or practice exams. To find your code, read every email from Amazon after you purchase the book. Also do the usual checks for ensuring your email arrives, like checking your spam folder.

NOTE Other eBook customers: As of the time of publication, only the publisher and Amazon supply PTP access codes when you purchase their eBook editions of this book.

How This Book Is Organized

Although this book could be read cover to cover, it is designed to be flexible and allow you to easily move between chapters and sections of chapters to cover just the material that you need more work with. If you do intend to read them all, the order in the book is an excellent sequence to use.

The book includes the following chapters:

- **Chapter 1, "Packet Forwarding":** This chapter provides a review of basic network fundamentals and then dives deeper into technical concepts related to how network traffic is forwarded through a router or switch architecture.
- **Chapter 2, "Spanning Tree Protocol":** This chapter explains how switches prevent forwarding loops while allowing for redundant links with the use of Spanning Tree Protocol (STP) and Rapid Spanning Tree Protocol (RSTP).

- **Chapter 3, "Advanced STP Tuning":** This chapter reviews common techniques that are in Cisco Validated Design guides. Topics include root bridge placement and protection.
- **Chapter 4, "Multiple Spanning Tree Protocol":** This chapter completes the section of spanning tree by explaining Multiple Spanning Tree (MST) protocol.
- **Chapter 5, "VLAN Trunks and EtherChannel Bundles":** This chapter covers features such as VTP, DTP, and EtherChannel for switch-to-switch connectivity.
- **Chapter 6, "IP Routing Essentials":** This chapter revisits the fundamentals from Chapter 1 and examines some of the components of the operations of a router. It reinforces the logic of the programming of the Routing Information Base (RIB), reviews differences between common routing protocols, and explains common concepts related to static routes.
- **Chapter 7, "EIGRP":** This chapter explains the underlying mechanics of the EIGRP routing protocol, the path metric calculations, and the failure detection mechanisms and techniques for optimizing the operations of the routing protocol.
- **Chapter 8, "OSPF":** This chapter explains the core concepts of OSPF and the basics in establishing neighborships and exchanging routes with other OSPF routers.
- **Chapter 9, "Advanced OSPF":** This chapter expands on Chapter 8 and explains the functions and features found in larger enterprise networks. By the end of this chapter, you should have a solid understanding of the route advertisement within a multi-area OSPF domain, path selection, and techniques to optimize an OSPF environment.
- **Chapter 10, "OSPFv3":** This chapter explains how the OSPF protocol has changed to accommodate support of IPv6.
- **Chapter 11, "BGP":** This chapter explains the core concepts of BGP and its path attributes. This chapter explains configuration of BGP and advertisement and summarization of IPv4 and IPv6 network prefixes.
- **Chapter 12, "Advanced BGP":** This chapter expands on Chapter 11 and explains BGP's advanced features and concepts, such as BGP multihoming, route filtering, BGP communities, and the logic for identifying the best path for a specific network prefix.
- **Chapter 13, "Multicast":** This chapter describes the fundamental concepts related to multicast and how it operates. It also describes the protocols that are required to understand its operation in more detail, such as Internet Group Messaging Protocol (IGMP), IGMP snooping, Protocol Independent Multicast (PIM) Dense Mode/Sparse Mode, and rendezvous points (RPs).
- **Chapter 14, "QoS":** This chapter describes the different QoS models available: best effort, Integrated Services (IntServ), and Differentiated Services (DiffServ). It also describes tools and mechanisms used to implement QoS such as classification and marking, policing and shaping, and congestion management and avoidance.
- **Chapter 15, "IP Services":** In addition to routing and switching network packets, a router can perform additional functions to enhance the network. This chapter covers time synchronization, virtual gateway technologies, and network addr

- **Chapter 16, "Overlay Tunnels":** This chapter explains Generic Routing Encapsulation (GRE) and IP Security (IPsec) fundamentals and how to configure them. It also explains Locator ID/Separation Protocol (LISP) and Virtual Extensible Local Area Network (VXLAN).
- **Chapter 17, "Wireless Signals and Modulation":** This chapter covers the basic theory behind radio frequency (RF) signals, measuring and comparing the power of RF signals, and basic methods and standards involved in carrying data wirelessly.
- **Chapter 18, "Wireless Infrastructure":** This chapter describes autonomous, cloud-based, centralized, embedded, and Mobility Express wireless architectures. It also explains the process that lightweight APs must go through to discover and bind to a wireless LAN controller. Various AP modes and antennas are also described.
- **Chapter 19, "Understanding Wireless Roaming and Location Services":** This chapter discusses client mobility from the AP and controller perspectives so that you can design and configure a wireless network properly as it grows over time. It also explains how components of a wireless network can be used to compute the physical locations of wireless devices.
- **Chapter 20, "Authenticating Wireless Clients":** This chapter covers several methods you can use to authenticate users and devices in order to secure a wireless network.
- **Chapter 21, "Troubleshooting Wireless Connectivity":** This chapter helps you get some perspective about problems wireless clients may have with their connections, develop a troubleshooting strategy, and become comfortable using a wireless LAN controller as a troubleshooting tool.
- **Chapter 22, "Enterprise Network Architecture":** This chapter provides a high-level overview of the enterprise campus architectures that can be used to scale from a small environment to a large campus-size network.
- **Chapter 23, "Fabric Technologies":** This chapter defines the benefits of Software-Defined Access (SD-Access) over traditional campus networks as well as the components and features of the Cisco SD-Access solution, including the nodes, fabric control plane, and data plane. It also defines the benefits of Software-Defined WAN (SD-WAN) over traditional WANs, as well as the components and features of the Cisco SD-WAN solution, including the orchestration plane, management plane, control plane, and data plane.
- **Chapter 24, "Network Assurance":** This chapter covers some of the tools most commonly used for operations and troubleshooting in the network environment. Cisco DNA Center with Assurance is also covered, to showcase how the tool can improve mean time to innocence (MTTI) and root cause analysis of issues.
- **Chapter 25, "Secure Network Access Control":** This chapter describes a Cisco security framework to protect networks from evolving cybersecurity threats as well as the security components that are part of the framework, such as next-generation firewalls, web security, email security, and much more. It also describes network access control (NAC) technologies such as 802.1x, Web Authentication (WebAuth), MAC Authentication Bypass (MAB), TrustSec, and MACsec.

- **Chapter 26, "Network Device Access Control and Infrastructure Security":** This chapter focuses on how to configure and verify network device access control through local authentication and authorization as well through AAA. It also explains how to configure and verify router security features, such as access control lists (ACLs), control plane policing (CoPP) and zone-based firewalls (ZBFWs), that are used to provide device and infrastructure security.
- **Chapter 27, "Virtualization":** This chapter describes server virtualization technologies such as virtual machines, containers, and virtual switching. It also describes the network functions virtualization (NFV) architecture and Cisco's enterprise NFV solution.
- **Chapter 28, "Foundational Network Programmability Concepts":** This chapter covers current network management methods and tools as well as key network programmability methods. It also covers how to use software application programming interfaces (APIs) and common data formats.
- **Chapter 29, "Introduction to Automation Tools":** This chapter discusses some of the most common automation tools that are available. It covers on-box, agent-based, and agentless tools and examples.
- **Chapter 30, "Final Preparation":** This chapter details a set of tools and a study plan to help you complete your preparation for the CCNP and CCIE Enterprise Core ENCOR 350-401 exam.

Certification Exam Topics and This Book

The questions for each certification exam are a closely guarded secret. However, we do know which topics you must know to *successfully* complete the CCNP and CCIE Enterprise Core ENCOR 350-401 exam. Cisco publishes them as an exam blueprint. Table I-1 lists each exam topic listed in the blueprint along with a reference to the book chapter that covers the topic. These are the same topics you should be proficient in when working with enterprise technologies in the real world.

Table I-1 CCNP and CCIE Enterprise Core ENCOR 350-401 Topics and Chapter References

CCNP and CCIE Enterprise Core ENCOR (350-401) Exam Topic	Chapter(s) in Which Topic Is Covered
1.0 Architecture	
1.1 Explain the different design principles used in an enterprise network	
1.1.a Enterprise network design such as Tier 2, Tier 3, and Fabric Capacity planning	22
1.1.b High availability techniques such as redundancy, FHRP, and SSO	15, 22
1.2 Analyze design principles of a WLAN deployment	
1.2.a Wireless deployment, models (centralized, distributed, controller-less, controller based, cloud, remote branch)	18
1.2.b Location services in a WLAN design	19

CCNP and CCIE Enterprise Core ENCOR (350-401) Exam Topic	Chapter(s) in Which Topic Is Covered
1.3 Differentiate between on-premises and cloud infrastructure deployments	23
1.4 Explain the working principles of the Cisco SD-WAN solution	
1.4.a SD-WAN control and data planes elements	23
1.4.b Traditional WAN and SD-WAN solutions	23
1.5 Explain the working principles of the Cisco SD-Access solution	
1.5.a SD-Access control and data planes elements	23
1.5.b Traditional campus interoperating with SD-Access	23
1.6 Describe concepts of QoS	
1.6.a QoS components	14
1.6.b QoS policy	14
1.7 Differentiate hardware and software switching mechanisms	
1.7.a Process and CEF	1
1.7.b MAC address table and TCAM	1
1.7.c FIB vs. RIB	1
2.0 Virtualization	
2.1 Describe device virtualization technologies	
2.1.a Hypervisor type 1 and 2	27
2.1.b Virtual machine	27
2.1.c Virtual switching	27
2.2 Configure and verify data path virtualization technologies	
2.2.a VRF	6
2.2.b GRE and IPsec tunneling	16
2.3 Describe network virtualization concepts	
2.3.a LISP	16
2.3.b VXLAN	16
3.0 Infrastructure	
3.1 Layer 2	
3.1.a Troubleshoot static and dynamic 802.1q trunking protocols	5
3.1.b Troubleshoot static and dynamic EtherChannels	5
3.1.c Configure and verify common Spanning Tree Protocols (RSTP and MST)	2, 3, 4
3.2 Layer 3	
3.2.a Compare routing concepts of EIGRP and OSPF (advanced distance vector vs. linked state, load balancing, path selection, path operations, metrics)	6, 7, 8, 9

CCNP and CCIE Enterprise Core ENCOR (350-401) Exam Topic	Chapter(s) in Which Topic Is Covered
3.2.b Configure and verify simple OSPF environments, including multiple normal areas, summarization, and filtering (neighbor adjacency, point-to-point and broadcast network types, and passive interface)	8, 9, 10
3.2.c Configure and verify eBGP between directly connected neighbors (best path selection algorithm and neighbor relationships)	11, 12
3.3 Wireless	
3.3.a Describe the main RF signal concepts, such as RSSI, SNR, Tx-power, and wireless client devices capabilities	17
3.3.b Describe AP modes and antenna types	18
3.3.c Describe access point discovery and join process	18
3.3.d Describe the main principles and use cases for Layer 2 and Layer 3 roaming	19
3.3.e Troubleshoot WLAN configuration and wireless client connectivity issues	21
3.4 IP Services	
3.4.a Describe Network Time Protocol (NTP)	15
3.4.b Configure and verify NAT/PAT	15
3.4.c Configure first hop redundancy protocols, such as HSRP and VRRP	15
3.4.d Describe multicast protocols, such as PIM and IGMP v2/v3	13
4.0 Network Assurance	24
4.1 Diagnose network problems using tools such as debugs, conditional debugs, trace route, ping, SNMP, and syslog	24
4.2 Configure and verify device monitoring using syslog for remote logging	24
4.3 Configure and verify NetFlow and Flexible NetFlow	24
4.4 Configure and verify SPAN/RSPAN/ERSPAN	24
4.5 Configure and verify IPSLA	24
4.6 Describe Cisco DNA Center workflows to apply network configuration, monitoring, and management	24
4.7 Configure and verify NETCONF and RESTCONF	28
5.0 Security	
5.1 Configure and verify device access control	26
5.1.a Lines and password protection	26
5.1.b Authentication and authorization using AAA	26

CCNP and CCIE Enterprise Core ENCOR (350-401) Exam Topic	Chapter(s) in Which Topic Is Covered
5.2 Configure and verify infrastructure security features	26
5.2.a ACLs	26
5.2.b CoPP	26
5.3 Describe REST API security	28
5.4 Configure and verify wireless security features	
5.4.a EAP	20
5.4.b WebAuth	20
5.4.c PSK	20
5.5 Describe the components of network security design	25
5.5.a Threat defense	25
5.5.b Endpoint security	25
5.5.c Next-generation firewall	25
5.5.d TrustSec, MACsec	25
5.5.e Network access control with 802.1x, MAB, and WebAuth	20, 25
6.0 Automation	
6.1 Interpret basic Python components and scripts	29
6.2 Construct valid JSON encoded file	28
6.3 Describe the high-level principles and benefits of a data modeling language, such as YANG	28
6.4 Describe APIs for Cisco DNA Center and vManage	28
6.5 Interpret REST API response codes and results in payload using Cisco DNA Center and RESTCONF	28
6.6 Construct EEM applet to automate configuration, troubleshooting, or data collection	29
6.7 Compare agent vs. agentless orchestration tools, such as Chef, Puppet, Ansible, and SaltStack	29

Each version of the exam may emphasize different functions or features, and some topics are rather broad and generalized. The goal of this book is to provide the most comprehensive coverage to ensure that you are well prepared for the exam. Although some chapters might not address specific exam topics, they provide a foundation that is necessary for a clear understanding of important topics.

It is also important to understand that this book is a static reference, whereas the exam topics are dynamic. Cisco can and does change the topics covered on certification exams often.

This exam guide should not be your only reference when preparing for the certification exam. You can find a wealth of information available at Cisco.com that covers each topic in great detail. If you think that you need more detailed information on a specific topic, read the Cisco documentation that focuses on your chosen topic.

Note that as technologies continue to evolve, Cisco reserves the right to change the exam topics without notice. Although you can refer to the list of exam topics in Table I-1, always check Cisco.com to verify the actual list of topics to ensure that you are prepared before taking the exam. You can view the current exam topics on any current Cisco certification exam by visiting the Cisco.com website, hovering over Training & Events, and selecting from the Certifications list. Note also that, if needed, Cisco Press might post additional preparatory content on the web page associated with this book: http://www.ciscopress.com/title/9781587145230. It's a good idea to check the website a couple weeks before taking the exam to be sure that you have up-to-date content.

Figure Credits

Figure 28-2, screenshot of Postman dashboard © 2019 Postman, Inc.

Figure 28-3, screenshot of Postman clear history © 2019 Postman, Inc.

Figure 28-4, screenshot of Postman collection © 2019 Postman, Inc.

Figure 28-5, screenshot of Postman URL bar © 2019 Postman, Inc.

Figure 28-6, screenshot of Postman URL bar © 2019 Postman, Inc.

Figure 28-7, screenshot of Postman URL bar © 2019 Postman, Inc.

Figure 28-8, screenshot of Postman URL bar © 2019 Postman, Inc.

Figure 28-9, screenshot of Postman URL bar © 2019 Postman, Inc.

Figure 28-10, screenshot of Postman URL bar © 2019 Postman, Inc.

Figure 28-11, screenshot of Postman URL bar © 2019 Postman, Inc.

Figure 28-12, screenshot of Postman URL bar © 2019 Postman, Inc.

Figure 28-13, screenshot of Postman URL bar © 2019 Postman, Inc.

Figure 28-14, screenshot of Postman URL bar © 2019 Postman, Inc.

Figure 28-20, screenshot of GitHub main webpage © 2019 GitHub, Inc.

Figure 28-21, screenshot of GitHub ENCORE repository © 2019 GitHub, Inc.

Figure 28-22, screenshot of JSON_Example.txt contents © 2019 GitHub, Inc.

Figure 28-23, screenshot of JSON_Example.txt contents © 2019 GitHub, Inc.

Figure 29-4, screenshot of Chef Architecture © 2019 Chef Software, Inc.

Figure 29-5, screenshot of SaltStack CLI Command © SaltStack, Inc.

Figure 29-6, screenshot of SaltStack CLI Command © SaltStack, Inc.

Figure 29-7, screenshot of SaltStack CLI Command © SaltStack, Inc.

Figure 29-10, screenshot of YAML Lint © YAML Lint

Figure 29-11, screenshot of IExecuting ConfigureInterface © YAML Lint

Figure 29-12, screenshot of Executing EIGRP_Configuration © YAML Lint

Figure 29-14, screenshot of Puppet © 2019 Puppet

This page intentionally left blank

CHAPTER 1

Packet Forwarding

This chapter covers the following subjects:

Network Device Communication: This section explains how switches forward traffic from a Layer 2 perspective and routers forward traffic from a Layer 3 perspective.

Forwarding Architectures: This section examines the mechanisms used in routers and switches to forward network traffic.

This chapter provides a review of basic network fundamentals and then dives deeper into the technical concepts related to how network traffic is forwarded through a router or switch architecture.

"Do I Know This Already?" Quiz

The "Do I Know This Already?" quiz allows you to assess whether you should read the entire chapter. If you miss no more than one of these self-assessment questions, you might want to move ahead to the "Exam Preparation Tasks" section. Table 1-1 lists the major headings in this chapter and the "Do I Know This Already?" quiz questions covering the material in those headings so you can assess your knowledge of these specific areas. The answers to the "Do I Know This Already?" quiz appear in Appendix A, "Answers to the 'Do I Know This Already?' Quiz Questions."

Table 1-1 "Do I Know This Already?" Foundation Topics Section-to-Question Mapping

Foundation Topics Section	Questions
Network Device Communication	1–4
Forwarding Architectures	5–7

1. Forwarding of network traffic from a Layer 2 perspective uses what information?
   - **a.** Source IP address
   - **b.** Destination IP address
   - **c.** Source MAC address
   - **d.** Destination MAC address
   - **e.** Data protocol
2. What type of network device helps reduce the size of a collision domain?
   - **a.** Hub
   - **b.** Switch
   - **c.** Load balancer
   - **d.** Router

3. Forwarding of network traffic from a Layer 3 perspective uses what information?
   a. Source IP address
   b. Destination IP address
   c. Source MAC address
   d. Destination MAC address
   e. Data protocol
4. What type of network device helps reduce the size of a broadcast domain?
   a. Hub
   b. Switch
   c. Load balancer
   d. Router
5. The _________ can be directly correlated to the MAC address table.
   a. Adjacency table
   b. CAM
   c. TCAM
   d. Routing table
6. A ___________ forwarding architecture provides increased port density and forwarding scalability.
   a. Centralized
   b. Clustered
   c. Software
   d. Distributed
7. CEF is composed of which components? (Choose two.)
   a. Routing Information Base
   b. Forwarding Information Base
   c. Label Information Base
   d. Adjacency table
   e. MAC address table

Foundation Topics

Network Device Communication

The primary function of a network is to provide connectivity between devices. There used to be a variety of network protocols that were device specific or preferred; today, almost everything is based on *Transmission Control Protocol/Internet Protocol (TCP/IP)*. It is important to note that TCP/IP is based on the conceptual *Open Systems Interconnection (OSI)* model that is composed of seven layers. Each layer describes a specific function, and a layer can be modified or changed without requiring changes to the layer above or below it.

The OSI model, which provides a structured approach for compatibility between vendors, is illustrated in Figure 1-1.

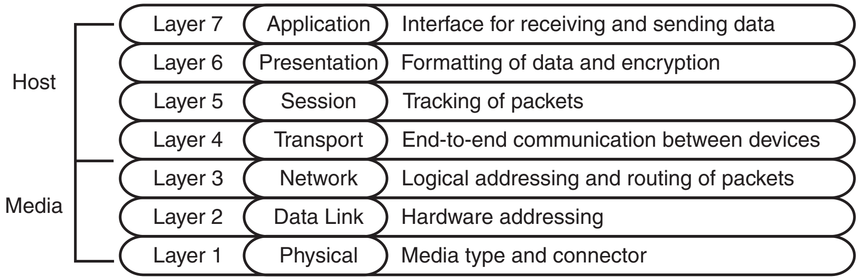


Figure 1-1 *OSI Model*

When you think about the flow of data, most network traffic involves communication of data between applications. The applications generate data at Layer 7, and the device/host sends data down the OSI model. As the data moves down the OSI model, it is encapsulated or modified as needed.

At Layer 3, the device/host decides whether the data needs to be sent to another application on the same device, and it would then start to move the data up the stack. Or, if the data needs to be sent to a different device, the device/host continues processing down the OSI model toward Layer 1. Layer 1 is responsible for transmitting the information on to the media (for example, cable, fiber, radio waves). On the receiving side, data starts at Layer 1, then moves to Layer 2, and so on, until it has moved completely up to Layer 7 and on to the receiving application.

This chapter reinforces concepts related to how a network device forwards traffic from either a Layer 2 or a Layer 3 perspective. The first Layer 2 network devices were bridges or switches, and Layer 3 devices were strictly routers. As technology advanced, the development of faster physical media required the ability to forward packets in hardware through ASICs. As ASIC functionality continued to develop, multilayer switches (MLSs) were invented to forward Layer 2 traffic in hardware as if they were switches; however, they can also perform other functions, such as routing packets, from a Layer 3 perspective.

Layer 2 Forwarding

The second layer of the OSI model, the data link layer, handles addressing beneath the IP protocol stack so that communication is directed between hosts. Network packets include Layer 2 addressing with unique source and destination addresses for segments. Ethernet commonly uses *media access control (MAC)* addresses, and other data link layer protocols such as Frame Relay use an entirely different method of Layer 2 addressing.

The focus of the Enterprise Core exam is on Ethernet and wireless technologies, both of which use MAC addresses for *Layer 2* addressing. This book focuses on the MAC address for Layer 2 forwarding.

Answers to the "Do I Know This Already?" quiz:

1 D **2** B **3** B **4** D **5** B **6** D **7** B, D

NOTE A MAC address is a 48-bit address that is split across six octets and notated in hexadecimal. The first three octets are assigned to a device manufacturer, known as the organizationally unique identifier (OUI), and the manufacturer is responsible for ensuring that the last three octets are unique. A device listens for network traffic that contains its MAC address as the packet's destination MAC address before moving the packet up the OSI stack to Layer 3 for processing.

Network broadcasts with MAC address FF:FF:FF:FF:FF:FF are the exception to the rule and will always be processed by all network devices on the same network segment. Broadcasts are not typically forwarded beyond a Layer 3 boundary.

Collision Domains

The Ethernet protocol first used technologies like Thinnet (10BASE-2) and Thicknet (10BASE-5), which connected all the network devices using the same cable and T connectors. This caused problems when two devices tried to talk at the same time because the transmit cable shared the same segment with other devices, and the communication become garbled if two devices talked at the same time. Ethernet devices use *Carrier Sense Multiple Access/Collision Detect (CSMA/CD)* to ensure that only one device talks at time in a *collision domain*. If a device detects that another device is transmitting data, it delays transmitting packets until the cable is quiet. This means devices can only transmit or receive data at one time (that is, operate at half-duplex).

Key Topic

As more devices are added to a cable, the less efficient the network becomes as devices wait until there is not any communication. All of the devices are in the same collision domain. Network hubs proliferate the problem because they add port density while repeating traffic, thereby increasing the size of the collision domain. Network hubs do not have any intelligence in them to direct network traffic; they simply repeat traffic out of every port.

Network switches enhance scalability and stability in a network through the creation of virtual channels. A switch maintains a table that associates a host's *Media Access Control (MAC)* Ethernet addresses to the port that sourced the network traffic. Instead of flooding all traffic out of every switch port, a switch uses the local *MAC address table* to forward network traffic only to the destination switch port associated with where the destination MAC is attached. This drastically reduces the size of the collision domain between the devices and enables the devices to transmit and receive data at the same time (that is, operate at full duplex).

Figure 1-2 demonstrates the collision domains on a hub versus on a switch. Both of these topologies show the same three PCs, as well as the same cabling. On the left, the PCs are connected to a network hub. Communication between PC-A and PC-B is received by PC-C's NIC, too, because all three devices are in the same collision domain. PC-C must process the frame—in the process consuming resources—and then it discards the packet after determining that the destination MAC address does not belong to it. In addition, PC-C has to wait until the PC-A/PC-B conversation finishes before it can transmit data. On the right, the PCs are connected to a network switch. Communication between PC-A and PC-B are split into two collision domains. The switch can connect the two collision domains by using information from the MAC address table.

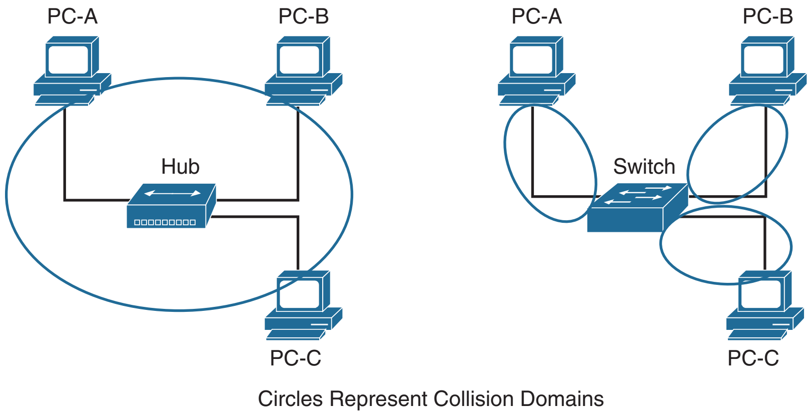
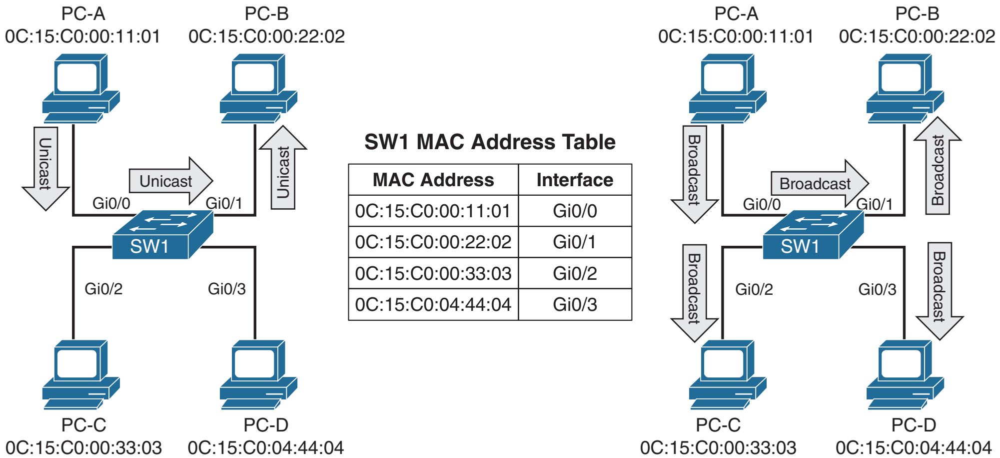


Figure 1-2 *Collision Domains on a Hub Versus a Switch*

When a packet contains a destination MAC address that is not in the switch's MAC address table, the switch forwards the packet out of every switch port. This is known as *unknown unicast flooding* because the destination MAC address is not known.

Broadcast traffic is network traffic intended for every host on the LAN and is forwarded out of every switch port interface. This is disruptive as it diminishes the efficiencies of a network switch compared to those of a hub because it causes communication between network devices to stop due to CSMA/CD. Network broadcasts do not cross Layer 3 boundaries (that is, from one subnet to another subnet). All devices that reside in the same Layer 2 segment are considered to be in the same *broadcast domain*.

Figure 1-3 displays SW1's MAC address table, which correlates the local PCs to the appropriate switch port. In the scenario on the left, PC-A is transmitting unicast traffic to PC-B. SW1 does not transmit data out of the Gi0/2 or Gi0/3 interface (which could potentially disrupt any network transmissions between those PCs). In the scenario on the right, PC-A is transmitting broadcast network traffic out all active switch ports.



MAC Address	Interface
0C:15:C0:00:11:01	Gi0/0
0C:15:C0:00:22:02	Gi0/1
0C:15:C0:00:33:03	Gi0/2
0C:15:C0:04:44:04	Gi0/3

Figure 1-3 *Unicast and Broadcast Traffic Patterns*

NOTE The terms *network device* and *host* are considered interchangeable in this text.

Virtual LANs

Adding a router between LAN segments helps shrink broadcast domains and provides for optimal network communication. Host placement on a LAN segment varies because of network addressing. Poor host network assignment can lead to inefficient use of hardware as some switch ports could be unused.

Key Topic

Virtual LANs (VLANs) provide logical segmentation by creating multiple broadcast domains on the same network switch. VLANs provide higher utilization of switch ports because a port can be associated to the necessary broadcast domain, and multiple broadcast domains can reside on the same switch. Network devices in one VLAN cannot communicate with devices in a different VLAN via traditional Layer 2 or broadcast traffic.

VLANs are defined in the Institute of Electrical and Electronic Engineers (IEEE) 802.1Q standard, which states that 32 bits are added to the packet header in the following fields:

- **Tag protocol identifier (TPID):** This 16-bit is field set to 0x8100 to identify the packet as an 802.1Q packet.
- **Priority code point (PCP):** This 3-bit field indicates a class of service (CoS) as part of Layer 2 quality of service (QoS) between switches.
- **Drop elgible indicator (DEI):** This 1-bit field indicates whether the packet can be dropped when there is bandwidth contention.
- **VLAN identifier (VLAN ID):** This 12-bit field specifies the VLAN associated with a network packet.

Figure 1-4 displays the VLAN packet structure.

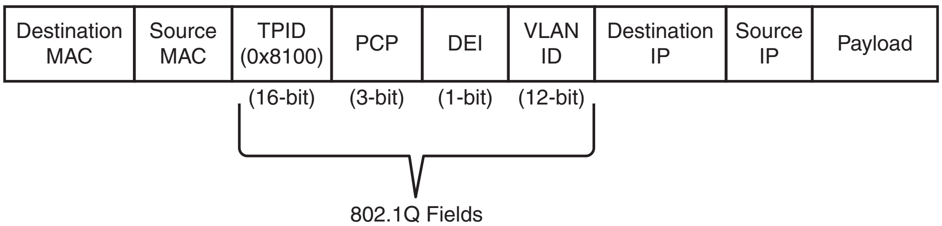


Figure 1-4 *VLAN Packet Structure*

The VLAN identifier has only 12 bits, which provides 4094 unique VLANs. Catalyst switches use the following logic for VLAN identifiers:

- VLAN 0 is reserved for 802.1P traffic and cannot be modified or deleted.
- VLAN 1 is the default VLAN and cannot be modified or deleted.
- VLANs 2 to 1001 are in the normal VLAN range and can be added, deleted, or modified as necessary.

- VLANS 1002 to 1005 are reserved and cannot be deleted.
- VLANs 1006 to 4094 are in the extended VLAN range and can be added, deleted, or modified as necessary.

VLANs are created by using the global configuration command **vlan** *vlan-id*. A friendly name (32 characters) is associated with a VLAN through the VLAN submode configuration command **name** *vlanname*. The VLAN is not created until the command-line interface (CLI) has been moved back to the global configuration context or a different VLAN identifier.

Example 1-1 demonstrates the creation of VLAN 10 (PCs), VLAN 20 (Phones), and VLAN 99 (Guest) on SW1.

Example 1-1 *Creating a VLAN*

```
SW1# configure term
Enter configuration commands, one per line. End with CNTL/Z.
SW1(config)# vlan 10
SW1(config-vlan)# name PCs
SW1(config-vlan)# vlan 20
SW1(config-vlan)# name Phones
SW1(config-vlan)# vlan 99
SW1(config-vlan)# name Guest
```

VLANs and their port assignment are verified with the **show vlan** [{**brief** | **id** *vlan-id* | **name** *vlanname* | **summary**}] command, as demonstrated in Example 1-2. Notice that the output is split into four main sections: VLAN-to-port assignments, system MTU, SPAN sessions, and private VLANs.

Example 1-2 *Viewing VLAN Assignments to Port Mapping*

```
SW1# show vlan
! Traditional and common VLANs will be listed in this section. The ports
! associated to these VLANs are displayed to the right.
VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Gi1/0/1, Gi1/0/2, Gi1/0/3
                                                Gi1/0/4, Gi1/0/5, Gi1/0/6
                                                Gi1/0/10, Gi1/0/11, Gi1/0/17
                                                Gi1/0/18, Gi1/0/19, Gi1/0/20
                                                Gi1/0/21, Gi1/0/22, Gi1/0/23
                                                Gi1/1/1, Gi1/1/2, Te1/1/3
                                                Te1/1/4
10   PCs                              active    Gi1/0/7, Gi1/0/8, Gi1/0/9
                                                Gi1/0/12, Gi1/0/13
20   Phones                           active    Gi1/0/14
99   Guest                            active    Gi1/0/15, Gi1/0/16
```

```
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
!  This section displays the system wide MTU setting for all 1Gbps and faster
!  interface

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------


VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet  100001     1500  -      -      -        -    -        0      0
10   enet  100010     1500  -      -      -        -    -        0      0
20   enet  100020     1500  -      -      -        -    -        0      0
99   enet  100099     1500  -      -      -        -    -        0      0
1002 fddi  101002     1500  -      -      -        -    -        0      0
1003 tr    101003     1500  -      -      -        -    -        0      0
1004 fdnet 101004     1500  -      -      -        ieee -        0      0
1005 trnet 101005     1500  -      -      -        ibm  -        0      0

!  If a Remote SPAN VLAN is configured, it will be displayed in this section.
!  Remote SPAN VLANs are explained in Chapter 24
Remote SPAN VLANs
------------------------------------------------------------------------------

!  If Private VLANs are configured, they will be displayed in this section.
!  Private VLANs are outside of the scope of this book, but more information
!  can be found at http://www.cisco.com
Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
```

The optional **show vlan** keywords provide the following benefits:

- **brief:** Displays only the relevant port-to-VLAN mappings.
- **summary:** Displays a count of VLANS, VLANs participating in VTP, and VLANs that are in the extended VLAN range.
- **id** *vlan-id*: Displays all the output from the original command but filtered to only the VLAN number that is specified.
- **name** *vlanname*: Displays all the output from the original command but filtered to only the VLAN name that is specified.

Example 1-3 shows the use of the optional keywords. Notice that the output from the optional keywords **id** *vlan-id* is the same as the output from **name** *vlanname*.

Example 1-3 *Using the Optional* **show vlan** *Keywords*

```
SW1# show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Gi1/0/1, Gi1/0/2, Gi1/0/3
                                                Gi1/0/4, Gi1/0/5, Gi1/0/6
                                                Gi1/0/10, Gi1/0/11, Gi1/0/17
                                                Gi1/0/18, Gi1/0/19, Gi1/0/20
                                                Gi1/0/21, Gi1/0/22, Gi1/0/23
                                                Gi1/1/1, Gi1/1/2, Te1/1/3
                                                Te1/1/4
10   PCs                              active    Gi1/0/7, Gi1/0/8, Gi1/0/9
                                                Gi1/0/12, Gi1/0/13
20   Phones                           active    Gi1/0/14
99   Guest                            active    Gi1/0/15, Gi1/0/16
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
```

```
SW1# show vlan summary
Number of existing VLANs                 : 8
 Number of existing VTP VLANs            : 8
 Number of existing extended VLANS       : 0
```

```
SW1# show vlan id 99

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
99   Guest                            active    Gi1/0/15, Gi1/0/16

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
99   enet  100099     1500  -      -      -        -    -        0      0

Remote SPAN VLAN
----------------
Disabled

Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
```

```
SW1# show vlan name Guest

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
99   Guest                            active    Gi1/0/15, Gi1/0/16

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
99   enet  100099     1500  -      -      -        -    -        0      0

Remote SPAN VLAN
----------------
Disabled

Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
```

Key Topic

Access Ports

Access ports are the fundamental building blocks of a managed switch. An access port is assigned to only one VLAN. It carries traffic from the specified VLAN to the device connected to it or from the device to other devices on the same VLAN on that switch. The 802.1Q tags are not included on packets transmitted or received on access ports.

Catalyst switches place switch ports as Layer 2 access ports for VLAN 1 by default. The port can be manually configured as an access port with the command **switchport mode access**. A specific VLAN is associated to the port with the command **switchport access {vlan** *vlan-id* | **name** *vlanname***}**. The ability to set VLANs to an access port by name was recently added with newer code but is stored in numeric form in the configuration.

Example 1-4 demonstrates the configuration of switch ports Gi1/0/15 and Gi1/0/16 as access ports in VLAN 99 for Guests. Notice that the final configuration is stored as numbers for both ports, even though different commands are issued.

Example 1-4 *Configuring an Access Port*

```
SW1# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
SW1(config)# vlan 99
SW1(config-vlan)# name Guests
SW1(config-vlan)# interface gi1/0/15
SW1(config-if)# switchport mode access
SW1(config-if)# switchport access vlan 99
SW1(config-if)# interface gi1/0/16
SW1(config-if)# switchport mode access
SW1(config-if)# switchport access vlan name Guest
```

```
SW1# show running-config | begin interface GigabitEthernet1/0/15
interface GigabitEthernet1/0/15
 switchport access vlan 99
 switchport mode access
!
interface GigabitEthernet1/0/16
 switchport access vlan 99
 switchport mode access
```

Key Topic

Trunk Ports

Trunk ports can carry multiple VLANs. Trunk ports are typically used when multiple VLANs need connectivity between a switch and another switch, router, or firewall and use only one port. Upon receipt of the packet on the remote trunk link, the headers are examined, traffic is associated to the proper VLAN, then the 802.1Q headers are removed, and traffic is forwarded to the next port, based on MAC address for that VLAN.

NOTE Thanks to the introduction of virtualization, some servers run a hypervisor for the operating system and contain a virtualized switch with different VLANs. These servers provide connectivity via a trunk port as well.

Trunk ports are statically defined on Catalyst switches with the interface command **switch-port mode trunk**. Example 1-5 displays Gi1/0/2 and Gi1/0/3 being converted to a trunk port.

Example 1-5 *Configuring a Trunk Port*

```
SW1# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
SW1(config)# interface gi1/0/2
SW1(config-if)# switchport mode trunk
SW1(config-if)# interface gi1/0/3
SW1(config-if)# switchport mode trunk
```

The command **show interfaces trunk** provides a lot of valuable information in several sections for troubleshooting connectivity between network devices:

- The first section lists all the interfaces that are trunk ports, the status, the association to an EtherChannel, and whether a VLAN is a native VLAN. Native VLANs are explained in the next section. EtherChannel is explained in Chapter 5, “VLAN Trunks and EtherChannel Bundles.”
- The second section of the output displays the list of VLANs that are allowed on the trunk port. Traffic can be minimized on trunk ports to restrict VLANs to specific switches, thereby restricting broadcast traffic, too. Other use cases involve a form of load balancing between network links where select VLANs are allowed on one trunk link, while a different set of VLANs are allowed on a different trunk port.

- The third section displays the VLANs that are in a forwarding state on the switch. Ports that are in blocking state are not listed in this section.

Example 1-6 demonstrates the use of the **show interfaces trunk** command with an explanation of each section.

Example 1-6 *Verifying Trunk Port Status*

```
SW1# show interfaces trunk
! Section 1 displays the native VLAN associated on this port, the status and
! if the port is associated to a EtherChannel

Port        Mode             Encapsulation  Status        Native vlan
Gi1/0/2     on               802.1q         trunking      1
Gi1/0/3     on               802.1q         trunking      1

! Section 2 displays all of the VLANs that are allowed to be transmitted across
! the trunk ports

Port        Vlans allowed on trunk
Gi1/0/2     1-4094
Gi1/0/3     1-4094

Port        Vlans allowed and active in management domain
Gi1/0/2     1,10,20,99
Gi1/0/3     1,10,20,99

! Section 3 displays all of the VLANs that are allowed across the trunk and are
! in a spanning tree forwarding state

Port        Vlans in spanning tree forwarding state and not pruned
Gi1/0/2     1,10,20,99
Gi1/0/3     1,10,20,99
```

Native VLANs

In the 802.1Q standard, any traffic that is advertised or received on a trunk port without the 802.1Q VLAN tag is associated to the *native VLAN*. The default native VLAN is VLAN 1. This means that when a switch has two access ports configured as access ports and associated to VLAN 10—that is, a host attached to a trunk port with a native VLAN set to 10—the host could talk to the devices connected to the access ports.

The native VLAN should match on both trunk ports, or traffic can change VLANs unintentionally. While connectivity between hosts is feasible (assuming that they are on the different VLAN numbers), this causes confusion for most network engineers and is not a best practice.

A native VLAN is a port-specific configuration and is changed with the interface command **switchport trunk native vlan** *vlan-id*.

NOTE All switch control plane traffic is advertised using VLAN 1. The Cisco security hardening guidelines recommend changing the native VLAN to something other than VLAN 1. More specifically, it should be set to a VLAN that is not used at all (that is, has no hosts attached to it).

Allowed VLANs

As stated earlier, VLANs can be restricted from certain trunk ports as a method of traffic engineering. This can cause problems if traffic between two hosts is expected to traverse a trunk link and the VLAN is not allowed to traverse that trunk port. The interface command **switchport trunk allowed vlan** *vlan-ids* specifies the VLANs that are allowed to traverse the link. Example 1-7 displays sample a configuration for limiting the VLANs that can cross the Gi1/0/2 trunk port for VLANs 1, 10, 20, and 99.

Example 1-7 *Viewing the VLANs That Are Allowed on a Trunk Link*

```
SW1# show run interface gi1/0/1
interface GigabitEthernet1/0/1
 switchport trunk allowed vlan 1,10,20,99
 switchport mode trunk
```

The full command syntax **switchport trunk allowed** {*vlan-ids* | **all** | **none** | **add** *vlan-ids* | **remove** *vlan-ids* | **except** *vlan-ids*} provides a lot of power in a single command. The optional keyword **all** allows for all VLANs, while **none** removes all VLANs from the trunk link. The **add** keyword adds additional VLANs to those already listed, and the **remove** keyword removes the specified VLAN from the VLANs already identified for that trunk link.

NOTE When scripting configuration changes, it is best to use the **add** and **remove** keywords as they are more prescriptive. A common mistake is to use the **switchport trunk allowed vlan** *vlan-ids* command to list only the VLAN that is being added. This results in the current list being overwritten, causing traffic loss for the VLANs that were omitted.

Layer 2 Diagnostic Commands

The information in the “Layer 2 Forwarding” section, earlier in this chapter, provides a brief primer on the operations of a switch. The following sections provide some common diagnostic commands that are used in the daily administration, operation, and troubleshooting of a network.

MAC Address Table

The MAC address table is responsible for identifying the switch ports and VLANs with which a device is associated. A switch builds the MAC address table by examining the source MAC address for traffic that it receives. This information is then maintained to shrink the collision domain (point-to-point communication between devices and switches) by reducing the amount of unknown unicast flooding.

The MAC address table is displayed with the command **show mac address-table** [**address** *mac-address* | **dynamic** | **vlan** *vlan-id*]. The optional keywords with this command provide the following benefits:

- **address** *mac-address*: Displays entries that match the explicit MAC address. This command could be beneficial on switches with hundreds of ports.
- **dynamic:** Displays entries that are dynamically learned and are not statically set or burned in on the switch.
- **vlan** *vlan-id*: Displays entries that matches the specified VLAN.

Example 1-8 shows the MAC address table on a Catalyst. The command in this example displays the VLAN, MAC address, type, and port that the MAC address is connected to. Notice that port Gi1/0/3 has multiple entries, which indicates that this port is connected to a switch.

Example 1-8 *Viewing the MAC Address Table*

```
SW1# show mac address-table dynamic
          Mac Address Table
-------------------------------------------


Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
   1    0081.c4ff.8b01    DYNAMIC     Gi1/0/2
   1    189c.5d11.9981    DYNAMIC     Gi1/0/3
   1    189c.5d11.99c7    DYNAMIC     Gi1/0/3
   1    7070.8bcf.f828    DYNAMIC     Gi1/0/17
   1    70df.2f22.b882    DYNAMIC     Gi1/0/2
   1    70df.2f22.b883    DYNAMIC     Gi1/0/3
   1    bc67.1c5c.9304    DYNAMIC     Gi1/0/2
   1    bc67.1c5c.9347    DYNAMIC     Gi1/0/3
  99    189c.5d11.9981    DYNAMIC     Gi1/0/3
  99    7069.5ad4.c228    DYNAMIC     Gi1/0/15
  10    0087.31ba.3980    DYNAMIC     Gi1/0/9
  10    0087.31ba.3981    DYNAMIC     Gi1/0/9
  10    189c.5d11.9981    DYNAMIC     Gi1/0/3
  10    3462.8800.6921    DYNAMIC     Gi1/0/8
  10    5067.ae2f.6480    DYNAMIC     Gi1/0/7
  10    7069.5ad4.c220    DYNAMIC     Gi1/0/13
  10    e8ed.f3aa.7b98    DYNAMIC     Gi1/0/12
  20    189c.5d11.9981    DYNAMIC     Gi1/0/3
  20    7069.5ad4.c221    DYNAMIC     Gi1/0/14
Total Mac Addresses for this criterion: 19
```

NOTE Troubleshooting network traffic problems from a Layer 2 perspective involves locating the source and destination device and port; this can be done by examining the MAC address table. If multiple MAC addresses appear on the same port, you know that a switch, hub, or server with a virtual switch is connected to that switch port. Connecting to the switch may be required to identify the port that a specific network device is attached to.

Some older technologies (such as load balancing) require a static MAC address entry in the MAC address table to prevent unknown unicast flooding. The command **mac address-table static mac-address vlan** *vlan-id* {**drop** | **interface** *interface-id*} adds a manual entry with the ability to associate it to a specific switch port or to drop traffic upon receipt.

The command **clear mac address-table dynamic** [{**address** *mac-address* | **interface** *interface-id* | **vlan** *vlan-id*}] flushes the MAC address table for the entire switch. Using the optional keywords can flush the MAC address table for a specific MAC address, switch port, or interface.

Key Topic

The MAC address table resides in *content addressable memory (CAM)*. The CAM uses high-speed memory that is faster than typical computer RAM due to its search techniques. The CAM table provides a binary result for any query of 0 for true or 1 for false. The CAM is used with other functions to analyze and forward packets very quickly. Switches are built with large CAM to accommodate all the Layer 2 hosts for which they must maintain forwarding tables.

Switch Port Status

Examining the configuration for a switch port can be useful; however, some commands stored elsewhere in the configuration preempt the configuration set on the interface. The command **show interfaces** *interface-id* **switchport** provides all the relevant information for a switch port's status. The command **show interfaces switchport** displays the same information for all ports on the switch.

Example 1-9 shows the output from the **show interfaces gi1/0/5 switchport** command on SW1. The key fields to examine at this time are the switch port state, operational mode, and access mode VLAN.

Example 1-9 *Viewing the Switch Port Status*

```
SW1# show interfaces gi1/0/5 switchport
Name: Gi1/0/5
! The following line indicates if the port is shut or no shut
Switchport: Enabled
Administrative Mode: dynamic auto
! The following line indicates if the port is acting as static access port, trunk
! port, or if is down due to carrier detection (i.e. link down)
Operational Mode: down
Administrative Trunking Encapsulation: dot1q
Negotiation of Trunking: On
! The following line displays the VLAN assigned to the access port
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
```

```
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none
```

Interface Status

The command **show interface status** is another useful command for viewing the status of switch ports in a very condensed and simplified manner. Example 1-10 demonstrates the use of this command and includes following fields in the output:

- **Port:** Displays the interface ID or port channel.
- **Name:** Displays the configured interface description.
- **Status:** Displays *connected* for links where a connection was detected and established to bring up the link. Displays *notconnect* for when a link is not detected and *err-disabled* when an error has been detected and the switch has disabled the ability to forward traffic out of that port.
- **VLAN:** Displays the VLAN number assigned for access ports. Trunk links appear as *trunk*, and ports configured as Layer 3 interfaces display *routed*.
- **Duplex:** Displays the duplex of the port. If the duplex auto-negotiated, it is prefixed by *a-*.
- **Speed:** Displays the speed of the port. If the port speed was auto-negotiated, it is prefixed by *a-*.
- **Type:** Displays the type of interface for the switch port. If it is a fixed RJ-45 copper port, it includes TX in the description (for example, 10/100/1000BASE-TX). Small form-factor pluggable (SFP)–based ports are listed with the SFP model if there is a driver for it in the software; otherwise, it says *unknown*.

Example 1-10 *Viewing Overall Interface Status*

```
SW1# show interface status

Port      Name               Status       Vlan       Duplex  Speed Type
Gi1/0/1                      notconnect   1            auto   auto 10/100/1000BaseTX
Gi1/0/2   SW-2 Gi1/0/1       connected    trunk      a-full a-1000 10/100/1000BaseTX
Gi1/0/3   SW-3 Gi1/0/1       connected    trunk      a-full a-1000 10/100/1000BaseTX
Gi1/0/4                      notconnect   1            auto   auto 10/100/1000BaseTX
Gi1/0/5                      notconnect   1            auto   auto 10/100/1000BaseTX
Gi1/0/6                      notconnect   1            auto   auto 10/100/1000BaseTX
Gi1/0/7   Cube13.C           connected    10         a-full a-1000 10/100/1000BaseTX
Gi1/0/8   Cube11.F           connected    10         a-full a-1000 10/100/1000BaseTX
Gi1/0/9   Cube10.A           connected    10         a-full  a-100 10/100/1000BaseTX
Gi1/0/10                     notconnect   1            auto   auto 10/100/1000BaseTX
Gi1/0/11                     notconnect   1            auto   auto 10/100/1000BaseTX
Gi1/0/12  Cube14.D Phone     connected    10         a-full a-1000 10/100/1000BaseTX
Gi1/0/13  R1-G0/0/0          connected    10         a-full a-1000 10/100/1000BaseTX
Gi1/0/14  R2-G0/0/1          connected    20         a-full a-1000 10/100/1000BaseTX
Gi1/0/15  R3-G0/1/0          connected    99         a-full a-1000 10/100/1000BaseTX
Gi1/0/16  R4-G0/1/1          connected    99         a-full a-1000 10/100/1000BaseTX
Gi1/0/17                     connected    1          a-full a-1000 10/100/1000BaseTX
Gi1/0/18                     notconnect   1            auto   auto 10/100/1000BaseTX
Gi1/0/19                     notconnect   1            auto   auto 10/100/1000BaseTX
Gi1/0/20                     notconnect   1            auto   auto 10/100/1000BaseTX
Gi1/0/21                     notconnect   1            auto   auto 10/100/1000BaseTX
Gi1/0/22                     notconnect   1            auto   auto 10/100/1000BaseTX
Gi1/0/23                     notconnect   routed       auto   auto 10/100/1000BaseTX
Gi1/0/24                     disabled     4011         auto   auto 10/100/1000BaseTX
Te1/1/1                      notconnect   1            full    10G SFP-10GBase-SR
Te1/1/2                      notconnect   1            auto   auto unknown
```

Layer 3 Forwarding

Now that we have looked at the mechanisms of a switch and how it forwards Layer 2 traffic, let's review the process for forwarding a packet from a Layer 3 perspective. Recall that all traffic starts at Layer 7 and works its way down to Layer 1, so some of the *Layer 3 forwarding* logic occurs before Layer 2 forwarding. There are two main methodologies for Layer 3 forwarding:

- Forwarding traffic to devices on the same subnet
- Forwarding traffic to devices on a different subnet

The following sections explain these two methodologies.

Local Network Forwarding

Two devices that reside on the same subnet communicate locally. As the data is encapsulated with its IP address, the device detects that the destination is on the same network. However, the device still needs to encapsulate the Layer 2 information (that is, the source and destination MAC addresses) to the packet. It knows its own MAC address but does not initially know the destination's MAC address.

Key Topic

The *Address Resolution Protocol (ARP)* table provides a method of mapping Layer 3 IP addresses to Layer 2 MAC addresses by storing the IP address of a host and its corresponding MAC address. The device then uses the ARP table to add the appropriate Layer 2 headers to the data packet before sending it down to Layer 2 for processing and forwarding.

For example, an IP host that needs to perform address resolution for another IP host connected by Ethernet can send an ARP request using the LAN broadcast address, and it then waits for an ARP reply from the IP host. The ARP reply includes the required Layer 2 physical MAC address information.

The ARP table contains entries for remote devices that the host has communicated with recently and that are on the same IP network segment. It does not contain entries for devices on a remote network but does contain the ARP entry for the IP address of the next hop to reach the remote network. If communication has not occurred with a host after a length of time, the entry becomes stale and is removed from the local ARP table.

If an entry does not exist in the local ARP table, the device broadcasts an ARP request to the entire Layer 2 switching segment. The ARP request strictly asks that whoever owns the IP address in the ARP request reply. All hosts in the Layer 2 segment receive the response, but only the device with the matching IP address should respond to the request.

The response is unicast and includes the MAC and IP addresses of the requestor. The device then updates its local ARP table upon receipt of the ARP reply, adds the appropriate Layer 2 headers, and sends the original data packet down to Layer 2 for processing and forwarding.

NOTE The ARP table can be viewed with the command **show ip arp** [*mac-address* | *ip-address* | **vlan** *vlan-id* | *interface-id*]. The optional keywords make it possible to filter the information.

Packet Routing

Packets must be routed when two devices are on different networks. As the data is encapsulated with its IP address, a device detects that its destination is on a different network and must be routed. The device checks its local routing table to identify its next-hop IP address, which may be learned in one of several ways:

- From a static route entry, it can get the destination network, subnet mask, and next-hop IP address.
- A default-gateway is a simplified static default route that just asks for the local next-hop IP address for all network traffic.
- Routes can be learned from routing protocols.

Key Topic

The source device must add the appropriate Layer 2 headers (source and destination MAC addresses), but the destination MAC address is needed for the next-hop IP address. The device looks for the next-hop IP addresses entry in the ARP table and uses the MAC address from the next-hop IP address's entry as the destination MAC address. The next step is to send the data packet down to Layer 2 for processing and forwarding.

The next router receives the packet based on the destination MAC address, analyzes the destination IP address, locates the appropriate network entry in its routing table, identifies the outbound interface, and then finds the MAC address for the destination device (or the MAC address for the next-hop address if it needs to be routed further). The router then modifies the source MAC address to the MAC address of the router's outbound interface and modifies the destination MAC address to the MAC address for the destination device (or next-hop router).

Figure 1-5 illustrates the concept, with PC-A sending a packet to PC-B through an Ethernet connection to R1. PC-A sends the packet to R1's MAC address, 00:C1:5C:00:00:A1. R1 receives the packet, removes the Layer 2 information, and looks for a route to the 192.168.2.2 address. R1 identifies that connectivity to the 192.168.2.2 IP address is through Gigabit Ethernet 0/1. R1 adds the Layer 2 source address by using its Gigabit Ethernet 0/1 MAC address 00:C1:5C:00:00:B1 and the destination address 00:00:00:BB:BB:BB for PC-B.

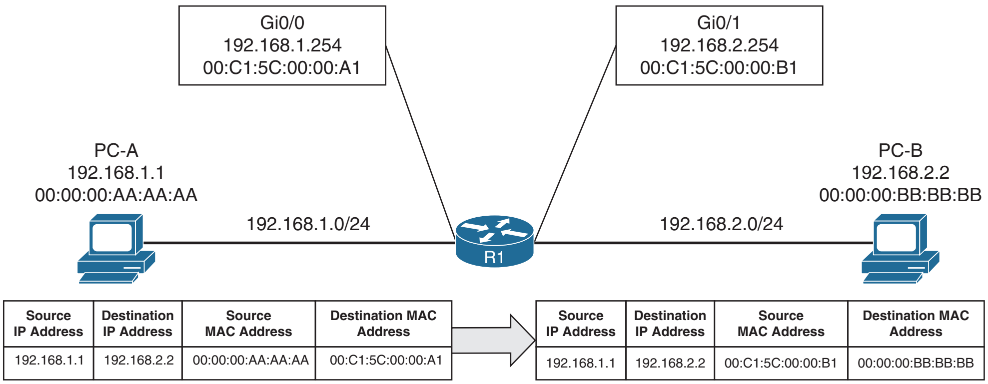


Figure 1-5 *Layer 2 Addressing Rewrite*

NOTE This process continues on and on as needed to get the packet from the source device to the destination device.

IP Address Assignment

TCP/IP has become the standard protocol for most networks. Initially it was used with IPv4 and 32-bit network addresses. The number of devices using public IP addresses has increased at an exponential rate and depleted the number of publicly available IP addresses. To deal with the increase in the number of addresses, a second standard, called IPv6, was developed in 1998; it provides 128 bits for addressing. Technologies and mechanisms have been created to allow IPv4 and IPv6 networks to communicate with each other. With either version, an IP address must be assigned to an interface for a router or multilayer switch to route packets.

Key Topic

IPv4 addresses are assigned with the interface configuration command **ip address** *ip-address subnet-mask*. An interface with a configured IP address and that is in an *up* state injects the associated network into the router's routing table (*Routing Information Base [RIB]*). Connected networks or routes have an *administrative distance (AD)* of zero. It is not possible for any other routing protocol to preempt a connected route in the RIB.

It is possible to attach multiple IPv4 networks to the same interface by attaching a secondary IPv4 address to the same interface with the command **ip address** *ip-address subnet-mask* **secondary**.

IPv6 addresses are assigned with the interface configuration command **ipv6 address** *ipv6-address/prefix-length*. This command can be repeated multiple times to add multiple IPv6 addresses to the same interface.

Example 1-11 demonstrates the configuration of IP addresses on routed interfaces. A routed interface is basically any interface on a router. Notice that a second IPv4 address requires the use of the **secondary** keyword; the **ipv6 address** command can be used multiple times to configure multiple IPv6 addresses.

Example 1-11 *Assigning IP Addresses to Routed Interfaces*

```
R1# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)# interface gi0/0/0
R1(config-if)# ip address 10.10.10.254 255.255
R1(config-if)# ip address 172.16.10.254 255.255.255.0 secondary
R1(config-if)# ipv6 address 2001:db8:10::254/64
R1(config-if)# ipv6 address 2001:DB8:10:172::254/64
R1(config-if)# interface gi0/0/1
R1(config-if)# ip address 10.20.20.254 255.255.255.0
R1(config-if)# ip address 172.16.20.254 255.255.255.0 secondary
R1(config-if)# ipv6 address 2001:db8:20::254/64
R1(config-if)# ipv6 address 2001:db8:20:172::254/64
```

Routed Subinterfaces

In the past, there might have been times when multiple VLANs on a switch required routing, and there were not enough physical router ports to accommodate all those VLANs. It is possible to overcome this issue by configuring the switch's interface as a trunk port and creating logical subinterfaces on a router. A subinterface is created by appending a period and a numeric value after the period. Then the VLAN needs to be associated with the subinterface with the command **encapsulation dot1q** *vlan-id*.

Example 1-12 demonstrates the configuration of two subinterfaces on R2. The subinterface number does not have to match the VLAN ID, but if it does, it helps with operational support.

Example 1-12 *Configuring Routed Subinterfaces*

```
R2# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R2(config-if)# int g0/0/1.10
R2(config-subif)# encapsulation dot1Q 10
R2(config-subif)# ip address 10.10.10.2 255.255.255.0
R2(config-subif)# ipv6 address 2001:db8:10::2/64
R2(config-subif)# int g0/0/1.99
R2(config-subif)# encapsulation dot1Q 99
R2(config-subif)# ip address 10.20.20.2 255.255.255.0
R2(config-subif)# ipv6 address 2001:db8:20::2/64
```

Switched Virtual Interfaces

With Catalyst switches it is possible to assign an IP address to a *switched virtual interface (SVI)*, also known as a *VLAN interface*. An SVI is configured by defining the VLAN on the switch and then defining the VLAN interface with the command **interface vlan** *vlan-id*. The switch must have an interface associated to that VLAN in an *up* state for the SVI to be in an *up* state. If the switch is a multilayer switch, the SVIs can be used for routing packets between VLANs without the need of an external router.

Example 1-13 demonstrates the configuration of the SVI for VLANs 10 and 99.

Example 1-13 *Creating a Switched Virtual Interface (SVI)*

```
SW1# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
SW1(config)# interface Vlan 10
SW1(config-if)# ip address 10.10.10.1 255.255.255.0
SW1(config-if)# ipv6 address 2001:db8:10::1/64
SW1(config-if)# no shutdown
SW1(config-if)# interface vlan 99
SW1(config-if)# ip address 10.99.99.1 255.255.255.0
SW1(config-if)# ipv6 address 2001:db8:99::1/64
SW1(config-if)# no shutdown
```

Routed Switch Ports

Some network designs include a point-to-point link between switches for routing. For example, when a switch needs to connect to a router, some network engineers would build out a transit VLAN (for example, VLAN 2001), associate the port connecting to the router to VLAN 2001, and then build an SVI for VLAN 2001. There is always the potential that VLAN 2001 could exist elsewhere in the Layer 2 realm or that spanning tree could impact the topology.

Instead, the multilayer switch port can be converted from a Layer 2 switch port to a routed switch port with the interface configuration command **no switchport**. Then the IP address can be assigned to it. Example 1-14 demonstrates port Gi1/0/14 being converted from a Layer 2 switch port to a routed switch port and then having an IP address assign

Example 1-14 *Configuring a Routed Switch Port*

```
SW1# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
SW1(config)# int gi1/0/14
SW1(config-if)# no switchport
SW1(config-if)# ip address 10.20.20.1 255.255.255.0
SW1(config-if)# ipv6 address 2001:db8:20::1/64
SW1(config-if)# no shutdown
```

Verification of IP Addresses

IPv4 addresses can be viewed with the command **show ip interface [brief** | *interface-id* | **vlan** *vlan-id***]**. This command's output contains a lot of useful information, such as MTU, DHCP relay, ACLs, and the primary IP address. The optional **brief** keyword displays the output in a condensed format. However, on devices with large port counts, using the CLI parser and adding an additional **| exclude** field (for example, **unassigned**) yields a streamlined view of interfaces that are configured with IP addresses.

Example 1-15 shows the **show ip interface brief** command used with and without the CLI parser. Notice the drastic reduction in unnecessary data that is presented.

Example 1-15 *Viewing Device IPv4 Addresses*

```
SW1# show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
Vlan1                  unassigned      YES manual up                    up
Vlan10                 10.10.10.1      YES manual up                    up
Vlan99                 10.99.99.1      YES manual up                    up
GigabitEthernet0/0     unassigned      YES unset  down                  down
GigabitEthernet1/0/1   unassigned      YES unset  down                  down
GigabitEthernet1/0/2   unassigned      YES unset  up                    up
GigabitEthernet1/0/3   unassigned      YES unset  up                    up
GigabitEthernet1/0/4   unassigned      YES unset  down                  down
GigabitEthernet1/0/5   unassigned      YES unset  down                  down
GigabitEthernet1/0/6   unassigned      YES unset  down                  down
GigabitEthernet1/0/7   unassigned      YES unset  up                    up
GigabitEthernet1/0/8   unassigned      YES unset  up                    up
GigabitEthernet1/0/9   unassigned      YES unset  up                    up
GigabitEthernet1/0/10  unassigned      YES unset  down                  down
GigabitEthernet1/0/11  unassigned      YES unset  down                  down
GigabitEthernet1/0/12  unassigned      YES unset  down                  down
GigabitEthernet1/0/13  unassigned      YES unset  up                    up
GigabitEthernet1/0/14  10.20.20.1      YES manual up                    up
GigabitEthernet1/0/15  unassigned      YES unset  up                    up
GigabitEthernet1/0/16  unassigned      YES unset  up                    up
GigabitEthernet1/0/17  unassigned      YES unset  down                  down
```

```
SW1# show ip interface brief | exclude unassigned
Interface              IP-Address      OK? Method Status                Protocol
Vlan10                 10.10.10.1      YES manual up                    up
Vlan99                 10.99.99.1      YES manual up                    up
GigabitEthernet1/0/14  10.20.20.1      YES manual up                    up
GigabitEthernet1/0/23  192.168.1.1     YES manual down                  down
```

The same information can be viewed for IPv6 addresses with the command **show ipv6 interface** [**brief** | *interface-id* | **vlan** *vlan-id*]. Just as with IPv4 addresses, a CLI parser can be used to reduce the information to what is relevant, as demonstrated in Example 1-16.

Example 1-16 *Viewing Device IPv6 Addresses*

```
SW1# show ipv6 interface brief
! Output omitted for brevity
Vlan1                  [up/up]
    FE80::262:ECFF:FE9D:C547
    2001:1::1
Vlan10                 [up/up]
    FE80::262:ECFF:FE9D:C546
    2001:DB8:10::1
Vlan99                 [up/up]
    FE80::262:ECFF:FE9D:C55D
    2001:DB8:99::1
GigabitEthernet0/0     [down/down]
    unassigned
GigabitEthernet1/0/1   [down/down]
    unassigned
GigabitEthernet1/0/2   [up/up]
    unassigned
GigabitEthernet1/0/3   [up/up]
    unassigned
GigabitEthernet1/0/4   [down/down]
    unassigned
GigabitEthernet1/0/5   [down/down]
    Unassigned
```

```
SW1# show ipv6 interface brief | exclude unassigned|GigabitEthernet
Vlan1                  [up/up]
    FE80::262:ECFF:FE9D:C547
    2001:1::1
Vlan10                 [up/up]
    FE80::262:ECFF:FE9D:C546
    2001:DB8:10::1
Vlan99                 [up/up]
    FE80::262:ECFF:FE9D:C55D
    2001:DB8:99::1
```

Forwarding Architectures

The first Cisco routers would receive a packet, remove the Layer 2 information, and verify that the route existed for the destination IP address. If a matching route could not be found, the packet was dropped. If a matching route was found, the router would identify and add new Layer 2 header information to the packet.

Advancements in technologies have streamlined the process so that routers do not remove and add the Layer 2 addressing but simply rewrite the addresses. IP packet switching or IP packet forwarding is a faster process for receiving an IP packet on an input interface and making a decision about whether to forward the packet to an output interface or drop it. This process is simple and streamlined so that a router can forward large numbers of packets.

When the first Cisco routers were developed, they used a mechanism called process switching to switch the packets through the routers. As network devices evolved, Cisco created *fast switching* and Cisco Express Forwarding (CEF) to optimize the switching process for the routers to be able to handle larger packet volumes.

Key Topic

Process Switching

Process switching, also referred to as *software switching* or *slow path*, is a switching mechanism in which the general-purpose CPU on a router is in charge of packet switching. In IOS, the ip_input process runs on the general-purpose CPU for processing incoming IP packets. Process switching is the fallback for CEF because it is dedicated to processing punted IP packets when they cannot be switched by CEF.

The types of packets that require software handling include the following:

- Packets sourced or destined to the router (using control traffic or routing protocols)
- Packets that are too complex for the hardware to handle (that is, IP packets with IP options)
- Packets that require extra information that is not currently known (for example, ARP)

NOTE Software switching is significantly slower than switching done in hardware. The NetIO process is designed to handle a very small percentage of traffic handled by the system. Packets are hardware switched whenever possible.

Figure 1-6 illustrates how a packet that cannot be CEF switched is punted to the CPU for processing. The *ip_input* process consults the routing table and ARP table to obtain the next-hop router's IP address, outgoing interface, and MAC address. It then overwrites the destination MAC address of the packet with the next-hop router's MAC address, overwrites the source MAC address with the MAC address of the outgoing Layer 3 interface, decrements the IP time-to-live (TTL) field, recomputes the IP header checksum, and finally delivers the packet to the next-hop router.

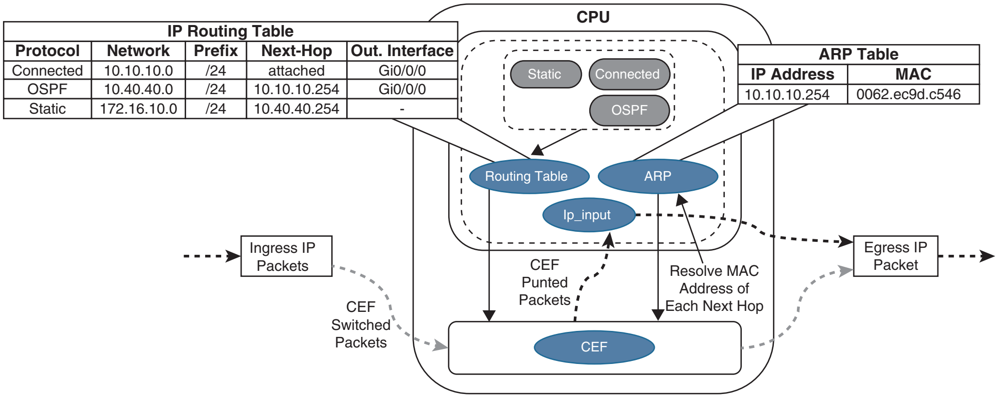


Figure 1-6 *Process Switching*

The routing table, also known as the *Routing Information Base (RIB)*, is built from information obtained from dynamic routing protocols and directly connected and static routes. The ARP table is built from information obtained from the ARP protocol.

Key Topic

Cisco Express Forwarding

Cisco Express Forwarding (CEF) is a Cisco proprietary switching mechanism developed to keep up with the demands of evolving network infrastructures. It has been the default switching mechanism on most Cisco platforms that do all their packet switching using the general-purpose CPU (software-based routers) since the 1990s, and it is the default switching mechanism used by all Cisco platforms that use specialized application-specific integrated circuits (ASICs) and network processing units (NPUs) for high packet throughput (hardware-based routers).

The general-purpose CPUs on software-based and hardware-based routers are similar and perform all the same functions; the difference is that on software-based routers, the general-purpose CPU is in charge of all operations, including CEF switching (software CEF), and the hardware-based routers do CEF switching using forwarding engines that are implemented in specialized ASICs, ternary content addressable memory (TCAM), and NPUs (hardware CEF). Forwarding engines provide the packet switching, forwarding, and route lookup capability to routers.

Key Topic

Ternary Content Addressable Memory

A switch's *ternary content addressable memory (TCAM)* allows for the matching and evaluation of a packet on more than one field. TCAM is an extension of the CAM architecture but enhanced to allow for upper-layer processing such as identifying the Layer 2/3 source/destination addresses, protocol, QoS markings, and so on. TCAM provides more flexibility in searching than does CAM, which is binary. A TCAM search provides three results: 0 for true, 1 false, and X for do not care, which is a ternary combination.

The TCAM entries are stored in Value, Mask, and Result (VMR) format. The value indicates the fields that should be searched, such as the IP address and protocol fields. The mask indicates the field that is of interest and that should be queried. The result indicates the action that should be taken with a match on the value and mask. Multiple actions can be selected besides allowing or dropping traffic, but tasks like redirecting a flow to a QoS policer or specifying a pointer to a different entry in the routing table are possible.

Most switches contain multiple TCAM entries so that inbound/outbound security, QoS, and Layer 2 and Layer 3 forwarding decisions occur all at once. TCAM operates in hardware, providing faster processing and scalability than process switching. This allows for some features like ACLs to process at the same speed regardless of whether there are 10 entries or 500.

Centralized Forwarding

Given the low cost of general-purpose CPUs, the price of software-based routers is becoming more affordable, but at the expense of total packet throughput.

When a route processor (RP) engine is equipped with a forwarding engine so that it can make all the packet switching decisions, this is known as a *centralized forwarding architecture*. If the line cards are equipped with forwarding engines so that they can make packet switching decisions without intervention of the RP, this is known as a *distributed forwarding architecture*.

For a centralized forwarding architecture, when a packet is received on the ingress line card, it is transmitted to the forwarding engine on the RP. The forwarding engine examines the packet's headers and determines that the packet will be sent out a port on the egress line card and forwards the packet to the egress line card to be forwarded.

Distributed Forwarding

For a distributed forwarding architecture, when a packet is received on the ingress line card, it is transmitted to the local forwarding engine. The forwarding engine performs a packet lookup, and if it determines that the outbound interface is local, it forwards the packet out a local interface. If the outbound interface is located on a different line card, the packet is sent across the switch fabric, also known as the backplane, directly to the egress line card, bypassing the RP.

Figure 1-7 shows the difference between centralized and distributed forwarding architectures.

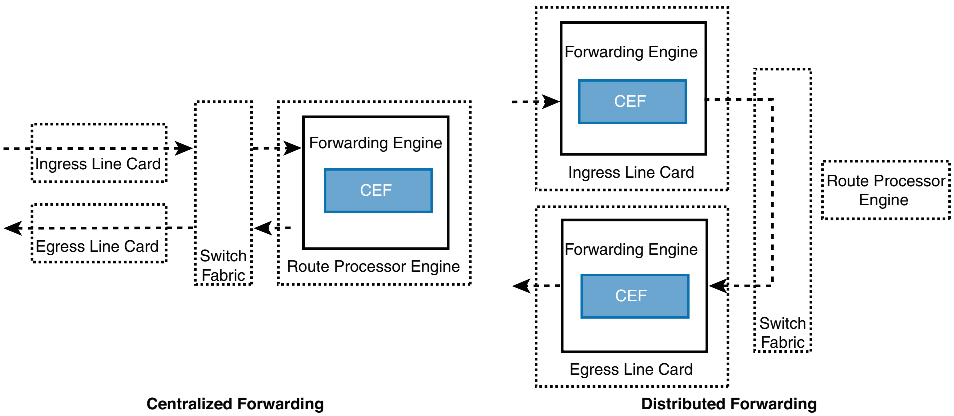


Figure 1-7 *Centralized Versus Distributed Forwarding Architectures*

Key Topic

Software CEF

Software CEF, also known as the *software Forwarding Information Base*, consists of the following components:

- **Forwarding Information Base:** The FIB is built directly from the routing table and contains the next-hop IP address for each destination in the network. It keeps a mirror image of the forwarding information contained in the IP routing table. When a routing or topology change occurs in the network, the IP routing table is updated, and these changes are reflected in the FIB. CEF uses the FIB to make IP destination prefix-based switching decisions.
- **Adjacency table:** The adjacency table, also known as the Adjacency Information Base (AIB), contains the directly connected next-hop IP addresses and their corresponding next-hop MAC addresses, as well as the egress interface's MAC address. The adjacency table is populated with data from the ARP table or other Layer 2 protocol tables.

Figure 1-8 illustrates how the CEF table is built from the routing table. First, the FIB is built from the routing table. The 172.16.10.0/24 prefix is a static route with a next hop of 10.40.40.254, which is dependent upon the 10.40.40.0/24 prefix learned via OSPF. The adjacency pointer in the FIB for the 172.16.10.0/24 entry is exactly the same IP address OSPF uses for the 10.40.40.0/24 prefix (10.10.10.254). The adjacency table is then built using the ARP table and cross-referencing the MAC address with the MAC address table to identify the outbound interface.

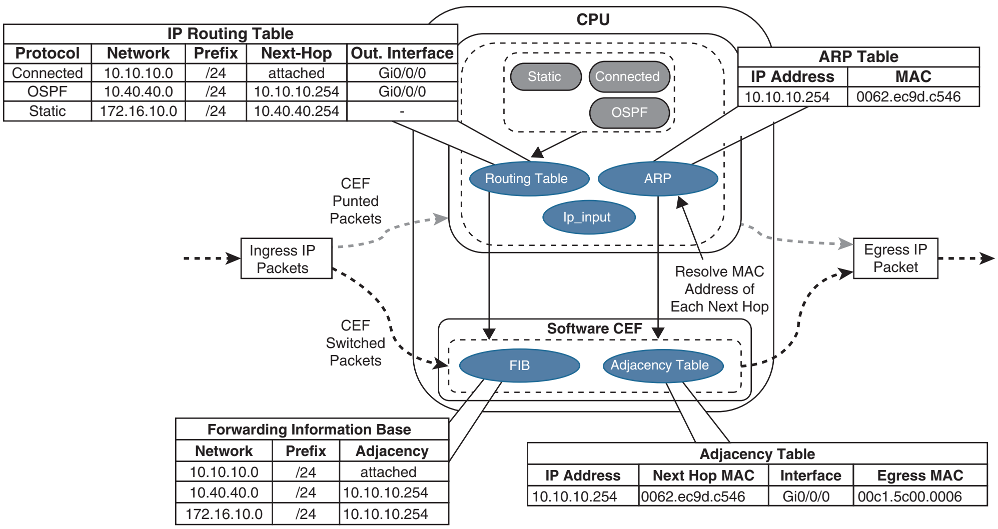


Figure 1-8 *CEF Switching*

Upon receipt of an IP packet, the FIB is checked for a valid entry. If an entry is missing, it is a "glean" adjacency in CEF, which means the packet should go to the CPU because CEF is unable to handle it. Valid FIB entries continue processing by checking the adjacency table for each packet's destination IP address. Missing adjacency entries invoke the ARP process. Once ARP is resolved, the complete CEF entry can be created.

As part of the packet forwarding process, the packet's headers are rewritten. The overwrites the destination MAC address of a packet with the next-hop router's M from the adjacency table, overwrites the source MAC address with the MAC address of the outgoing Layer 3 interface, decrements the IP time-to-live (TTL) field, recomputes the IP header checksum, and finally delivers the packet to the next-hop router.

NOTE Packets processed by the CPU are typically subject to a rate limiter when an invalid or incomplete adjacency exists to prevent the starving of CPU cycles from other essential processes.

NOTE The TTL is a Layer 3 loop prevention mechanism that reduces a packet's TTL field by 1 for every Layer 3 hop. If a router receives a packet with a TTL of 0, the packet is discarded.

Hardware CEF

The ASICs in hardware-based routers are expensive to design, produce, and troubleshoot. ASICs allow for very high packet rates, but the trade-off is that they are limited in their functionality because they are hardwired to perform specific tasks. The routers are equipped with NPUs that are designed to overcome the inflexibility of ASICs. Unlike ASICs, NPUs are programmable, and their firmware can be changed with relative ease.

The main advantage of the distributed forwarding architectures is that the packet throughput performance is greatly improved by offloading the packet switching responsibilities to the line cards. Packet switching in distributed architecture platforms is done via distributed CEF (dCEF), which is a mechanism in which the CEF data structures are downloaded to forwarding ASICs and the CPUs of all line cards so that they can participate in packet switching; this allows for the switching to be done at the distributed level, thus increasing the packet throughput of the router.

NOTE Software CEF in hardware-based platforms is not used to do packet switching as in software-based platforms; instead, it is used to program the hardware CEF.

Stateful Switchover

Routers specifically designed for high availability include hardware redundancy, such as dual power supplies and route processors (RPs). An RP is responsible for learning the network topology and building the route table (RIB). An RP failure can trigger routing protocol adjacencies to reset, resulting in packet loss and network instability. During an RP failure, it may be more desirable to hide the failure and allow the router to continue forwarding packets using the previously programmed CEF table entries rather than temporarily drop packets while waiting for the secondary RP to reestablish the routing protocol adjacencies and rebuild the forwarding table.

Stateful switchover (SSO) is a redundancy feature that allows a Cisco router with two RPs to synchronize router configuration and control plane state information. The process of mirroring information between RPs is referred to as *checkpointing*. SSO-enabled routers always checkpoint line card operation and Layer 2 protocol states. During a switchover, the standby RP immediately takes control and prevents basic problems such as interface link flaps. However, Layer 3 packet forwarding is disrupted without additional cor

The RP switchover triggers a routing protocol adjacency flap that clears the route table. When the routing table is cleared, the CEF entries are purged, and traffic is no longer routed until the network topology is relearned and the forwarding table is reprogrammed. Enabling nonstop forwarding (NSF) or nonstop routing (NSR) high availability capabilities informs the router(s) to maintain the CEF entries for a short duration and continue forwarding packets through an RP failure until the control plane recovers.

Key Topic

SDM Templates

The capacity of MAC addresses that a switch needs compared to the number of routes that it holds depends on where it is deployed in the network. The memory used for TCAM tables is limited and statically allocated during the bootup sequence of the switch. When a section of a hardware resource is full, all processing overflow is sent to the CPU, which seriously impacts the performance of the switch.

The allocation ratios between the various TCAM tables are stored and can be modified with Switching Database Manager (SDM) templates. Multiple Cisco switches exist, and the SDM template can be configured on Catalyst 9000 switches with the global configuration command **sdm prefer** {**vlan** | **advanced**}. The switch must then be restarted with the **reload** command.

NOTE Every switch in a switch stack must be configured with the same SDM template.

Table 1-2 shows the approximate number of resources available per template. This could vary based on the switch platform or software version in use. These numbers are typical for Layer 2 and IPv4 features. Some features, such as IPv6, use twice the entry size, which means only half as many entries can be created.

Table 1-2 Approximate Number of Feature Resources Allowed by Templates

Resource	Advanced	VLAN
Number of VLANs	4094	4094
Unicast MAC addresses	32,000	32,000
Overflow unicast MAC addresses	512	512
IGMP groups and multicast routes	4000	4000
Overflow IGMP groups and multicast routes	512	512
Directly connected routes	16,000	16,000
Indirectly connected IP hosts	7000	7000
Policy-based routing access control entries (ACEs)	1024	0
QoS classification ACEs	3000	3000
Security ACEs	3000	3000
NetFlow ACEs	1024	1024
Input Microflow policer ACEs	256,000	0
Output Microflow policer ACEs	256,000	0
FSPAN ACEs	256	256
Control Plane Entries	512	512

The current SDM template can viewed with the command **show sdm prefer**, as demonstrated in Example 1-17.

Example 1-17 *Viewing the Current SDM Template*

```
SW1# show sdm prefer
Showing SDM Template Info

This is the Advanced (high scale) template.
  Number of VLANs:                                  4094
  Unicast MAC addresses:                            32768
  Overflow Unicast MAC addresses:                   512
  IGMP and Multicast groups:                        4096
  Overflow IGMP and Multicast groups:               512
  Directly connected routes:                        16384
  Indirect routes:                                  7168
  Security Access Control Entries:                  3072
  QoS Access Control Entries:                       2560
  Policy Based Routing ACEs:                        1024
  Netflow ACEs:                                     768
  Wireless Input Microflow policer ACEs:            256
  Wireless Output Microflow policer ACEs:           256
  Flow SPAN ACEs:                                   256
  Tunnels:                                          256
  Control Plane Entries:                            512
  Input Netflow flows:                              8192
  Output Netflow flows:                             16384
  SGT/DGT and MPLS VPN entries:                     3840
  SGT/DGT and MPLS VPN Overflow entries:            512
These numbers are typical for L2 and IPv4 features.
Some features such as IPv6, use up double the entry size;
so only half as many entries can be created.
```

Exam Preparation Tasks

As mentioned in the section "How to Use This Book" in the Introduction, you have a couple of choices for exam preparation: the exercises here, Chapter 30, "Final Preparation," and the exam simulation questions in the Pearson Test Prep Software Online.

Review All Key Topics

Review the most important topics in the chapter, noted with the Key Topic icon in the outer margin of the page. Table 1-3 lists these key topics and the page number on which each is found.

Key Topic

Table 1-3 Key Topics for Chapter 1

Key Topic Element	Description	Page
Paragraph	Collision domain	5
Paragraph	Virtual LANs (VLANs)	7
Section	Access ports	11
Section	Trunk ports	12
Paragraph	Content addressable memory	16
Paragraph	Address resolution protocol (ARP)	19
Paragraph	Packet Routing	20
Paragraph	IP address assignment	21
Section	Process switching	25
Section	Cisco Express Forwarding (CEF)	26
Section	Ternary content addressable memory	26
Section	Software CEF	28
Section	SDM templates	30

Complete Tables and Lists from Memory

There are no memory tables in this chapter.

Define Key Terms

Define the following key terms from this chapter and check your answers in the Glossary:

access port, Address Resolution Protocol (ARP), broadcast domain, Cisco Express Forwarding (CEF), collision domain, content addressable memory (CAM), Layer 2 forwarding, Layer 3 forwarding, Forwarding Information Base (FIB), MAC address table, native VLAN, process switching, Routing Information Base (RIB), trunk port, ternary content addressable memory (TCAM), virtual LAN (VLAN)

Use the Command Reference to Check Your Memory

Table 1-4 lists the important commands from this chapter. To test your memory, cover the right side of the table with a piece of paper, read the description on the left side, and see how much of the command you can remember.

Table 1-4 Command Reference

Task	Command Syntax
Define a VLAN	**vlan** *vlan-id* **name** *vlanname*
Configure an interface as a trunk port	**switchport mode trunk**
Configure an interface as an access port assigned to a specific VLAN	**switchport mode access** **switchport access** {**vlan** *vlan-id* | **name** *name*}
Configure a static MAC address entry	**mac address-table static** *mac-address* **vlan** *vlan-id* **interface** *interface-id*

Task	Command Syntax
Clear MAC addresses from the MAC address table	**clear mac address-table dynamic** [{**address** *mac-address* | **interface** *interface-id* | **vlan** *vlan-id*}]
Assign an IPv4 address to an interface	**ip address** *ip-address subnet-mask*
Assign a secondary IPv4 address to an interface	**ip address** *ip-address subnet-mask* **secondary**
Assign an IPv6 address to an interface	**ipv6 address** *ipv6-address/prefix-length*
Modify the SDM database	**sdm prefer** {**vlan** | **advanced**}
Display the interfaces that are configured as a trunk port and all the VLANs that they permit	**show interfaces trunk**
Display the list of VLANs and their associated ports	**show vlan** [{**brief** | **id** *vlan-id* | **name** *vlanname* | **summary**}]
Display the MAC address table for a switch	**show mac address-table** [**address** *mac-address* | **dynamic** | **vlan** *vlan-id*]
Display the current interface state, including duplex, speed, and link state	**show interfaces**
Display the Layer 2 configuration information for a specific switchport	**show interfaces** *interface-id* **switchport**
Display the ARP table	**show ip arp** [*mac-address* | *ip-address* | **vlan** *vlan-id* | *interface-id*].
Displays the IP interface table	**show ip interface** [**brief** | *interface-id* | **vlan** *vlan-id*]
Display the IPv6 interface table	**show ipv6 interface** [**brief** | *interface-id* | **vlan** *vlan-id*]

References in This Chapter

Bollapragada, Vijay, Russ White, and Curtis Murphy. *Inside Cisco IOS Software Architecture*. (ISBN-13: 9781587058165).

Stringfield, Nakia, Russ White, and Stacia McKee. *Cisco Express Forwarding*. (ISBN-13: 9780133433340).

CHAPTER 2

Spanning Tree Protocol

This chapter covers the following subjects:

Spanning Tree Protocol Fundamentals: This section provides an overview of how switches become aware of other switches and prevent forwarding loops.

Rapid Spanning Tree Protocol: This section examines the improvements made to STP for faster convergence.

A good network design provides redundancy in devices and network links (that is, paths). The simplest solution involves adding a second link between switches to overcome a network link failure or ensuring that a switch is connected to at least two other switches in a topology. However, such topologies cause problems when a switch must forward broadcasts or when unknown unicast flooding occurs. Network broadcasts forward in a continuous loop until the link becomes saturated, and the switch is forced to drop packets. In addition, the MAC address table must constantly change ports as the packets make loops. The packets continue to loop around the topology because there is not a time-to-live (TTL) mechanism for Layer 2 forwarding. The switch CPU utilization increases, as does memory consumption, which could result in the crashing of the switch.

This chapter explains how switches prevent forwarding loops while allowing for redundant links with the use of Spanning Tree Protocol (STP) and Rapid Spanning Tree Protocol (RSTP). Two other chapters also explain STP-related topics:

- **Chapter 3, "Advanced STP Tuning":** Covers advanced STP topics such as BPDU guard and BPDU filter.
- **Chapter 4, "Multiple Spanning Tree Protocol":** Covers Multiple Spanning Tree Protocol.

"Do I Know This Already?" Quiz

The "Do I Know This Already?" quiz allows you to assess whether you should read the entire chapter. If you miss no more than one of these self-assessment questions, you might want to move ahead to the "Exam Preparation Tasks" section. Table 2-1 lists the major headings in this chapter and the "Do I Know This Already?" quiz questions covering the material in those headings so you can assess your knowledge of these specific areas. The answers to the "Do I Know This Already?" quiz appear in Appendix A, "Answers to the 'Do I Know This Already?' Quiz Questions."

Table 2-1 "Do I Know This Already?" Foundation Topics Section-to-Question Mapping

Foundation Topics Section	Questions
Spanning Tree Protocol Fundamentals	1–6
Rapid Spanning Tree Protocol	7–9

1. How many different BPDU types are there?
   a. One
   b. Two
   c. Three
   d. Four
2. What attributes are used to elect a root bridge?
   a. Switch port priority
   b. Bridge priority
   c. Switch serial number
   d. Path cost
3. The original 802.1D specification assigns what value to a 1 Gbps interface?
   a. 1
   b. 2
   c. 4
   d. 19
4. All of the ports on a root bridge are assigned what role?
   a. Root port
   b. Designated port
   c. Superior port
   d. Master port
5. Using default settings, how long does a port stay in the listening state?
   a. 2 seconds
   b. 5 seconds
   c. 10 seconds
   d. 15 seconds
6. Upon receipt of a configuration BPDU with the topology change flag set, how do the downstream switches react?
   a. By moving all ports to a blocking state on all switches
   b. By flushing out all MAC addresses from the MAC address table
   c. By temporarily moving all non-root ports to a listening state
   d. By flushing out all old MAC addresses from the MAC address table
   e. By updating the Topology Change version flag on the local switch database

7. Which of the following is not an RSTP port state?
   a. Blocking
   b. Listening
   c. Learning
   d. Forwarding
8. True or false: In a large Layer 2 switch topology, the infrastructure must fully converge before any packets can be forwarded.
   a. True
   b. False
9. True or false: In a large Layer 2 switch topology that is running RSTP, the infrastructure must fully converge before any packets can be forwarded.
   a. True
   b. False

Foundation Topics

Spanning Tree Protocol Fundamentals

Spanning Tree Protocol (STP) enables switches to become aware of other switches through the advertisement and receipt of bridge protocol data units (BPDUs). STP builds a Layer 2 loop-free topology in an environment by temporarily blocking traffic on redundant ports. STP operates by selecting a specific switch as the master switch and running a tree-based algorithm to identify which redundant ports should not forward traffic.

STP has multiple iterations:

- 802.1D, which is the original specification
- Per-VLAN Spanning Tree (PVST)
- Per-VLAN Spanning Tree Plus (PVST+)
- 802.1W Rapid Spanning Tree Protocol (RSTP)
- 802.1S Multiple Spanning Tree Protocol (MST)

Catalyst switches now operate in PVST+, RSTP, and MST modes. All three of these modes are backward compatible with 802.1D.

IEEE 802.1D STP

The original version of STP comes from the IEEE 802.1D standards and provides support for ensuring a loop-free topology for one VLAN. This topic is vital to understand as a foundation for Rapid Spanning Tree Protocol (RSTP) and Multiple Spanning Tree Protocol (MST).

802.1D Port States

In the 802.1D STP protocol, every port transitions through the following states:

- **Disabled:** The port is in an administratively off position (that is, shut down).
- **Blocking:** The switch port is enabled, but the port is not forwarding any traffic to ensure that a loop is not created. The switch does not modify the MAC address table. It can only receive BPDUs from other switches.
- **Listening:** The switch port has transitioned from a blocking state and can now send or receive BPDUs. It cannot forward any other network traffic. The duration of the state correlates to the STP forwarding time. The next port state is learning.
- **Learning:** The switch port can now modify the MAC address table with any network traffic that it receives. The switch still does not forward any other network traffic besides BPDUs. The duration of the state correlates to the STP forwarding time. The next port state is forwarding.
- **Forwarding:** The switch port can forward all network traffic and can update the MAC address table as expected. This is the final state for a switch port to forward network traffic.
- **Broken:** The switch has detected a configuration or an operational problem on a port that can have major effects. The port discards packets as long as the problem continues to exist.

NOTE The entire 802.1D STP initialization time takes about 30 seconds for a port to enter the forwarding state using default timers.

802.1D Port Types

Key Topic

The 802.1D STP standard defines the following three port types:

- **Root port (RP):** A network port that connects to the root bridge or an upstream switch in the spanning-tree topology. There should be only one root port per VLAN on a switch.
- **Designated port (DP):** A network port that receives and forwards BPDU frames to other switches. Designated ports provide connectivity to downstream devices and switches. There should be only one active designated port on a link.
- **Blocking port:** A network that is not forwarding traffic because of STP calculations.

STP Key Terminology

Key Topic

Several key terms are related to STP:

- **Root bridge:** The root bridge is the most important switch in the Layer 2 topology. All ports are in a forwarding state. This switch is considered the top of the spanning tree for all path calculations by other switches. All ports on the root bridge are categorized as designated ports.

- **Bridge protocol data unit (BPDU):** This network packet is used for network switches to identify a hierarchy and notify of changes in the topology. A BPDU uses the destination MAC address 01:80:c2:00:00:00. There are two types of BPDUs:
  - **Configuration BPDU:** This type of BPDU is used to identify the root bridge, root ports, designated ports, and blocking ports. The configuration BPDU consists of the following fields: STP type, root path cost, root bridge identifier, local bridge identifier, max age, hello time, and forward delay.
  - **Topology change notification (TCN) BPDU:** This type of BPDU is used to communicate changes in the Layer 2 topology to other switches. This is explained in greater detail later in the chapter.
- **Root path cost:** This is the combined cost for a specific path toward the root switch.
- **System priority:** This 4-bit value indicates the preference for a switch to be root bridge. The default value is 32,768.
- **System ID extension:** This 12-bit value indicates the VLAN that the BPDU correlates to. The system priority and system ID extension are combined as part of the switch's identification of the root bridge.
- **Root bridge identifier:** This is a combination of the root bridge system MAC address, system ID extension, and system priority of the root bridge.
- **Local bridge identifier:** This is a combination of the local switch's bridge system MAC address, system ID extension, and system priority of the root bridge.
- **Max age:** This is the maximum length of time that passes before a bridge port saves its BPDU information. The default value is 20 seconds, but the value can be configured with the command **spanning-tree vlan** *vlan-id* **max-age** *maxage*. If a switch loses contact with the BPDU's source, it assumes that the BPDU information is still valid for the duration of the Max Age timer.
- **Hello time:** This is the time that a BPDU is advertised out of a port. The default value is 2 seconds, but the value can be configured to 1 to 10 seconds with the command **spanning-tree vlan** *vlan-id* **hello-time** *hello-time*.
- **Forward delay:** This is the amount of time that a port stays in a listening and learning state. The default value is 15 seconds, but the value can be changed to a value of 15 to 30 seconds with the command **spanning-tree vlan** *vlan-id* **forward-time** *forward-time*.

Answers to the "Do I Know This Already?" quiz:

1 B **2** B **3** C **4** B **5** D **6** D **7** A, B **8** B **9** B

NOTE STP was defined before modern switches existed. The devices that originally used STP were known as bridges. Switches perform the same role at a higher speed and scale while essentially bridging Layer 2 traffic. The terms *bridge* and *switch* are interchangeable in this context.

Spanning Tree Path Cost

The interface STP cost is an essential component for root path calculation because the root path is found based on the cumulative interface STP cost to reach the root bridge. The interface STP cost was originally stored as a 16-bit value with a reference value of 20 Gbps. As switches have developed with higher-speed interfaces, 10 Gbps might not be enough. Another method, called *long mode*, uses a 32-bit value and uses a reference speed of 20 Tbps. The original method, known as *short mode*, is the default mode.

Table 2-2 displays a list of interface speeds and the correlating interface STP costs.

Table 2-2 Default Interface STP Port Costs

Link Speed	Short-Mode STP Cost	Long-Mode STP Cost
10 Mbps	100	2,000,000
100 Mbps	19	200,000
1 Gbps	4	20,000
10 Gbps	2	2,000
20 Gbps	1	1,000
100 Gbps	1	200
1 Tbps	1	20
10 Tbps	1	2

Devices can be configured with the long-mode interface cost with the command **spanning-tree pathcost method long**. The entire Layer 2 topology should use the same setting for every device in the environment to ensure a consistent topology. Before enabling this setting in an environment, it is important to conduct an audit to ensure that the setting will work.

Building the STP Topology

This section focuses on the logic switches use to build an STP topology. Figure 2-1 shows the simple topology used here to demonstrate some important spanning tree concepts. The configurations on all the switches do not include any customizations for STP, and the focus is primarily on VLAN 1, but VLANs 10, 20, and 99 also exist in the topology. SW1 has been identified as the root bridge, and the RP, DP, and blocking ports have been identified visually to assist in the following sections.

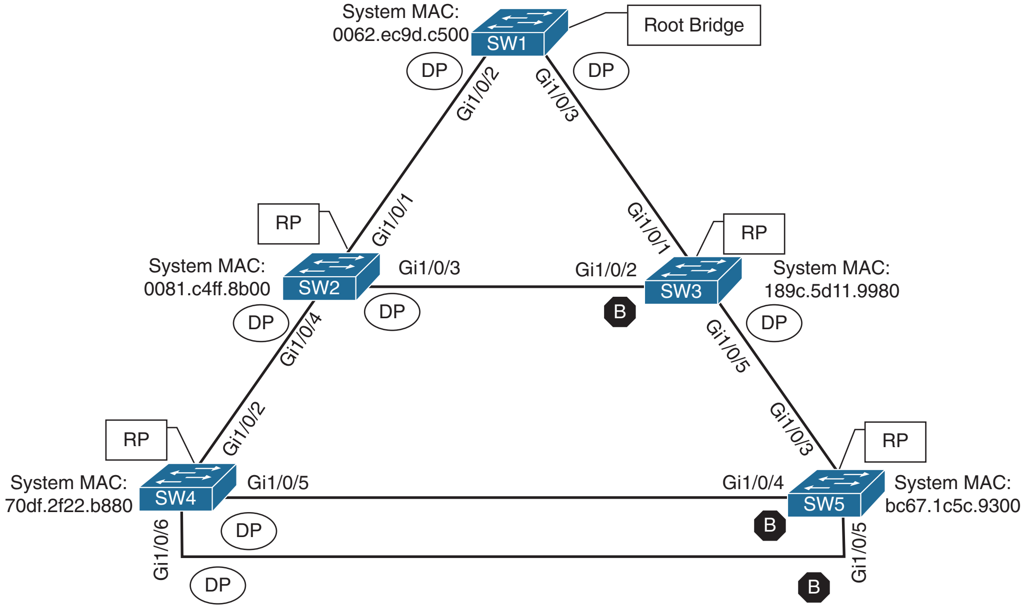


Figure 2-1 *Basic STP Topology*

Key Topic

Root Bridge Election

The first step with STP is to identify the root bridge. As a switch initializes, it assumes that it is the root bridge and uses the local bridge identifier as the root bridge identifier. It then listens to its neighbor's configuration BPDU and does the following:

- If the neighbor's configuration BPDU is inferior to its own BPDU, the switch ignores that BPDU.
- If the neighbor's configuration BPDU is preferred to its own BPDU, the switch updates its BPDUs to include the new root bridge identifier along with a new root path cost that correlates to the total path cost to reach the new root bridge. This process continues until all switches in a topology have identified the root bridge switch.

STP deems a switch more preferable if the priority in the bridge identifier is lower than the priority of the other switch's configuration BPDUs. If the priority is the same, then the switch prefers the BPDU with the lower system MAC.

NOTE Generally, older switches have a lower MAC address and are considered more preferable. Configuration changes can be made for optimizing placement of the root switch in a Layer 2 topology.

In Figure 2-1, SW1 can be identified as the root bridge because its system MAC address (0062.ec9d.c500) is the lowest in the topology. This is further verified by using the command **show spanning-tree root** to display the root bridge. Example 2-1 demonstrates this command being executed on SW1. The output includes the VLAN number, root bridge identifier, root path cost, hello time, max age time, and forwarding delay. Because SW1 is the root bridge, all ports are designated ports, so the Root Port field is empty. This is one way to verify that the connected switch is the root bridge for the VLAN.

Example 2-1 *Verifying the STP Root Bridge*

```
SW1# show spanning-tree root
                                        Root    Hello Max Fwd
Vlan                   Root ID          Cost    Time  Age Dly  Root Port
---------------- -------------------- --------- ----- --- ---  ------------
VLAN0001         32769 0062.ec9d.c500         0    2   20  15
VLAN0010         32778 0062.ec9d.c500         0    2   20  15
VLAN0020         32788 0062.ec9d.c500         0    2   20  15
VLAN0099         32867 0062.ec9d.c500         0    2   20  15
```

In Example 2-1, notice that the root bridge priority on SW1 for VLAN 1 is 32,769 and not 32,768. The priority in the configuration BPDU packets is actually the priority plus the value of the *sys-id-ext* (which is the VLAN number). You can confirm this by looking at VLAN 10, which has a priority of 32,778, which is 10 higher than 32,768.

The advertised root path cost is always the value calculated on the local switch. As the BPDU is received, the local root path cost is the advertised root path cost plus the local interface port cost. The root path cost is always zero on the root bridge. Figure 2-2 illustrates the root path cost as SW1 advertises the configuration BPDUs toward SW3 and then SW3's configuration BPDUs toward SW5.

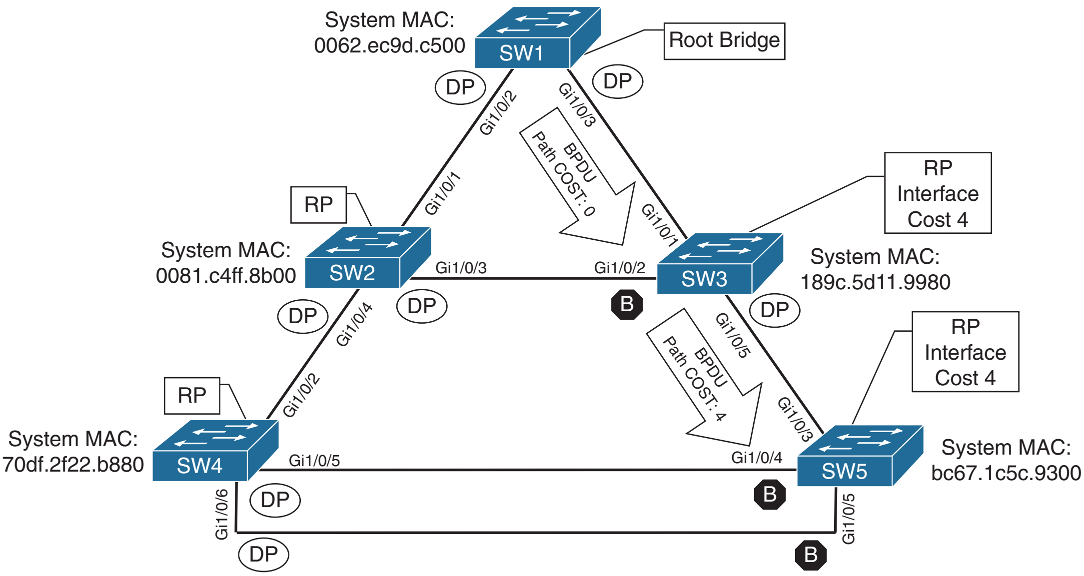


Figure 2-2 *STP Path Cost Advertisements*

Example 2-2 shows the output of the **show spanning-tree root** command run on SW2 and SW3. The Root ID field is exactly the same as for SW1, but the root path cost has changed to 4 because both switches must use the 1 Gbps link to reach SW1. Gi1/0/1 has been identified on both switches as the root port.

Example 2-2 *Identifying the Root Ports*

```
SW2# show spanning-tree root
                                        Root    Hello Max Fwd
Vlan                   Root ID          Cost    Time  Age Dly  Root Port
---------------- -------------------- --------- ----- --- ---  ------------
VLAN0001         32769 0062.ec9d.c500         4    2   20  15  Gi1/0/1
VLAN0010         32778 0062.ec9d.c500         4    2   20  15  Gi1/0/1
VLAN0020         32788 0062.ec9d.c500         4    2   20  15  Gi1/0/1
VLAN0099         32867 0062.ec9d.c500         4    2   20  15  Gi1/0/1
```

```
SW3# show spanning-tree root
                                        Root    Hello Max Fwd
Vlan                   Root ID          Cost    Time  Age Dly  Root Port
---------------- -------------------- --------- ----- --- ---  ------------
VLAN0001         32769 0062.ec9d.c500         4    2   20  15  Gi1/0/1
VLAN0010         32778 0062.ec9d.c500         4    2   20  15  Gi1/0/1
VLAN0020         32788 0062.ec9d.c500         4    2   20  15  Gi1/0/1
VLAN0099         32867 0062.ec9d.c500         4    2   20  15  Gi1/0/1
```

Key Topic

Locating Root Ports

After the switches have identified the root bridge, they must determine their root port (RP). The root bridge continues to advertise configuration BPDUs out all of its ports. The switch compares the BPDU information to identify the RP. The RP is selected using the following logic (where the next criterion is used in the event of a tie):

1. The interface associated to lowest path cost is more preferred.
2. The interface associated to the lowest system priority of the advertising switch is preferred next.
3. The interface associated to the lowest system MAC address of the advertising switch is preferred next.
4. When multiple links are associated to the same switch, the lowest port priority from the advertising switch is preferred.
5. When multiple links are associated to the same switch, the lower port number from the advertising switch is preferred.

Example 2-3 shows the output of running the command **show spanning-tree root** on SW4 and SW5. The Root ID field is exactly the same as on SW1, SW2, and SW3 in Examples 2-1 and 2-2. However, the root path cost has changed to 8 because both switches (SW4 and SW5) must traverse two 1 Gbps link to reach SW1. Gi1/0/2 was identified as the RP for SW4, and Gi1/0/3 was identified as the RP for SW5.

Example 2-3 *Identifying the Root Ports on SW4 and SW5*

```
SW4# show spanning-tree root

                                        Root    Hello Max Fwd
Vlan                   Root ID          Cost    Time  Age Dly  Root Port
---------------- ------------------- --------- ----- --- ---  ------------
VLAN0001         32769 0062.ec9d.c500         8    2   20  15  Gi1/0/2
VLAN0010         32778 0062.ec9d.c500         8    2   20  15  Gi1/0/2
VLAN0020         32788 0062.ec9d.c500         8    2   20  15  Gi1/0/2
VLAN0099         32867 0062.ec9d.c500         8    2   20  15  Gi1/0/2
```

```
SW5# show spanning-tree root
                                        Root    Hello Max Fwd
Vlan                   Root ID          Cost    Time  Age Dly  Root Port
---------------- ------------------- --------- ----- --- ---  ------------
VLAN0001         32769 0062.ec9d.c500         8    2   20  15  Gi1/0/3
VLAN0010         32778 0062.ec9d.c500         8    2   20  15  Gi1/0/3
VLAN0020         32788 0062.ec9d.c500         8    2   20  15  Gi1/0/3
VLAN0099         32867 0062.ec9d.c500         8    2   20  15  Gi1/0/3
```

The root bridge can be identified for a specific VLAN through the use of the command **show spanning-tree root** and examination of the CDP or LLDP neighbor information to identify the host name of the RP switch. The process can be repeated until the root bridge is located.

Locating Blocked Designated Switch Ports

Now that the root bridge and RPs have been identified, all other ports are considered designated ports. However, if two non-root switches are connected to each other on their designated ports, one of those switch ports must be set to a blocking state to prevent a forwarding loop. In our sample topology, this would apply to the following links:

SW2 Gi1/0/3 ← → SW3 Gi1/0/2

SW4 Gi1/0/5 ← → SW5 Gi1/0/4

SW4 Gi1/0/6 ← → SW5 Gi1/0/5

The logic to calculate which ports should be blocked between two non-root switches is as follows:

1. The interface is a designated port and must not be considered an RP.
2. The switch with the lower path cost to the root bridge forwards packets, and the one with the higher path cost blocks. If they tie, they move on to the next step.
3. The system priority of the local switch is compared to the system priority of the remote switch. The local port is moved to a blocking state if the remote system priority is lower than that of the local switch. If they tie, they move on to the next step.
4. The system MAC address of the local switch is compared to the system priority of the remote switch. The local designated port is moved to a blocking state if the remote system MAC address is lower than that of the local switch. If the links are connected to the same switch, they move on to the next step.

All three links (SW2 Gi1/0/3 ↔ SW3 Gi1/0/2, SW4 Gi1/0/5 ↔ SW5 Gi1/0/4, and SW4 Gi1/0/6 ↔ SW5 Gi1/0/5) would use step 4 of the process just listed to identify which port moves to a blocking state. SW3's Gi1/0/2, SW5's Gi1/0/5, and SW5's Gi1/0/6 ports would all transition to a blocking state because the MAC addresses are lower for SW2 and SW4.

The command **show spanning-tree** [**vlan** *vlan-id*] provides useful information for locating a port's STP state. Example 2-4 shows this command being used to show SW1's STP information for VLAN 1. The first portion of the output displays the relevant root bridge's information, which is followed by the local bridge's information. The associated interface's STP port cost, port priority, and port type are displayed as well. All of SW1's ports are designated ports (Desg) because SW1 is the root bridge.

These port types are expected on Catalyst switches:

- **Point-to-point (P2P):** This port type connects with another network device (PC or RSTP switch).
- **P2P edge:** This port type specifies that portfast is enabled on this port.

Example 2-4 *Viewing SW1's STP Information*

```
SW1# show spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol rstp
! This section displays the relevant information for the STP root bridge
  Root ID     Priority     32769
              Address      0062.ec9d.c500
              This bridge is the root
              Hello Time    2 sec  Max Age 20 sec  Forward Delay 15 sec
! This section displays the relevant information for the Local STP bridge
  Bridge ID  Priority     32769  (priority 32768 sys-id-ext 1)
              Address      0062.ec9d.c500
              Hello Time    2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time   300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi1/0/2             Desg FWD 4         128.2    P2p
Gi1/0/3             Desg FWD 4         128.3    P2p
Gi1/0/14            Desg FWD 4         128.14   P2p Edge
```

NOTE If the Type field includes *TYPE_Inc -, this indicates a port configuration mismatch between this Catalyst switch and the switch it is connected to. Common issues are the port type being incorrect and the port mode (access versus trunk) being misconfigured.

Example 2-5 shows the STP topology for SW2 and SW3. Notice that in the first root bridge section, the output provides the total root path cost and the port on the switch that is identified as the RP.

All the ports on SW2 are in a forwarding state, but port Gi1/0/2 on SW3 is in a blocking (BLK) state. Specifically, SW3's Gi1/0/2 port has been designated as an alternate port to reach the root in the event that the Gi1/0/1 connection fails.

The reason that SW3's Gi1/0/2 port rather than SW2's Gi1/0/3 port was placed into a blocking state is that SW2's system MAC address (0081.c4ff.8b00) is lower than SW3's system MAC address (189c.5d11.9980). This can be deduced by looking at the system MAC addresses in the output and confirmed by the topology in Figure 2-1.

Example 2-5 *Verifying the Root and Blocking Ports for a VLAN*

```
SW2# show spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol rstp
  Root ID    Priority    32769
             Address     0062.ec9d.c500
             Cost        4
             Port        1 (GigabitEthernet1/0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     0081.c4ff.8b00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi1/0/1             Root FWD 4         128.1    P2p
Gi1/0/3             Desg FWD 4         128.3    P2p
Gi1/0/4             Desg FWD 4         128.4    P2p
```

```
SW3# show spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol rstp
! This section displays the relevant information for the STP root bridge
  Root ID    Priority    32769
             Address     0062.ec9d.c500
             Cost        4
             Port        1 (GigabitEthernet1/0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
```

```
! This section displays the relevant information for the Local STP bridge
  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     189c.5d11.9980
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi1/0/1             Root FWD 4         128.1    P2p
Gi1/0/2             Altn BLK 4         128.2    P2p
Gi1/0/5             Desg FWD 4         128.5    P2p
```

Verification of VLANS on Trunk Links

All the interfaces that participate in a VLAN are listed in the output of the command **show spanning-tree**. Using this command can be a daunting task for trunk ports that carry multiple VLANs. The output includes the STP state for every VLAN on an interface for every switch interface. The command **show spanning-tree interface** *interface-id* [**detail**] drastically reduces the output to the STP state for only the specified interface. The optional **detail** keyword provides information on port cost, port priority, number of transitions, link type, and count of BPDUs sent or received for every VLAN supported on that interface. Example 2-6 demonstrates the use of both iterations of the command.

If a VLAN is missing on a trunk port, you can check the trunk port configuration for accuracy. Trunk port configuration is covered in more detail in Chapter 5, "VLAN Trunks and EtherChannel Bundles." A common problem is that a VLAN may be missing from the allowed VLANs list for that trunk interface.

Example 2-6 *Viewing VLANs Participating with STP on an Interface*

```
SW3# show spanning-tree interface gi1/0/1

Vlan                Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
VLAN0001            Root FWD 4         128.1    P2p
VLAN0010            Root FWD 4         128.1    P2p
VLAN0020            Root FWD 4         128.1    P2p
VLAN0099            Root FWD 4         128.1    P2p
```

```
SW3# show spanning-tree interface gi1/0/1 detail
! Output omitted for brevity
 Port 1 (GigabitEthernet1/0/1) of VLAN0001 is root forwarding
   Port path cost 4, Port priority 128, Port Identifier 128.1.
   Designated root has priority 32769, address 0062.ec9d.c500
   Designated bridge has priority 32769, address 0062.ec9d.c500
   Designated port id is 128.3, designated path cost 0
   Timers: message age 16, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
```

```
  BPDU: sent 15, received 45908

Port 1 (GigabitEthernet1/0/1) of VLAN0010 is root forwarding
  Port path cost 4, Port priority 128, Port Identifier 128.1.
  Designated root has priority 32778, address 0062.ec9d.c500
  Designated bridge has priority 32778, address 0062.ec9d.c500
  Designated port id is 128.3, designated path cost 0
  Timers: message age 15, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
MAC  BPDU: sent 15, received 22957
..
```

Key Topic

STP Topology Changes

In a stable Layer 2 topology, configuration BPDUs always flow from the root bridge toward the edge switches. However, changes in the topology (for example, switch failure, link failure, or links becoming active) have an impact on all the switches in the Layer 2 topology.

The switch that detects a link status change sends a topology change notification (TCN) BPDU toward the root bridge, out its RP. If an upstream switch receives the TCN, it sends out an acknowledgment and forwards the TCN out its RP to the root bridge.

Upon receipt of the TCN, the root bridge creates a new configuration BPDU with the Topology Change flag set, and it is then flooded to all the switches. When a switch receives a configuration BPDU with the Topology Change flag set, all switches change their MAC address timer to the forwarding delay timer (with a default of 15 seconds). This flushes out MAC addresses for devices that have not communicated in that 15-second window but maintains MAC addresses for devices that are actively communicating.

Flushing the MAC address table prevents a switch from sending traffic to a host that is no longer reachable by that port. However, a side effect of flushing the MAC address table is that it temporarily increases the unknown unicast flooding while it is rebuilt. Remember that this can impact hosts because of their CSMA/CD behavior. The MAC address timer is then reset to normal (300 seconds by default) after the second configuration BPDU is received.

TCNs are generated on a VLAN basis, so the impact of TCNs directly correlates to the number of hosts in a VLAN. As the number of hosts increase, the more likely TCN generation is to occur and the more hosts that are impacted by the broadcasts. Topology changes should be checked as part of the troubleshooting process. Chapter 3 describes mechanisms such as portfast that modify this behavior and reduce the generation of TCNs.

Topology changes are seen with the command **show spanning-tree** [**vlan** *vlan-id*] **detail** on a switch bridge. The output of this command shows the topology change count and time since the last change has occurred. A sudden or continuous increase in TCNs indicates a potential problem and should be investigated further for flapping ports or events on a connected switch.

Example 2-7 displays the output of the **show spanning-tree vlan 10 detail** command. Notice that it includes the time since the last TCN was detected and the interface from which the TCN originated.

Example 2-7 *Viewing a Detailed Version of Spanning Tree State*

```
SW1# show spanning-tree vlan 10 detail

 VLAN0010 is executing the rstp compatible Spanning Tree protocol
  Bridge Identifier has priority 32768, sysid 10, address 0062.ec9d.c500
  Configured hello time 2, max age 20, forward delay 15, transmit hold-count 6
  We are the root of the spanning tree
  Topology change flag not set, detected flag not set
  Number of topology changes 42 last change occurred 01:02:09 ago
          from GigabitEthernet1/0/2
  Times:  hold 1, topology change 35, notification 2
          hello 2, max age 20, forward delay 15
  Timers: hello 0, topology change 0, notification 0, aging 300
```

The process of determining why TCNs are occurring involves checking a port to see whether it is connected to a host or to another switch. If it is connected to another switch, you need to connect to that switch and repeat the process of examining the STP details. You might need to examine CDP tables or your network documentation. You can execute the **show spanning-tree** [**vlan** *vlan-id*] **detail** command again to find the last switch in the topology to identify the problematic port.

Converging with Direct Link Failures

When a switch loses power or reboots, or when a cable is removed from a port, the Layer 1 signaling places the port into a down state, which can notify other processes, such as STP. STP considers such an event a direct link failure and can react in one of three ways, depending upon the topology. This section explains each of these three possible scenarios with a simple three-switch topology where SW1 is the root switch.

Direct Link Failure Scenario 1

In the first scenario, the link between SW2 and SW3 fails. SW2's Gi1/0/3 port is the DP, and SW3's Gi1/0/2 port is in a blocking state. Because SW3's Gi1/0/2 port is already in a blocking state, there is no impact to traffic between the two switches as they both transmit data through SW1. Both SW2 and SW3 will advertise a TCN toward the root switch, which results in the Layer 2 topology flushing its MAC address table.

Direct Link Failure Scenario 2

In the second scenario, the link between SW1 and SW3 fails. Network traffic from SW1 or SW2 toward SW3 is impacted because SW3's Gi1/0/2 port is in a blocking state. Figure 2-3 illustrates the failure scenario and events that occur to stabilize the STP topology:

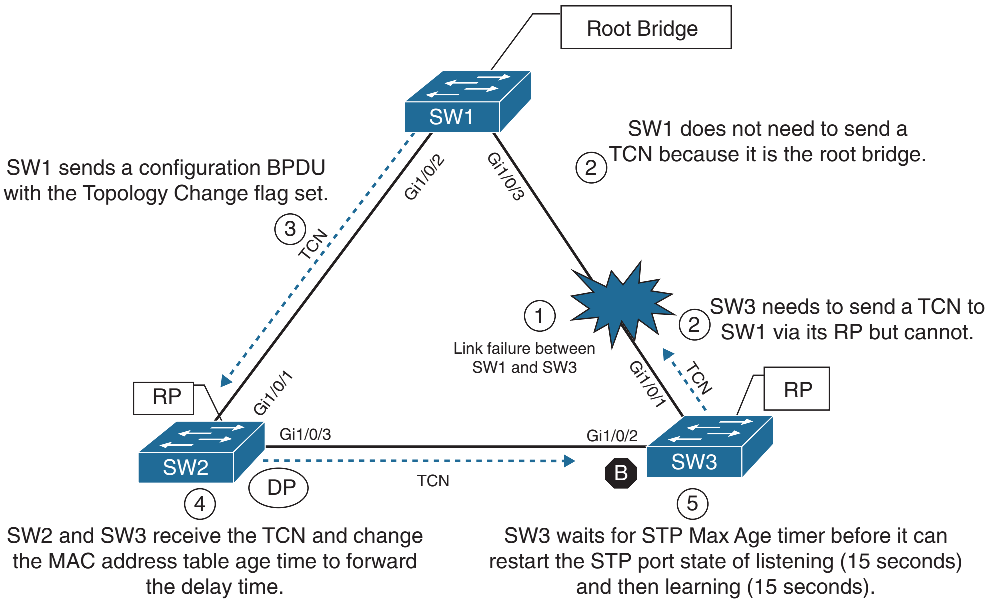


Figure 2-3 *Convergence with Direct Link Failure Between SW1 and SW3*

Phase 1. SW1 detects a link failure on its Gi1/0/3 interface. SW3 detects a link failure on its Gi1/0/1 interface.

Phase 2. Normally SW1 would generate a TCN flag out its root port, but it is the root bridge, so it does not. SW1 would advertise a TCN if it were not the root bridge.

SW3 removes its best BPDU received from SW1 on its Gi1/0/1 interface because it is now in a down state. At this point, SW3 would attempt to send a TCN toward the root switch to notify it of a topology change; however, its root port is down.

Phase 3. SW1 advertises a configuration BPDU with the Topology Change flag out of all its ports. This BPDU is received and relayed to all switches in the environment.

> **NOTE** If other switches were connected to SW1, they would receive a configuration BPDU with the Topology Change flag set as well. These packets have an impact for all switches in the same Layer 2 domain.

Phase 4. SW2 and SW3 receive the configuration BPDU with the Topology Change flag. These switches then reduce the MAC address age timer to the forward delay timer to flush out older MAC entries. In this phase, SW2 does not know what changed in the topology.

Phase 5. SW3 must wait until it hears from the root bridge again or the Max Age timer expires before it can reset the port state and start to listen for BPDUs on the Gi1/0/2 interface (which was in the blocking state previously).

The total convergence time for SW3 is 30 seconds: 15 seconds for the listening state and 15 seconds for the learning state before SW3's Gi1/0/2 can be made the RP.

Direct Link Failure Scenario 3

In the third scenario, the link between SW1 and SW2 fails. Network traffic from SW1 or SW3 toward SW2 is impacted because SW3's Gi1/0/2 port is in a blocking state. Figure 2-4 illustrates the failure scenario and events that occur to stabilize the STP topology:

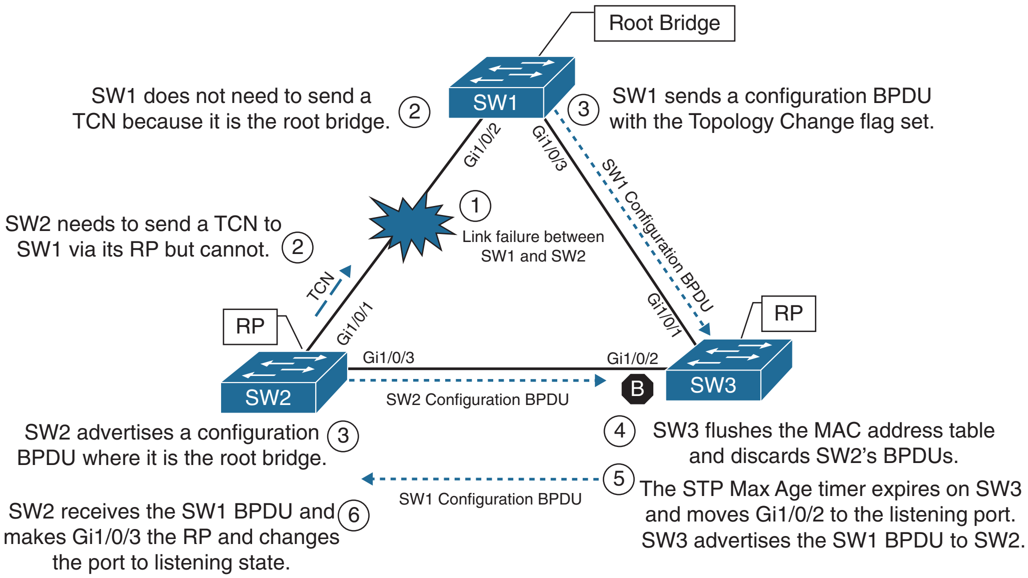


Figure 2-4 *Convergence with Direct Link Failure Between SW1 and SW2*

Phase 1. SW1 detects a link failure on its Gi1/0/1 interface. SW2 detects a link failure on its Gi1/0/3 interface.

Phase 2. Normally SW1 would generate a TCN flag out its root port, but it is the root bridge, so it does not. SW1 would advertise a TCN if it were not the root bridge.

SW2 removes its best BPDU received from SW1 on its Gi1/0/1 interface because it is now in a down state. At this point, SW2 would attempt to send a TCN toward the root switch to notify it of a topology change; however, its root port is down.

Phase 3. SW1 advertises a configuration BPDU with the Topology Change flag out of all its ports. This BPDU is then received and relayed to SW3. SW3 cannot relay this to SW2 as its Gi1/0/2 port is still in a blocking state.

SW2 assumes that it is now the root bridge and advertises configuration BPDUs with itself as the root bridge.

Phase 4. SW3 receives the configuration BPDU with the Topology Change flag from SW1. SW3 reduces the MAC address age timer to the forward delay timer to flush out older MAC entries. SW3 receives SW2's inferior BPDUs and discards them as it is still receiving superior BPDUs from SW1.

Phase 5. The Max Age timer on SW3 expires, and now SW3's Gi1/0/2 port transitions from blocking to listening state. SW3 can now forward the next configuration BPDU it receives from SW1 to SW2.

Phase 6. SW2 receives SW1's configuration BPDU via SW3 and recognizes it as superior. It marks its Gi1/0/3 interface as the root port and transitions it to the listening state.

The total convergence time for SW2 is 52 seconds: 20 seconds for the Max Age timer on SW3, 2 seconds for the configuration BPDU from SW3, 15 seconds for the listening state on SW2, and 15 seconds for the learning state.

Indirect Failures

There are some failure scenarios where STP communication between switches is impaired or filtered while the network link remains up. This situation is known as an *indirect link failure*, and timers are required to detect and remediate the topology. Figure 2-5 illustrates an impediment or data corruption on the link between SW1 and SW3 along with the logic to resolve the loss of network traffic:

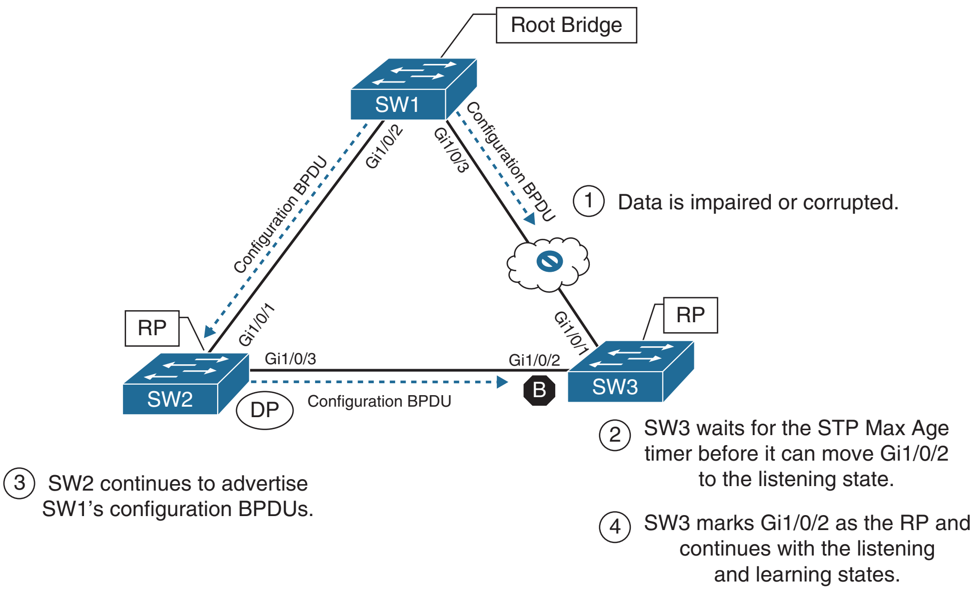


Figure 2-5 *Convergence with Indirect Link Failure*

Phase 1. An event occurs that impairs or corrupts data on the link. SW1 and SW3 still report a link up condition.

Phase 2. SW3 stops receiving configuration BPDUs on its RP. It keeps a cached entry for the RP on Gi1/0/1. SW1's configuration BPDUs that are being transmitted via SW2 are discarded as its Gi1/0/2 port is in a blocking state.

Once SW3's Max Age timer expires and flushes the RP's cached entry, SW3 transitions Gi1/0/2 from blocking to listening state.

Phase 3. SW2 continues to advertise SW1's configuration BPDUs toward SW3

Phase 4. SW3 receives SW1's configuration BPDU via SW2 on its Gi1/0/2 interface. This port is now marked as the RP and continues to transition through the listening and learning states.

The total time for reconvergence on SW3 is 52 seconds: 20 seconds for the Max Age timer on SW3, 2 seconds for the configuration BPDU advertisement on SW2, 15 seconds for the listening state on SW3, and 15 seconds for the learning state on SW3.

Key Topic

Rapid Spanning Tree Protocol

802.1D did a decent job of preventing Layer 2 forwarding loops, but it used only one topology tree, which introduced scalability issues. Some larger environments with multiple VLANs need different STP topologies for traffic engineering purposes (for example, load-balancing, traffic steering). Cisco created Per-VLAN Spanning Tree (PVST) and Per-VLAN Spanning Tree Plus (PVST+) to allow more flexibility.

PVST and PVST+ were proprietary spanning protocols. The concepts in these protocols were incorporated with other enhancements to provide faster convergence into the IEEE 802.1W specification, known as Rapid Spanning Tree Protocol (RSTP).

Key Topic

RSTP (802.1W) Port States

RSTP reduces the number of port states to three:

- **Discarding:** The switch port is enabled, but the port is not forwarding any traffic to ensure that a loop is not created. This state combines the traditional STP states disabled, blocking, and listening.
- **Learning:** The switch port modifies the MAC address table with any network traffic it receives. The switch still does not forward any other network traffic besides BPDUs.
- **Forwarding:** The switch port forwards all network traffic and updates the MAC address table as expected. This is the final state for a switch port to forward network traffic.

NOTE A switch tries to establish an RSTP handshake with the device connected to the other end of the cable. If a handshake does not occur, the other device is assumed to be non-RSTP compatible, and the port defaults to regular 802.1D behavior. This means that host devices such as computers, printers, and so on still encounter a significant transmission delay (around 30 seconds) after the network link is established.

RSTP (802.1W) Port Roles

RSTP defines the following port roles:

- **Root port (RP):** A network port that connects to the root switch or an upstream switch in the spanning-tree topology. There should be only one root port per VLAN on a switch.
- **Designated port (DP):** A network port that receives and forwards frames to other switches. Designated ports provide connectivity to downstream devices and switches. There should be only one active designated port on a link.

- **Alternate port:** A network port that provides alternate connectivity toward the root switch through a different switch.
- **Backup port:** A network port that provides link redundancy toward the current root switch. The backup port cannot guarantee connectivity to the root bridge in the event that the upstream switch fails. A backup port exists only when multiple links connect between the same switches.

RSTP (802.1W) Port Types

RSTP defines three types of ports that are used for building the STP topology:

- **Edge port:** A port at the edge of the network where hosts connect to the Layer 2 topology with one interface and cannot form a loop. These ports directly correlate to ports that have the STP portfast feature enabled.
- **Root port:** A port that has the best path cost toward the root bridge. There can be only one root port on a switch.
- **Point-to-point port:** Any port that connects to another RSTP switch with full duplex. Full-duplex links do not permit more than two devices on a network segment, so determining whether a link is full duplex is the fastest way to check the feasibility of being connected to a switch.

NOTE Multi-access Layer 2 devices such as hubs can only connect at half duplex. If a port can only connect via half duplex, it must operate under traditional 802.1D forwarding states.

Key Topic

Building the RSTP Topology

With RSTP, switches exchange handshakes with other RSTP switches to transition through the following STP states faster. When two switches first connect, they establish a bidirectional handshake across the shared link to identify the root bridge. This is straightforward for an environment with only two switches; however, large environments require greater care to avoid creating a forwarding loop. RSTP uses a synchronization process to add a switch to the RSTP topology without introducing a forwarding loop. The synchronization process starts when two switches (such as SW1 and SW2) are first connected. The process proceeds as follows:

1. As the first two switches connect to each other, they verify that they are connected with a point-to-point link by checking the full-duplex status.
2. They establish a handshake with each other to advertise a proposal (in configuration BPDUs) that their interface should be the DP for that port.
3. There can be only one DP per segment, so each switch identifies whether it is the superior or inferior switch, using the same logic as in 802.1D for the system identifier (that is, the lowest priority and then the lowest MAC address). Using the MAC addresses from Figure 2-1, SW1 (0062.ec9d.c500) is the superior switch to SW2 (0081.c4ff.8b00).

4. The inferior switch (SW2) recognizes that it is inferior and marks its local port (Gi1/0/1) as the RP. At that same time, it moves all non-edge ports to a discarding state. At this point in time, the switch has stopped all local switching for non-edge ports.
5. The inferior switch (SW2) sends an agreement (configuration BPDU) to the root bridge (SW1), which signifies to the root bridge that synchronization is occurring on that switch.
6. The inferior switch (SW2) moves its RP (Gi1/0/1) to a forwarding state. The superior switch moves its DP (Gi1/0/2) to a forwarding state, too.
7. The inferior switch (SW2) repeats the process for any downstream switches connected to it.

The RSTP convergence process can occur quickly, but if a downstream switch fails to acknowledge the proposal, the RSTP switch must default to 802.1D behaviors to prevent a forwarding loop.

Exam Preparation Tasks

As mentioned in the section "How to Use This Book" in the Introduction, you have a couple of choices for exam preparation: the exercises here, Chapter 30, "Final Preparation," and the exam simulation questions in the Pearson Test Prep Software Online.

Review All Key Topics

Review the most important topics in the chapter, noted with the Key Topic icon in the outer margin of the page. Table 2-3 lists these key topics and the page number on which each is found.

Key Topic

Table 2-3 Key Topics for Chapter 2

Key Topic Element	Description	Page
List	802.1D port types	37
Section	STP key terminology	38
Section	Root bridge election	40
Section	Locating root ports	42
Section	STP topology changes	47
Section	RSTP	52
Section	RSTP (802.1W) port states	52
Section	Building the RSTP topology	53

Complete Tables and Lists from Memory

There are no memory tables in this chapter.

Define Key Terms

Define the following key terms from this chapter and check your answers in the Glossary:

bridge protocol data unit (BPDU), configuration BPDU, hello time, designated port (DP) forward delay, local bridge identifier, Max Age, root bridge, root bridge identifier, root path cost, root port, system priority, system ID extension, topology change notification (TCN)

Use the Command Reference to Check Your Memory

Table 2-4 lists the important commands from this chapter. To test your memory, cover the right side of the table with a piece of paper, read the description on the left side, and see how much of the command you can remember.

Table 2-4 Command Reference

Task	Command Syntax
Set the STP max age	**spanning-tree vlan** *vlan-id* **max-age**
Set the STP hello interval	**spanning-tree vlan** *vlan-id* **hello-time** *hello-time*
Set the STP forwarding delay	**spanning-tree vlan** *vlan-id* **forward-time** *forward-time*
Display the STP root bridge and cost	**show spanning-tree root**
Display the STP information (root bridge, local bridge, and interfaces) for one or more VLANs	**show spanning-tree** [**vlan** *vlan-id*]
Identify when the last TCN occurred and which port was the reason for it.	**show spanning-tree** [**vlan** *vlan-id*] **detail**