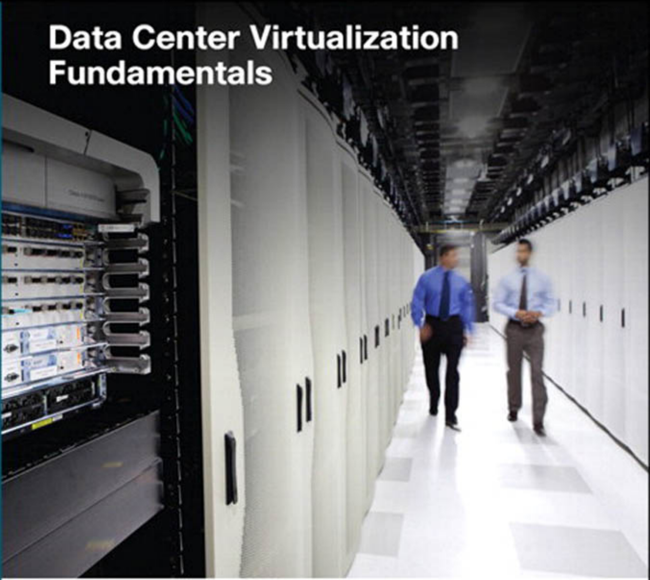
Data Center Virtualization Fundamentals



ciscopress.com

Gustavo Santana

Data Center Virtualization Fundamentals

Gustavo Alessandro Andrade Santana, CCIE No. 8806

Cisco Press

800 East 96th Street

Indianapolis, IN 46240

Data Center Virtualization Fundamentals

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

Library of Congress Control Number: 2013940880

Printed in the United States of America

First Printing June 2013

ISBN-13: 978-1-58714-324-3

ISBN-10: 1-58714-324-0

Warning and Disclaimer

This book is designed to provide information about data center technologies. 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 author, 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.

Corporate and Government Sales

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

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

Feedback Information

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

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

We greatly appreciate your assistance.

Publisher: Paul Boger

Associate Publisher: Dave Dusthimer

Development Editor: Eleanor C. Bru

Copy Editor: John Edwards

Technical Editors: Maurilio Gorito, Krishna Arji

Editorial Assistant: Vanessa Evans

Cover Designer: Mark Shirar

Composition: Jake McFarland

Business Operation Manager, Cisco Press: Jan Cornelssen

Executive Editor: Mary Beth Ray

Managing Editor: Sandra Schroeder

Project Editor: Seth Kerney

Proofreader: Sheri Cain

Indexer: Larry Sweazy

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 Author

Gustavo A. A. Santana, CCIE No. 8806, is a Cisco Technical Solutions Architect working in enterprise and service provider data center projects that require a greater integration among multiple technology areas such as networking, application optimization, storage, and servers.

With more than 15 years of experience in the data center industry, Gustavo has led and coordinated a team of specialized Cisco engineers in Brazil. A true believer of education as a technology catalyst, he has also dedicated himself to the technical development of many IT professionals from customer, partner, and strategic alliance organizations.

In addition to holding two CCIE certifications (Routing & Switching and Storage Networking), Gustavo is also a VMware Certified Professional (VCP) and an SNIA Certified Storage Networking Expert (SCSN-E). A frequent speaker at Cisco and data center industry events, he holds a degree in computer engineering from Instituto Tecnológico de Aeronáutica (ITA-Brazil) and an MBA in strategic IT management from Fundação Getúlio Vargas (FGV-Brazil).

Gustavo maintains a personal blog in which he discusses topics related to data center virtualization technologies at http://gustavoaasantana.net.

About the Technical Reviewers

Maurilio Gorito, CCIE, is the certification manager at Riverbed Technology Institute, leading Riverbed Technology's Certification Program. Previously, he managed the Cisco CCIE Routing and Switching certification program, and he is a triple Cisco Certified Internetwork Expert (CCIE). Maurilio has 25 years of experience in the IT field and more than 20 years of combined experience in education, testing, and assessment. Maurilio has written a book, written articles, and reviewed several technical books for Cisco Press. Maurilio holds bachelor's degrees in education, pedagogy, and mathematics from Centro Universitario Geraldo Di Biasi, Brazil. Maurilio is currently serving on the board of directors at Performance Testing Council (PTC).

Krishna Arji is a senior manager at Cisco. In this role, he is responsible for the development of technology that enables the delivery of Cisco services. Krishna has held various positions in the Cisco Services Technology Group at Cisco. His current assignment is to enable delivery for the Cisco BYOD service offerings. In the past, he played a key role in evaluating and developing technologies required for the delivery of cloud planning, design, and implementation services. Under his leadership, his team developed several technologies to perform routing, switching, data center, security, and WLAN assessments of customers' infrastructures. His areas of expertise include networking, software design and development, and data center technologies such as virtualization. Krishna holds a bachelor's degree in electronics and communications engineering, and he has a master's degree in enterprise software technologies. He is currently enrolled in an MBA program at Haas School of Business, University of California, Berkeley. He has a patent pending with USPTO for Automated Assessments of Storage Area Networks (Serial No. 13/115, 141).

Dedications

This book is dedicated to my wife and true love, Carlene, whose sacrifice and unconditional support were crucial to this endeavor, and to my lovely daughter, Carolina, whose one-year-old curiosity constantly inspired me to go one step further.

I also dedicate this book to my parents, Honorio and Cleia, who have taught me that one can only learn by being fearless and humble.

Finally, this book is also dedicated to every person who has devoted efforts to the rewarding experience of teaching someone.

Acknowledgments

The process of creating a book can be aptly defined as a "sponsored solitude." Certainly, the uncountable lonely writing hours would be fruitless without the support of an entire network of relatives, friends, and professionals who are acknowledged here.

First, I would like to thank my sister Raquel and brother André for the family support during this book writing.

I would also like to express my gratitude to my friend and trusted advisor Alexandre M. S. P. Moraes, who has shared invaluable opinions and insights since the very early stages of this book.

Many thanks to Andrey Lee for the wonderful illustrations in Chapters 1 and 17.

Sincere thanks to Paulo Quinta, Fernanda Spinardi, and Marcelo Ehalt for helping me coordinate my professional life and this writing.

My thanks to the technical reviewers Maurilio Gorito and Krishna Arji for their active contributions and focus to make this work more effective for its targeted readership.

A personal thanks to the Brazilian data center tiger team, which has always served as my favorite "think tank" for best practices and the exchange of experiences.

I would also like to thank two very talented instructors from Firefly: Dan Murray and Fabricio Grimaldi.

I am also very grateful to the people who have contributed with the equipment used in this publication: Shane Hudson and Bilal El-Ayi (from GoldLabs), Ohad Richberg (and his amazing CPOC Israel team), François Tallet and Mark Allen (from the Cisco Enterprise Core Business Unit), and Hugo Marques.

Thanks to all the Pearson production team, especially Ellie Bru and Seth Kerney, who helped me to create the final version of this book.

A special thank-you goes to Mary Beth Ray and Anand Sundaram for supporting the idea of a data center book with a different approach.

Contents at a Glance

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).
- Italics indicate arguments for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets [ ] indicate optional elements.
- Braces { } indicate a required choice.
- Braces within brackets [{ }] indicate a required choice within an optional element.
- Introduction

Foreword

With the rapid growth of the Internet economy and the explosion of information technology, the data center is playing a pivotal role and is one of the most exciting fields in the world of IT today. The trend continues with both virtualization and cloud computing fueling growth and making data center solutions more efficient and scalable. More specifically, organizations using virtualization technologies are seeing greater returns and more viability to deal with the growing demands of the economy.

Data center virtualization is an evolutionary process that was started several years ago within mainframe computer rooms, and it has dramatically intensified in the last few years. Its proposed freedom from physical boundaries has produced benefits in each technology area, and much more importantly, from an architectural perspective.

However, due to these environments' increasing complexity, a data center professional must possess a challenging breadth of knowledge in several different areas, such as networking, storage, servers, operating systems, application, and security.

Data Center Virtualization Fundamentals is a comprehensive book that introduces virtualization technologies in data center environments, encompassing all these knowledge areas. It does not take a product-based approach as many others do, but an architectural one, offering theoretical concepts, illustrative configurations, and real-world designs for each virtualization technique. The book provides a first step for students and professionals who want to understand the state of data center technologies today. And in my opinion, virtualization technologies are the best way to achieve this feat because one must be aware of the physical challenges of data center environments before learning such techniques.

There is a lot of misconception when talking about virtualization, and people immediately think of it in the context of virtual servers. However, virtualization is not restricted to a single technology area in the data center. This book intends to make an account of the main data center virtualization technologies, revealing their impact and applicability to these environments as a whole. It encourages readers to escape their technical comfort-zone and learn how each decision may impact other data center teams. A strong knowledge of the theoretical basis of the data center is necessary to walk amidst clouds, and this is exactly what this book brings.

Author Gustavo A. A. Santana is a seasoned expert with years of experience, and has done a superb job putting this material together. He has demonstrated his skills and command of the technology, using a unique approach in translating the most complex and highly technical information into simple, easy-to-understand material. Readers will definitely appreciate this book.

Finally, this book is an essential reference and will be valuable asset for potential candidates pursuing their Cisco Data Center certifications. I am confident that in reading this book, individuals will inevitably gain extensive knowledge and hands-on experience during their certification preparations. If you're looking for a truly comprehensive guide to virtualization, this is the one!

Yusuf Bhaiji
Senior Manager, Expert Certifications (CCIE, CCDE, CCAr)
Learning@Cisco

"*If you can't explain it simply, you don't understand it well enough.*"
—Albert Einstein

Introduction

"I am very interested in learning data center technologies. How should I start?"

Since I first heard this question, I have seen many IT professionals become overwhelmed with the vertigo-inducing development of new data center technologies. From my perspective, their frustration was mainly caused by attempting to understand this subject without being properly introduced to the most fundamental concepts and definitions related to these complex environments. And that opinion has always formed the basis of my advice to them.

However, as the years passed, I observed how my answer to this question was becoming more elaborate. Understandably, an increasingly diverse background was being required from these professionals, mainly because data center technologies were repeatedly consolidating different areas of knowledge such as networking, storage, application, servers, cabling, and several others. And much to my chagrin, I had to admit that the job of creating an effective introduction to these technologies was getting even harder to "crack."

After developing many learning road maps and customized trainings, I decided to challenge myself in writing a book that would address cutting-edge data center technologies and the core concepts they were based upon. From the start, the mammoth level of minutiae made me realize how close I was to a task such as writing a Beatles biography. And that exact thought inspired me to follow the steps of the best publications about the band: I had to use a *unifying theme*, something that could provide a firm backbone to a progressive presentation of these technologies. It was fairly easy for me to conclude that virtualization was this theme.

Nowadays, virtualization is deeply rooted in data center installations through technologies such as virtual memory, virtual gateways, VLANs, VRFs, virtual contexts, VDCs, vPCs, VNTag, VPLS, OTV, virtual LUNs, VSANs, IVR, NPV, FCoE, virtual machines, service profiles, virtual networking, virtual network services, and many others. All these successful techniques share a common characteristic: They were created to provide *resource optimization*. And for that reason, their examination opens up the chance to address the following:

- Traditional data center deployments and their limitations
- The benefits of each virtualization technology and their behavior
- The changes these technologies provide in data center designs and architectures

As the book cover suggests, virtualization has also modified the human aspects within data center environments. Relieved from the "chains of reality," technical teams have been able to simplify operational tasks and accelerate the adoption of new IT models such as cloud computing. With such a central theme, it was just a question of defining *how* to approach it.

Goals and Methods

This book provides a gradual introduction to innovative data center technologies through a systematic examination of the main infrastructure virtualization techniques. And as an intentional outcome, the book also introduces fundamental concepts and definitions that are required from any professional who is involved with modern data center designs.

Because it is primarily focused on the three main data center infrastructure areas (networking, storage, and server), the book is not based on a single product nor it is written as a configuration guide. Instead, it leverages the broad Cisco Data Center portfolio (and other solutions from the Cisco ecosystem of partners and alliances) to analyze the behavior of each virtualization technique and to offer an architectural perspective of the virtualized data center.

Besides providing an technical account of the evolution of these areas, the book will address each virtualization technology through a flow of topics that involves

- A virtualization classification system (explained in the first chapter), which quickly informs the reader about the main characteristics of the specific technology
- A technology primer that immerses the reader in the physical challenges this virtualization technology overcomes
- A detailed analysis of the technique, including its characteristics, possibilities, scalability, results, and consequences
- A real-world use case scenario that demonstrates the examined technology "in action."

I sincerely believe that design and deployment must be complementary processes. Therefore, *Data Center Virtualization Fundamentals* contains actual configuration examples that were exclusively created to illustrate each virtualization technology and its applicability to data center designs. Nonetheless, I have also included unusual topologies to specifically reinforce concepts explored throughout the book.

Who Should Read This Book

This book was written with a wide audience in mind. Because it provides an in-depth examination of data center virtualization technologies (from conceptualization to implementation), the book will satisfy beginners and experienced IT professionals alike.

In essence, its target audience comprises the following:

- Individuals with basic networking and operating system knowledge who are interested in modern data center design, deployment, and infrastructure optimization techniques
- Candidates for the Cisco Data Center certifications, including CCNA Data Center, CCNP Data Center, and CCIE Data Center

- Professionals that are specialized in a single data center technology area but also intend to acquire a broader architectural knowledge to accelerate their career development

How This Book Is Organized

With the explosion of information brought by the Internet, education in the twenty-first century must always present alternatives to the random accumulation of unstructured data. Therefore, I have intentionally applied constructivist learning theory principles (such as systematic analysis and concept synthesis) to distribute the content throughout the book. Although each chapter can be read out of sequence, their arrangement was designed to provide a logical progression of explanations that brings a more rewarding learning experience for the reader.

In times where blog posts and tweets provide "snacks" of information (do not get me wrong; there are nutritious knowledge bites out there), this book intends to serve a complete "meal," where order and harmonization between chapters matter.

Chapters 1 through 17 and the appendixes cover the following topics:

- **Chapter 1, "Virtualization History and Definitions":** This introductory chapter presents a historical account of virtualization in data center environments and, through some illustrative examples, provides a unified definition of virtualization in this context. It also proposes a classification system (which is called "virtualization taxonomy") that will be used throughout the book to quickly introduce a new virtualization technology for the reader.
- **Chapter 2, "Data Center Network Evolution":** Using the evolution of the Ethernet protocol as a canvas, this chapter addresses the main aspects and factors that govern traditional data center network topologies. It also discusses the general benefits that virtualization can offer to these networks.
- **Chapter 3, "The Humble Beginnings of Network Virtualization":** Focused on the explanation of virtual local-area networks (VLAN) and Virtual Routing and Forwarding (VRF), this chapter provides a deep analysis of these well-established structures as virtualization techniques, illustrating the book approach and revealing important concepts that are hidden behind common knowledge.
- **Chapter 4, "An Army of One: ACE Virtual Contexts":** This chapter discusses the importance of network services in data centers, concentrating on server load balancers. It presents virtual contexts as important tools that can increase flexibility and optimize hardware resources as these application environments scale.
- **Chapter 5, "Instant Switches: Virtual Device Contexts":** The innovative characteristics of virtual device contexts (VDC) are detailed in this chapter, which also shows their applicability in challenging data center network scenarios.

- **Chapter 6, "Fooling Spanning Tree":** This chapter demonstrates how virtualization techniques such as EtherChannel and virtual PortChannel (vPC) have adapted the limitation of Spanning Tree Protocol (STP) to the strict requirements of data center networks. It also introduces FabricPath, a Layer 2 multipathing technology that has provided the most secure path toward the replacement of STP in these environments.
- **Chapter 7, "Virtualized Chassis with Fabric Extenders":** Fabric Extenders (FEX) constitute a virtualization technique that provides cabling optimization and network management consolidation in the data center network access layer. This chapter fully explores the many flavors of this technology.
- **Chapter 8, "A Tale of Two Data Centers":** The classic problem of extending Layer 2 domains between geographically distinct data center sites is discussed throughout this chapter. It builds on concepts developed in previous chapters to offer a hands-on examination of the many different virtualization technologies that can solve this challenge.
- **Chapter 9, "Storage Evolution":** This chapter explores the main concepts related to storage and storage access technologies that are used in data centers today. It also provides an account of how virtualization is deeply ingrained in the interpretation of stored data.
- **Chapter 10, "Islands in the SAN":** Virtual storage-area networks (VSAN) can overcome Fibre Channel fabric challenges in a simple and elegant way. This chapter presents the necessary protocol concepts to understand how they can be applied in real-world scenarios.
- **Chapter 11, "Secret Identities":** This chapter presents three virtualization techniques whose dissimulation tactics benefits data protection, environment isolation, and scalability in storage-area networks.
- **Chapter 12, "One Cable to Unite Us All":** Binding concepts from network and storage virtualization, this chapter fully examines the details and benefits from the I/O consolidation brought about by Data Center Bridging (DCB) and Fibre Channel over Ethernet (FCoE).
- **Chapter 13, "Server Evolution":** This chapter introduces the main concepts related to modern server architectures. It also presents server virtualization and describes how it has changed the operational landscape of data centers in the beginning of the twenty-first century. The chapter also deals with the definition of unified computing and explains how its innovative architecture principles can drastically simplify server environments.
- **Chapter 14, "Changing Personalities":** Although server virtualization has helped to streamline server workloads within a data center, "bare metal" server provisioning and management are still considered massive challenges in these environments. This chapter demonstrates how service profiles can bring several server virtualization benefits to these scenarios.

- **Chapter 15, "Transcending the Rack":** Demonstrating how the technologies explored in this book are extremely intertwined, this chapter shows how server virtualization has also revolutionized networking. It presents the virtual networking concepts through the analysis of VMware vSwitches, Nexus 1000V, and Virtual Machine Fabric Extender (VM-FEX).
- **Chapter 16, "Moving Targets":** The way that virtual machines can migrate between different hosts and locations has also changed the way network services are deployed. This chapter explores the unique characteristics of services provided by solutions such as virtual firewalls (Virtual Security Gateway [VSG] and ASA 1000V), virtual accelerators (virtual Wide Area Application Services [vWAAS]), and virtual routers (CSR 1000V). It also presents site selection as a special network service and illustrates some solutions that can optimize client session routing to roaming virtual machines.
- **Chapter 17, "The Virtual Data Center and Cloud Computing":** This chapter consolidates concepts explained throughout the book to discuss how 1+1 can be more than 2. It discusses how the deployment of multiple virtualization technologies has created a perfect storm for "cloud computing" momentum and how this IT delivery model is influencing the evolution of data center networks.
- **Appendix A, "Cisco Data Center Portfolio":** To preserve the book's focus on virtualization concepts and feature behavior, this appendix contains the description of all Cisco Data Center products that actually deploy these technologies.
- **Appendix B, "IOS, NX-OS, and Application Control Software Command-Line Interface Basics":** If you are not used to the command-line interface characteristics from the different network operating systems used in this book, this appendix will introduce you to their most typical characteristics and definitions.

This page intentionally left blank

Chapter 7

Virtualized Chassis with Fabric Extenders

"Form follows function – that has been misunderstood. Form and function should be one, joined in a spiritual union." (Frank Lloyd Wright)

This chapter examines Fabric Extenders and how they optimize cabling and consolidate network management in data centers. It covers the following topics:

- Server Access Models
- Understanding Fabric Extenders
- Fabric Extender Topologies
- Use Case: Mixed Access Data Center

Table 7-1 categorizes Fabric Extenders using the virtualization taxonomy described in Chapter 1, "Virtualization History and Definitions."

Table 7-1 *Fabric Extender Virtualization Classification*

Virtualization Characteristics	Fabric Extender
Emulation	Modular Ethernet access switch
Type	Pooling
Subtype	Heterogeneous
Scalability	Hardware dependent[1]
Technology Area	Networking
Subarea	Data, control, and management planes

Virtualization Characteristics	Fabric Extender
Advantages	Cabling savings, consolidated management

[1] Refer to Appendix A, "Cisco Data Center Portfolio" for more details and the Cisco online documentation for updated information.

In data center projects, cabling is usually one of the first discussed topics. Unfortunately, it is very common that cabling design is defined before any network and server decisions are made. Because of its intrinsic affinity with three traditional data center teams (facilities, network, and server), server access architecture should be a shared decision that kick-starts optimized projects within each technology division.

To exacerbate this situation, one choice in particular does not seem to have a unanimous answer: Where do you position the access switches? Both Top-of-Rack and End-of-Row connectivity models present advantages and shortcomings, and during the last decades, they have attracted as many fans as detractors within the data center community.

Innovating again with its data center portfolio, Cisco introduced the concept of Fabric Extenders with the Nexus 2000 series. These devices are not Ethernet switches, but remote linecards of a virtualized modular chassis. This virtual entity permits server access to achieve the best aspects of both connectivity models.

Fabric Extenders allow scalable topologies that were not previously possible with traditional Ethernet switches. This chapter describes the design and configuration options of these devices and shows how they can optimize the server access layer within data centers.

Server Access Models

In 2005, the Electronic Industries Alliance (EIA) and the Telecommunications Industry Association (TIA) published the first formal specification for data center infrastructure: ANSI/TIA-942. This standard was intended to provide requirements and guidelines for the design and installation of a data center and includes the facility, network, and cabling design.

This specification defines horizontal cabling as the extension from the mechanical termination of the equipment distribution area (servers) to the horizontal distribution area (switches). Compared with backbone cabling (between switches and other communication equipment), horizontal cabling presents a much higher number of connections and, therefore, has a bigger impact over the entire infrastructure.

As discussed in Chapter 2, "Data Center Network Evolution," the ANSI/TIA-942 specification supports the most popular server connectivity models: Top-of-Rack (ToR) and End-of-Row (EoR). These models define where the access layer switches are positioned in relation to the server localization, and consequently, how the horizontal cabling is designed.

In ToR-based networks, access switches (with 1 or 2 rack units) are usually installed at the top position inside the server cabinets. As a result, horizontal cabling is intrarack and can use a great variety of media such as twisted-pair, fiber, or twinax. To allow easier uplink upgrades, fiber is generally used in the redundant connections to upper-layer devices (core, aggregation, or edge routers).

Figure 7-1 depicts a Top-of-Rack access network.

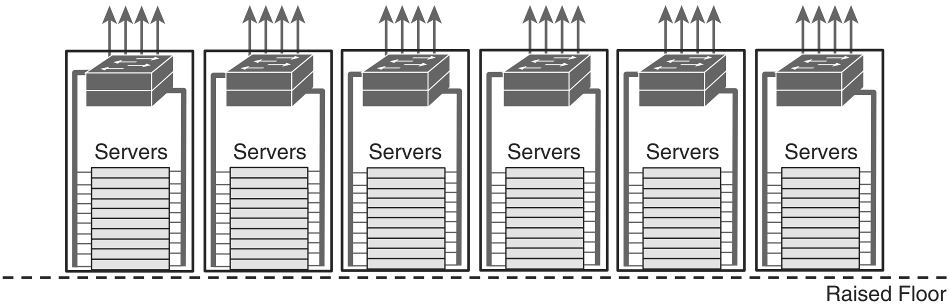


Figure 7-1 *Top-of-Rack Network*

The ToR model permits

- Savings in horizontal cabling (because cable length is reduced)
- Provisioning of fully populated server cabinets
- Per-cabinet migration of connection technologies (Gigabit Ethernet to 10 Gigabit Ethernet, for example).

The number of servers per cabinet heavily influences the port utilization in Top-of-Rack designs. ToR switches usually have 24, 32, 48, or 96 Ethernet interfaces, and if a cabinet does not support a considerable number of servers, some switch interfaces can remain unused.

To increase port utilization on low-populated cabinets, ToR designs can be adapted to other horizontal cabling variations. Figure 7-2 portrays an example where each switch is the "**Top-of-Many-Racks.**"

Note Data center power distribution and cooling capacity usually define how many servers can be installed per cabinet.

A pointed disadvantage for the ToR model is the management effort that must be spent on multiple switches. Because a data center can span hundreds or even thousands of server racks, regular operations (such as the identification of interfaces and firmware upgrades) can become quite challenging.

Alternatively, the EoR model allows the connectivity management of hundreds of servers, which are installed in a row of cabinets, with a pair of devices.

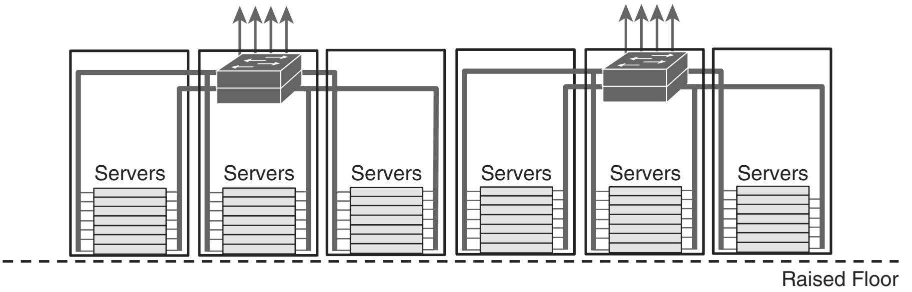


Figure 7-2 *"Top-of-Many-Racks" Network*

EoR switches are typically modular chassis with UTP-based interfaces for horizontal cabling and multiple fiber connections to the network upper-layer devices (core, aggregation, or edge routers).

Figure 7-3 illustrates an End-of-Row access network, while Figure 7-4 portrays a variation called **Middle-of-Row**. The latter decreases the average cable length, which can achieve substantial cost savings depending on the number of cabinets in each row.

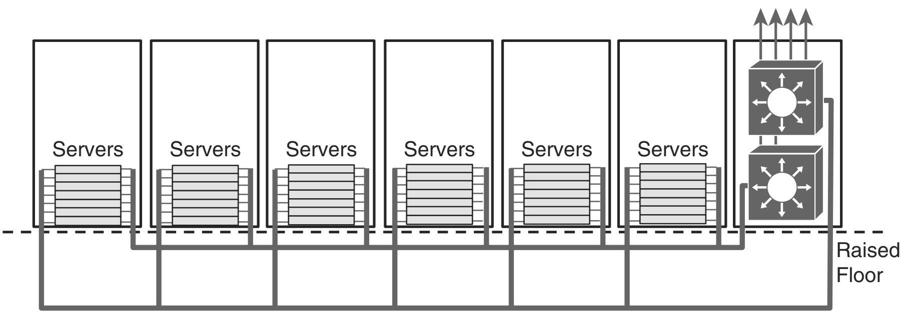


Figure 7-3 *End-of-Row*

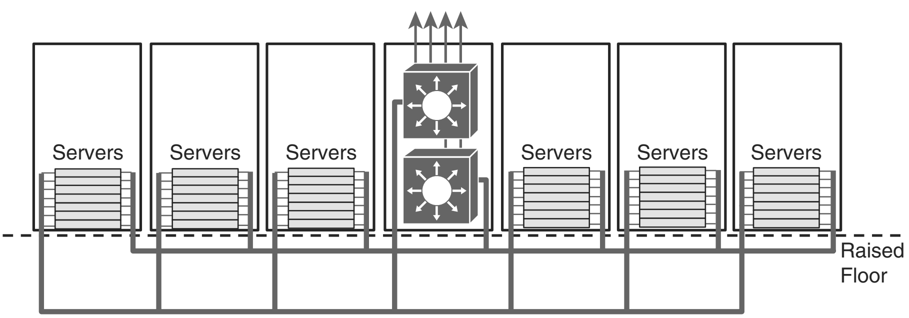


Figure 7-4 *Middle-of-Row*

EoR topologies are more flexible for low-density cabinets, because the horizontal cabling can potentially reach any rack in the row, leveraging the higher number of ports of modular chassis switches.

Cabling sprawl is the main disadvantage of EoR topologies. In a data center, excessive EoR cabling brings consequences such as

- Difficult cable management, troubleshooting, and decommission
- Blockage for air cooling (if it is installed underneath a raised access floor)

Data centers usually invest in structured cabling, preprovisioning a large part of the connections to avoid excessive cable installation. In these situations, EoR-based networks can only adopt new connectivity technologies if they were previously considered in the original cabling design.

Over the last decades, none of the access models were clearly defined as the "best" for cabling implementations. A design decision between ToR or EoR models depends on technical experience, knowledge about the server environment, and investment resources. However, best results should not be expected when there is little interaction between the facilities, network, and server teams.

Understanding Fabric Extenders

In 2009, Cisco launched the Nexus 2000 Fabric Extender series. Not conceived to be used as standalone Ethernet switches, these devices are remote linecards that are managed by a *parent* switch, such as a Nexus 5000, Nexus 6000, UCS Fabric Interconnect, or Nexus 7000 (with appropriate modules).

Fabric Extenders (or FEX) enable a data center to leverage the advantages from both ToR and EoR models because

- Multiple Fabric Extenders can be managed from a single parent switch (similar to EoR).
- A Fabric Extender can be installed inside a server cabinet and decrease cabling costs (similar to ToR).

Figure 7-5 illustrates an example of a server access topology that uses Fabric Extenders.

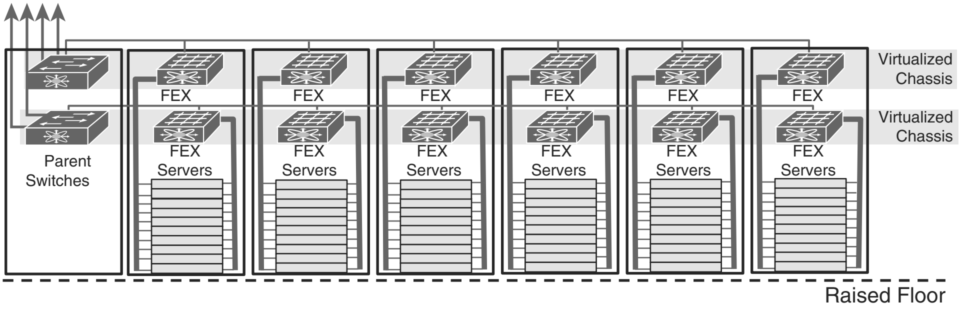


Figure 7-5 *Fabric Extender Topology*

As the figure suggests, a parent switch and multiple Fabric Extenders are elements of a virtualized modular chassis. Inside this virtual structure, every management operation is performed on the parent switch (which performs the role of a supervisor module of such chassis), and Ethernet frames are exchanged on the Fabric Extender interfaces (which represent the chassis interface modules).

The main endeavor of Fabric Extenders is to keep the configuration complexity within the parent switches and drive simplicity toward the server interfaces.

The Fabric Extender architecture introduces new types of ports to the network, including physical and virtual interfaces. Figure 7-6 deals with the interaction between the following FEX-related interfaces:

- **Fabric Interface (FIF):** This is a physical interface created to connect the Fabric Extender to a parent switch. It cannot be used for any other purpose.
- **Host Interface (HIF):** This is a standard physical Ethernet interface designed for server connection.
- **Logical Interface (LIF):** This is a data structure in the parent switch that emulates an Ethernet interface. It carries properties such as VLAN membership, access control list (ACL) labels, and STP states and is mapped to a virtual interface created on a Fabric Extender.
- **Virtual Interface (VIF):** This is a logical entity inside Fabric Extenders that receives its configuration from the parent switch, and it is used to map frames to a switch Logical Interface (LIF).

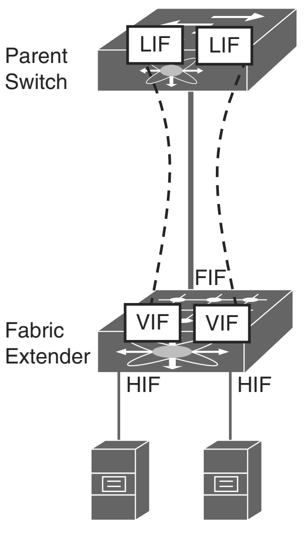


Figure 7-6 *Fabric Extender Interfaces*

The mapping between a parent switch LIF and a Fabric Extender VIF is called a *Virtual Network Link (VN-Link)*, and it is defined through a special tag that is inserted on all Ethernet frames that traverse these physical links. This extra header is called a *Virtual*

Network Tag (VNTag), and its main objective is to differentiate frames received from (or sent to) distinct FEX host interfaces.

Figure 7-7 depicts the 6-byte VNTag header implemented on Cisco Fabric Extenders.

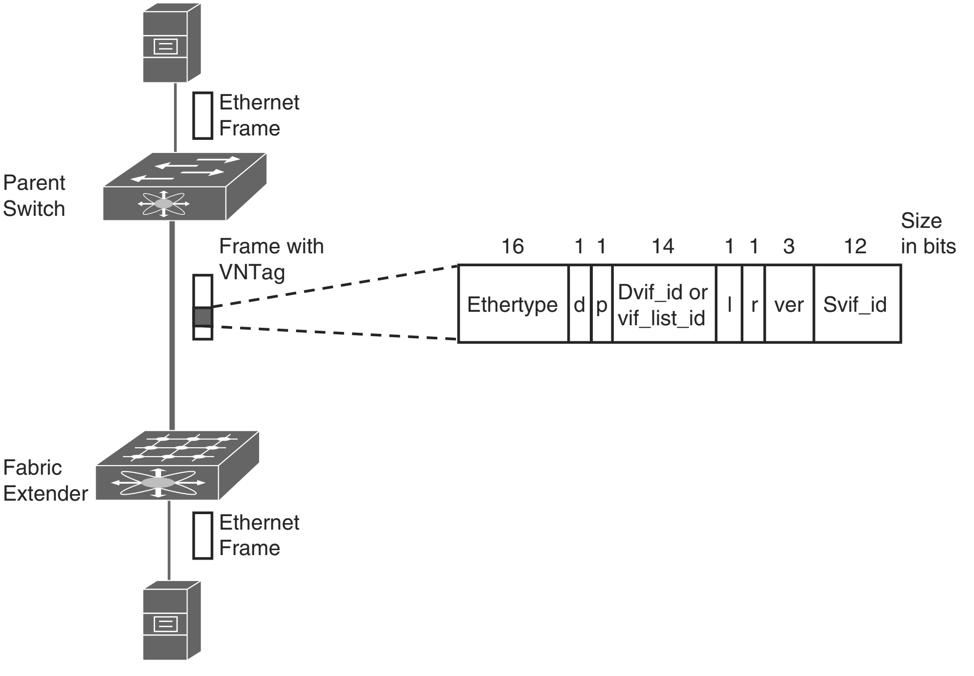


Figure 7-7 *Virtual Network Tag*

The VNTag is inserted between the source MAC address and the IEEE 802.1Q fields from the original Ethernet frame. The VNTag fields are

- **Ethertype:** This field identifies a VNTag frame. IEEE reserved the value 0x8926 for Cisco VNTag.
- **Direction bit (d):** A 0 indicates that the frame is traveling from the FEX to the parent switch. A 1 means that the frame is traveling from the parent switch to the FEX.
- **Pointer bit (p):** A 1 indicates that a Vif_list_id is included in the tag. A 0 signals that a Dvif_id is included in the frame.
- **Virtual Interface List Identifier (Vif_list_id):** This is a 14-bit value mapped to a list of host interfaces to which this frame must be forwarded.
- **Destination Virtual Interface Identifier (Dvif_id):** This is a 12-bit value mapped to a single host interface to which an Ethernet frame will be forwarded.

- **Looped bit (l):** This field indicates a multicast frame that was forwarded out the switch port and later received. In this case, the FEX checks the Svif_id and filters the frame from the corresponding port.
- **Reserved bit (r):** This bit is reserved for future use.
- **Version (ver):** This value is currently set to 0. It represents the version of the tag.
- **Source Virtual Interface Identifier (Svif_id):** This is a 12-bit value mapped to the host interface that received this frame (if it is going from the FEX to the parent switch).

When an Ethernet frame is received on a host interface,

The Fabric Extender adds a VNtag to the frame and forwards it to one of the fabric interfaces.

The parent switch recognizes the logical interface that sent the frame (through the Svif_id field), removes the tag, and forwards it according to its MAC address table.

Tip The parent switch MAC address table is updated if an unknown MAC address originates the frame received on an HIF. The new entry points to the logical interface index.

In the other direction (parent switch receives a frame that is destined to a FEX host interface),

The parent switch reads the frame destination MAC address and forwards it to a logical interface index in its MAC address table.

The switch inserts the VNtag associated with the logical interface and forwards it to the correct FEX.

Receiving this frame, the FEX recognizes the associated VIF (through the Dvif_id), removes the VNTag, and sends it to the mapped host interface.

From a data plane perspective, forwarding in Fabric Extender host interfaces completely depends on the parent switch. Consequently, a frame exchange between two host interfaces always traverses the parent switch, even if they are located in the same Fabric Extender.

Note Because of its hardware simplicity, Nexus 2000 Fabric Extenders have a very low latency (as low as 500 nanoseconds at the time of this writing) when compared to traditional Ethernet switches.

Multidestination VNTag frames (such as flooding, broadcast, and multicast) are characterized with the Pointer bit and Vif_list_id fields. When such a frame is being forwarded from the parent switch to the Fabric Extender, it is replicated inside the FEX to reduce the fabric interface traffic.

Note Cisco has been shipping VNTag as a prestandard port extension protocol since 2009. This additional header is intended to deliver the same capabilities defined in the IEEE 802.1BR (formerly IEEE 802.1Qbh) standard, which was published in July 2012. At the time of this writing, Cisco is expected to deliver products that fully support a VNTag and standards-based solutions.

Fabric Extender Options

Cisco has launched several Fabric Extender models since the Nexus 2148T was first shipped in 2009. Subsequent Cisco Fabric Extender products differ from each by factors such as

- **Fabric interfaces:** Most Cisco Fabric Extenders have four or eight 10 Gigabit Ethernet interfaces, which can use fiber (10GBASE-ER, 10GBASE-LR, Fabric Extender transceivers) or twinax cables. At the time of this writing, the Nexus 2248PQ exclusively supports 4 QSFP+ fabric interfaces that can deploy four 40-gigabit Ethernet connections or, using breakout cables, sixteen 10-gigabit Ethernet connections.
- **Host interfaces:** 8, 24, 32, or 48 Ethernet ports, which can be fixed (1000BASE-T, 100BASE-TX/1000BASE-T, 10GBASE-T) or SFP based (10GBASE-ER, 10GBASE-LR, twinax, 1000BASE-T).
- **Memory buffer:** Some models have larger shared buffers for applications such as large databases, shared storage, and video editing.
- **Form factor:** Fabric Extenders can be accommodated in server cabinets or in select blade server chassis (the latter option includes the UCS I/O Module).
- **Network capabilities:** These factors include multiple PortChannel members, number of quality of service (QoS) queues, and Fibre Channel over Ethernet (FCoE).
- **Choice of airflow:** Some models offer the choice to alternatively deploy front-to-back or back-to-front airflow.

A virtualized chassis can have different Fabric Extenders depending on server access requirements. This flexibility facilitates technology evolution, because migrations can be executed per server cabinet or even per server connection.

Devices as the Nexus 7000, Nexus 6000, Nexus 5000, and UCS Fabric Interconnect can act as parent switches to Fabric Extenders. Each one of these supports distinct Fabric Extender features and has different scalability characteristics.

Note For more details about Cisco Fabric Extenders models (and their parent switches), you can read Appendix A, "Cisco Data Center Portfolio." If you want to verify the capabilities supported on your hardware and software combination, also refer to the Cisco online documentation for the most recent information.

Connecting a Fabric Extender to a Parent Switch

When there is one active connection between them, a Fabric Extender and its parent switch use Satellite Discovery Protocol (SDP) periodic messages to discover each other. After this formal introduction, the Fabric Extender deploys a Satellite Registration Protocol (SRP) request to register itself to the parent switch.

Figure 7-8 illustrates the FEX discovery and registration process when a switch interface is configured with the **switchport mode fex-fabric** command.

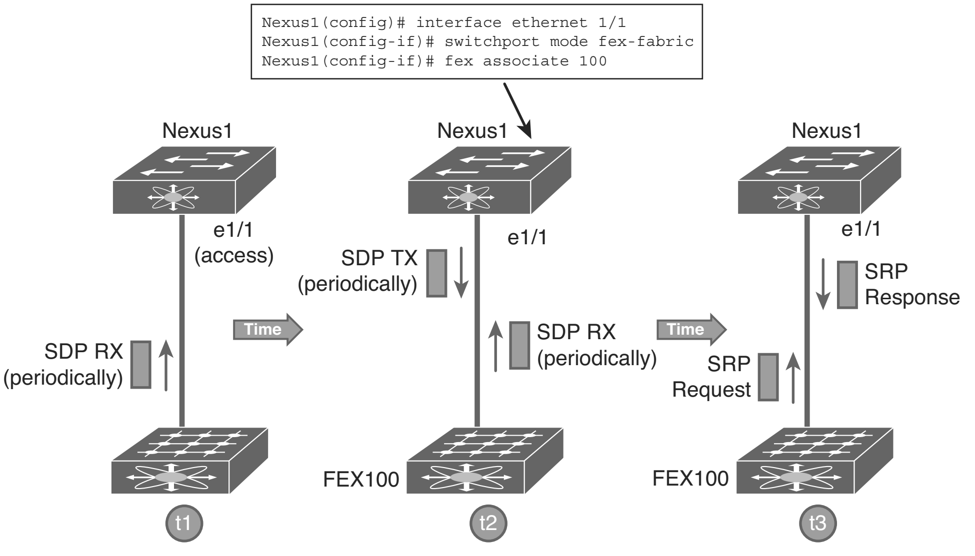


Figure 7-8 *Fabric Extender Discovery and Registration Process*

Example 7-1 shows a sample of the SDP messages sent from the Fabric Extender before any FEX configuration is executed in Nexus1 (at instant t1). As expected, the Fabric Extender automatically sends SDP messages as soon as it has an active fabric interface.

Example 7-1 *SDP Frames Sent at t1*

```
! Enabling the Fabric Extender processes in Nexus1
Nexus1# feature fex
! Enabling the tracing of SDP and SRP packets
Nexus1# debug fex pkt-trace
```

```
! Sample SDP Message received at interface Ethernet 1/1 (and repeated every 3
seconds)
08:35:03.086490 fex: Sdp-Rx: Interface: Eth1/1, Fex Id: 0, Ctrl Vntag: -1, Ctrl
Vlan: 1
08:35:03.086511 fex: Sdp-Rx: Refresh Intvl: 3000ms, Uid: 0x80a3e61d9cc8, device:
Fex, Remote link: 0x20000080
08:35:03.086523 fex: Sdp-Rx: Vendor: Cisco Systems  Model: N2K-C2232PP-10GE Serial:
JAF1509ECGT
[output suppressed]
```

In these SDP messages, the Fabric Extender exposes the VLAN from which it expects to receive control commands, SDP refresh interval (3 seconds), and hardware information.

After interface e1/1 in Nexus1 is configured with the **switchmode fex-fabric** and **fex associate** commands (instant t2 in Figure 7-8), this interface starts to send SDP packets, and both devices discover each other.

Example 7-2 details the Fabric Extender discovery process from the parent switch perspective.

Tip Fabric Extender identifiers must belong to the 100–199 range.

Example 7-2 *SDP Frames Sent After Ethernet 1/1 Configuration*

```
! SDP Messages sent from interface Ethernet 1/1 with Control VNTag -1  and Control
VLAN 4042
08:36:27.007428 fex: Sdp-Tx: Interface: Eth1/1, Fex Id: 100, Ctrl Vntag: -1, Ctrl
Vlan: 4042
08:36:27.007445 fex: Sdp-Tx: Refresh Intvl: 3000ms, Uid: 0xc03cc7730500, device:
Switch, Remote link: 0x1a000000
2012 Jun  4 08:36:27.007456 fex: Sdp-Tx: Vendor:  Model:  Serial: ----------
[output suppressed]
! SDP Messages sent from interface Ethernet 1/1 with new VNTag (zero)
08:36:27.101688 fex: Sdp-Tx: Interface: Eth1/1, Fex Id: 100, Ctrl Vntag: 0, Ctrl
Vlan: 4042
08:36:27.101704 fex: Sdp-Tx: Refresh Intvl: 3000ms, Uid: 0xc03cc7730500, device:
Switch, Remote link: 0x1a000000
08:36:27.101715 fex: Sdp-Tx: Vendor:  Model:  Serial: ----------
! Notice how the SDP Messages received from the FEX have changed
08:36:27.125135 fex: Sdp-Rx: Interface: Eth1/1, Fex Id: 0, Ctrl Vntag: 0, Ctrl Vlan:
4042
08:36:27.125153 fex: Sdp-Rx: Refresh Intvl: 3000ms, Uid: 0x80a3e61d9cc8, device:
Fex, Remote link: 0x20000080
08:36:27.125164 fex: Sdp-Rx: Vendor: Cisco Systems  Model: N2K-C2232PP-10GE Serial:
JAF1509ECGT
[output suppressed]
```

After the discovery is complete, the Fabric Extender sends an SRP Request message and waits for an SRP Response from the parent switch (instant t3). The registration process completes the FEX detection on the parent switch, and SDP messages continue to be exchanged between both devices after the registration.

Example 7-3 shows the registration process and describes how the Fabric Extender becomes operational for Nexus1.

Example 7-3 *SRP Registration and FEX Becoming Operational*

```
! FEX sens an SRP Request
08:36:30.114980 fex: Srp Req: Interface: Eth1/1, Uid: 0x80a3e61d9cc8, Card Id: 82,
IPC ver: 21
08:36:30.114992 fex: Srp Req: Version: 5.1(3)N2(1a), Interim Version: 5.1(3)N2(1a)
! Nexus1 sends an SRP Response confirming that FEX firmware is
supported . . .
08:36:30.116726 fex: Srp Resp: Interface: Eth1/1, Fex id: 100 Ver Chk: Compatible,
Img Uri:
! . . . and assigns an internal IP address for the FEX
08:36:30.116744 fex: Srp Resp: MTS addr: 0x2102, IP addr: 127.15.1.100/0 Switch MTS:
0x101, Switch Ip: 127.15.1.250
! After that the parent switch has a new remote linecard
08:26:16 5548P-3-174 %$ VDC-1 %$ %SATCTRL-FEX100-2-SOHMS_ENV_ERROR: FEX-100 Module
1: Check environment alarms.
[output suppressed]
08:36:33 5548P-3-174 %$ VDC-1 %$ %PFMA-2-FEX_STATUS: Fex 100 is online
```

From this moment on, the Fabric Extender is a remote linecard for the parent switch's supervisor module. Example 7-4 shows how the FEX interfaces can be seen and controlled on Nexus1.

Example 7-4 *Verifying the Fabric Extender Status*

```
! Verifying the operational Fabric Extenders
Nexus1# show module fex
FEX Mod Ports Card Type                          Model          Status.
--- --- ----- ------------------------------- -------------- -------
100 1   32    Fabric Extender 32x10GE + 8x10G N2K-C2232P-10GE present
[output suppressed]
! Verifying the available FEX interfaces
Nexus1# show interface brief
[output suppressed]
--------------------------------------------------------------------
Ethernet      VLAN    Type Mode   Status  Reason          Speed Port
Interface                                                       Ch #
--------------------------------------------------------------------
```

```
! Notice that all interfaces have the FEX ID and that interface e100/1/1 is down
Eth100/1/1    1   eth  access down    SFP not inserted 10G(D) --
Eth100/1/2    1   eth  access down    SFP not inserted 10G(D) --
[output suppressed]
Eth100/1/32   1   eth  access down    SFP not inserted 10G(D) --
```

Nexus1 logical interfaces (LIF) can be recognized through the FEX ID/slot/port index format. And because these interfaces are within Nexus1 configuration reach, its default configuration is applied to the FEX interfaces as well (**switchport mode access** and **no shutdown** commands, in this case).

With a control VLAN (4042) and an internal IP address (127.15.1.100), the Fabric Extender is configured through the Virtual Interface Configuration (VIC) protocol after its operationalization. Based on command/response messages, this protocol is also responsible for configuring the forwarding tables inside each FEX. The parent switch uses the VIC protocol to assign the Dvif_id to each virtual interface in the FEX and retrieve the FEX virtual interface list identifiers (Vif_list_id) for multidestination traffic.

Note Chapter 15, "Transcending the Rack," discusses a VNTag-based technology called Virtual Machine Fabric Extender (VM-FEX), where specialized adapters use the VIC protocol to dynamically request the creation, deletion, enabling, and disabling of logical interfaces in the parent switch.

Fabric Extended Interfaces and Spanning Tree Protocol

In the previous section, you learned that FEX host interfaces can be visualized as any other interface on the parent switch. These interfaces can also inherit various features from the same switch, justifying the concept of the virtualized modular chassis.

Example 7-5 illustrates the enablement of interface Ethernet 100/1/1, which is using a 1000BASE-T Small Form Pluggable (SFP) transceiver.

Example 7-5 *FEX Interface Configuration*

```
! Configuring interface Ethernet 100/1/1 to accept the GigabitEthernet SFP
Nexus1(config)# interface ethernet 100/1/1
Nexus1(config-if)# speed 1000
! Verifying the interface status
Nexus1(config-if)# show interface ethernet 100/1/1 brief

---------------------------------------------------------------------
Ethernet      VLAN    Type Mode   Status  Reason        Speed    Port
Interface                                                        Ch #
```

```
--------------------------------------------------------------------
Eth100/1/1    1       eth  access up      none           1000(D) --
```

Nevertheless, there are some differences on how a host interface behaves in the Spanning Tree Protocol context. In Example 7-6, you can spot three commands that expose this distinct behavior.

Example 7-6 *Fabric Extender Interface and Spanning Tree Protocol*

```
! Verifying the STP status on the Fabric Interface
Nexus1# show spanning-tree interface ethernet 1/1
No spanning tree information available for Ethernet1/1
! Verifying the STP status on FEX interface
Nexus1# show spanning-tree interface ethernet 100/1/1

Vlan             Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- ------------------------
VLAN0001         Desg FWD 4         128.1025 Edge P2p
! Verifying the BPDU features on the FEX interface
Nexus1# show spanning-tree interface e100/1/1 detail | include Bpdu
   Bpdu guard is enabled
   Bpdu filter is enabled by default
Nexus1# show spanning-tree interface e100/1/1 detail | include BPDU
   BPDU: sent 11, received 0
```

From the example, it is possible to infer that

- The interface connected to the Fabric Extender (Ethernet 1/1) does not participate in STP, because from the Nexus1 perspective, it is considered a "backplane connection."
- The host interfaces on the FEX are configured as RSTP edge point-to-point ports.

By default, host interfaces are configured with bridge protocol data unit (BPDU) filter and BPDU guard. These configurations mean, respectively, that the interfaces will not process any received BPDUs, and will actually be disabled if such a frame arrives at them. Nevertheless, as a safeguard mechanism, these interfaces do send some BPDUs when they are activated to avoid loops that can be caused with a mistakenly direct connection between two host interfaces.

As a conclusion, the parent switch controlling Fabric Extenders as linecards forms a virtualized chassis designed for host connections. In my opinion, a correct characterization for the virtualization technique detailed in this chapter would be *virtualized modular access switch*.

Caution Although Fabric Extenders are primarily designed for server connections, they also support switches connected to host interfaces as long as you disable STP on these devices. In consequence, to eliminate loops in redundant active connections, the switches must deploy a redundancy mechanism that is not STP dependent (Cisco Flex Link, for example).

Fabric Interfaces Redundancy

A Fabric Extender with multiple connections to a parent switch can have two different behaviors in the case of a fabric interface failure. By default, a FEX deploys an interface pinning policy to control which host interfaces are shut in the case of such a failure.

The objective of static pinning is to keep the exact amount of bandwidth at the host ports in the case of a fabric interface failure. Hence, for the remaining available host interfaces, there is no oversubscription increase.

Example 7-7 shows how a fabric interface (Ethernet 1/1) is "pinned" to an active host interface (Ethernet 100/1/1).

Example 7-7 *Static Pinning*

```
Nexus1# show fex 100 detail
FEX: 100 Description: FEX0100   state: Online
[output suppressed]
  Fabric interface state:
    Eth1/1 - Interface Up. State: Active
    Eth1/2 - Interface Up. State: Active
! Interface Ethernet 100/1/1 is active as long as Ethernet 1/1 is active
  Fex Port        State  Fabric Port
       Eth100/1/1    Up      Eth1/1
       Eth100/1/2  Down        None
[output suppressed]
       Eth100/1/32 Down        None
```

The example output exhibits the host interface status dependence whether or not its associated fabric interface is active. If Ethernet 1/1 fails, Ethernet 100/1/1 is automatically disabled, and hopefully, the connected server should activate a fault tolerance mechanism (NIC teaming).

Example 7-8 portrays the host interface behavior when the fabric interface is disabled. Notice that Ethernet 100/1/1 fails too, ignoring the fact that another fabric interface (Ethernet 1/2) is still active.

Example 7-8 *Failure in Static Pinning*

```
! Disabling fabric interface e1/1
Nexus1(config)# interface ethernet 1/1
Nexus1(config-if)# shutdown
! Vefifying e100/1/1 status
Nexus1(config-if)# show fex 100 detail
FEX: 100 Description: FEX0100   state: Online
[output suppressed]
Pinning-mode: static    Max-links: 1
[output suppressed]
  Fabric interface state:
    Eth1/1 - Interface Down. State: Configured
    Eth1/2 - Interface Up. State: Active
  Fex Port        State  Fabric Port
       Eth100/1/1  Down      Eth1/1
       Eth100/1/2  Down        None
[output suppressed]
       Eth100/1/32  Down        None
```

The **pinning max-link** command divides the host interfaces among a maximum number of fabric interfaces. With our 32-port Fabric Extender, if you issue the **pinning max-link 4** command, eight interfaces will be disabled if a fabric port fails.

In static pinning, host interface assignment follows the fabric interface order of configuration. However, if you reload the parent switch or execute the **fex pinning redistribute** command, the host interface groups will be reassigned to the fabric interface numerical order.

Table 7-2 illustrates this exact scenario, where the order of configuration for the fabric interfaces was Ethernet 1/3, Ethernet 1/2, Ethernet 1/4, and Ethernet 1/1.

Table 7-2 *Fabric Extender Pinning Example*

Parent Switch Interface	Associated Host Interfaces (Following Order of Configuration)	Associated Host Interfaces (After Reload or Redistribution)
e1/1	25 to 32	1 to 8
e1/2	9 to 16	9 to 16
e1/3	1 to 8	17 to 24
e1/4	17 to 24	25 to 32

> **Tip** Both the **pinning max-link** and **fex pinning redistribute** commands are disruptive to the Fabric Extender host interfaces.

Alternatively, the configuration of a PortChannel between the parent switch and the Fabric Extender guarantees that every host interface remains operational if a fabric interface fails. However, the remaining bandwidth to the parent switch will be shared by all host ports (increasing the oversubscription).

Example 7-9 presents the results from a PortChannel configuration on the fabric interfaces (shown later in Figure 7-9).

Example 7-9 *PortChannel Configuration*

```
! Verifying how FEX 100 uses PortChannel 100 as its fabric interface
Nexus1# show fex 100 detail
FEX: 100 Description: FEX0100   state: Online
[output suppressed]
! When using PortChannel Max-links must be one
Pinning-mode: static    Max-links: 1
 [output suppressed]
  Fabric interface state:
    Po100 - Interface Up. State: Active
    Eth1/1 - Interface Up. State: Active
    Eth1/2 - Interface Up. State: Active
  Fex Port        State  Fabric Port
       Eth100/1/1    Up       Po100
       Eth100/1/2  Down        None
 [output suppressed]
       Eth100/1/32 Down        None
```

When fabric interfaces are aggregated, the max-link pinning must be set to 1 (because it is not possible to have more than one upstream PortChannel per FEX), and the interface must be configured in mode ON because Fabric Extenders do not support Link Aggregation Control Protocol (LACP).

Example 7-10 demonstrates that a failure in Ethernet 1/1 does not have any effect over interface Ethernet 100/1/1 when the fabric interfaces are aggregated. As expected, PortChannel 100 remains operational if a single member interface is active (Ethernet 1/2).

Example 7-10 *Failure in Ethernet 1/1*

```
! Disabling fabric interface e1/1
Nexus1(config)# interface ethernet 1/1
Nexus1(config-if)# shutdown
```

```
! Vefifying e100/1/1 status
Nexus1(config-if)# show fex 100 detail
FEX: 100 Description: FEX0100   state: Online
[output suppressed]
Pinning-mode: static    Max-links: 1
 [output suppressed]
  Fabric interface state:
    Po100 - Interface Up. State: Active
    Eth1/1 - Interface Down. State: Configured
    Eth1/2 - Interface Up. State: Active
! PortChannel is still operational
  Fex Port        State  Fabric Port
       Eth100/1/1    Up       Po100
       Eth100/1/2  Down        None
 [output suppressed]
       Eth100/1/32 Down        None
```

Figure 7-9 exhibits the behavior for both fabric interface redundancy options (static pinning and PortChannel) and details the configuration required in each one.

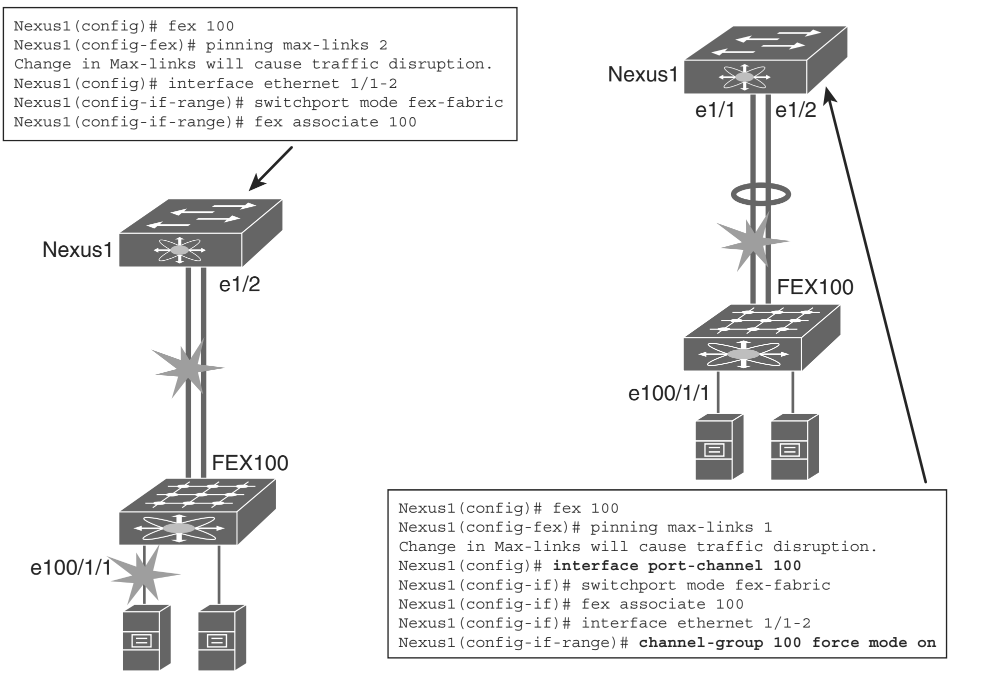


Figure 7-9 *Static Pinning Versus PortChannel: Configuration and Host Interface Behavior*

Fabric Extender Topologies

In the previous section, you were presented with simple scenarios that demonstrated basic functionalities of a Fabric Extender connected to one parent switch. Nevertheless, it is paramount that you understand the principles behind highly available FEX topologies.

There are basically two classes of topologies that provide fault tolerance in Fabric Extender designs:

- **Straight-through:** Where a Fabric Extender is connected to a single parent switch
- **Dual-homed:** Where a Fabric Extender is connected to a pair of parent switches

An example of each of these topologies is presented in Figure 7-10.

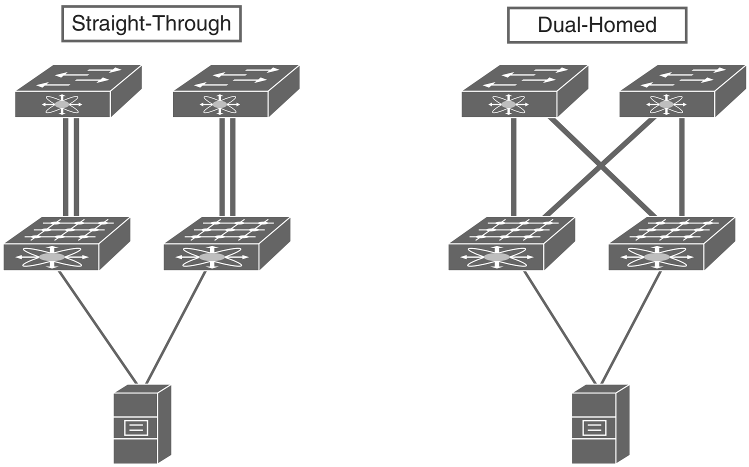


Figure 7-10 *Fabric Extender Topology Types*

> **Note** In the next sections, I will discuss topologies whose support varies depending on the parent switch and Fabric Extender hardware models and software versions. Because of the NX-OS constant state of evolution, I will avoid transcribing a matrix of supported topologies that will be quickly outdated. I highly recommend that you refer to the Cisco online documentation for the most recently supported FEX topologies.

Straight-Through Topologies

In straight-through topologies, it is recommended that each host has interfaces connected to Fabric Extenders that are managed by distinct parent switches. This practice avoids the total of loss of connectivity for a server in case of a switch failure.

Straight-through designs create a pair of NX-OS virtualized chassis with a single supervisor module in each. Likewise, IEEE 802.3ad–compatible servers can leverage active-active connections using virtual PortChannels (vPCs).

Figure 7-11 demonstrates how vPCs can be deployed in a straight-through topology comprised of two switches (Nexus1 and Nexus2) and two Fabric Extenders (FEX110 and FEX120). In the figure, each Fabric Extender uses two aggregated fabric connections to its parent switch. Both server interfaces are connected to access interfaces in VLAN 50 (e110/1/1 in Nexus1 and e120/1/1 in Nexus2) which will be vPC 10 member ports.

> **Tip** Notice that Nexus1 is configured to be the primary vPC peer, using the **role priority 1** command.

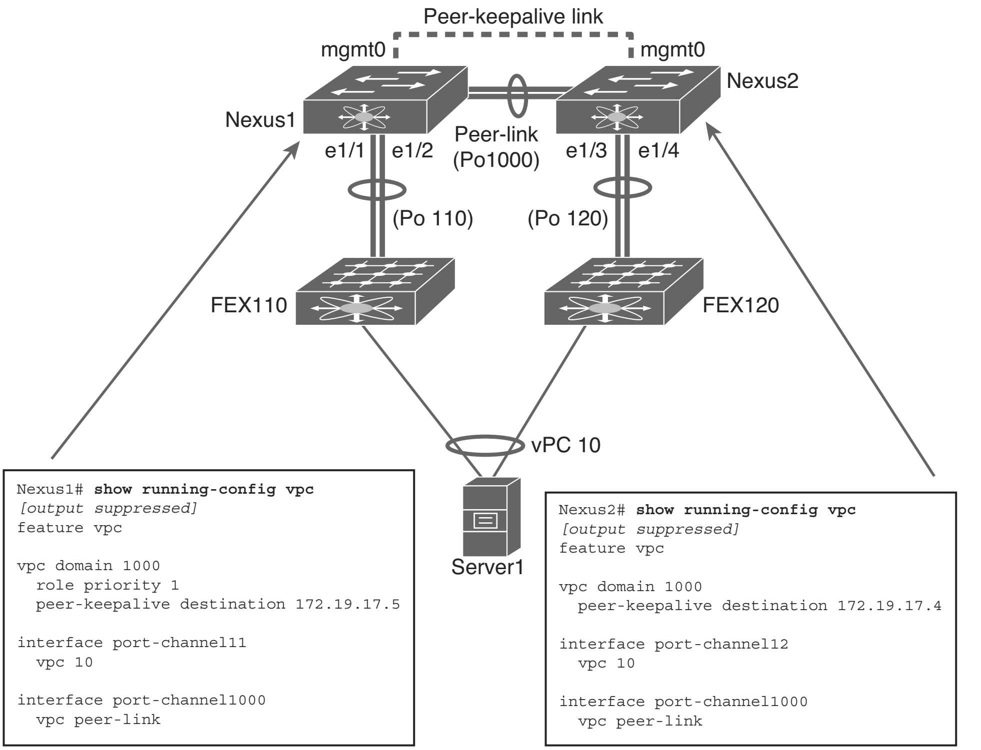


Figure 7-11 *Virtual PortChannel Straight-Through Topology*

For clarification, the PortChannel configuration options in this topology are explained in Table 7-3.

Table 7-3 *PortChannel Options*

Switch	PortChannel ID	Interfaces	Remote Peer	LACP
Nexus1	11 (vPC 10)	e110/1/1	Server A	Depends on peer
Nexus1	110	e1/1, e1/2	FEX110	No
Nexus1	1000	e1/31, e1/32	Nexus2	Depends on peer
Nexus2	12 (vPC 10)	e120/1/1	Server A	Depends on peer
Nexus2	120	e1/3, e1/4	FEX120	No
Nexus2	1000	e1/31, e1/32	Nexus1	Depends on peer

Note PortChannels 110 and 120 are not mandatory for this configuration because static pinning would work as well.

In Examples 7-11 and 7-12, you can verify the status of the virtual PortChannel created to aggregate Server1 Ethernet interfaces.

Example 7-11 *Virtual PortChannel 10 Status in Nexus1*

```
! Verifying vPC 10 status
Nexus1# show vpc 10
vPC status
----------------------------------------------------------------------
id     Port        Status Consistency Reason                     Active vlans
------ ----------- ------ ----------- -------------------------- ----
10     Po10        up     success     success                    50
```

Example 7-12 *Virtual PortChannel Status in Nexus2*

```
! Verifying vPC 10 status
Nexus1# show vpc 10
vPC status
----------------------------------------------------------------------
id     Port        Status Consistency Reason                     Active vlans
------ ----------- ------ ----------- -------------------------- ----
10     Po10        up     success     success                    50
```

In a straight-through vPC topology, the consequences of peer link total failure are the same as expected in a standard vPC configuration: All vPC member ports in the secondary vPC peer are automatically disabled. Example 7-13 exhibits this situation in Nexus2, when PortChannel 1000 is disabled on Nexus1 (through a **shutdown** command in the PortChannel).

Example 7-13 *Peer Link Effect on Nexus2*

```
! After the eer-link PortChannel is shut down in Nexus1, a peer-link failure
detected in Nexus2
Nexus2#
17:02:51 Nexus2 %$ VDC-1 %$ %VPC-2-VPC_SUSP_ALL_VPC: Peer-link going down, suspending
all vPCs on secondary
! Inspecting which interfaces are down due to peer-link failure
Nexus2# show interface brief | include vpc
Eth120/1/1    50    eth  access down  vpc peerlink is down 1000(D) 10
! Verifying FEX status in Nexus2
Nexus2# show fex
  FEX         FEX            FEX              FEX
Number    Description      State            Model            Serial
---------------------------------------------------------------------
120        FEX0120         Online    N2K-C2232PP-10GE   SSI150606H2
```

As expected, even though FEX120 is still operational, only vPC member ports are disabled because of the peer link failure.

Orphan ports, connected only to the secondary peer, can also be isolated in straight-through vPC topologies. The same recommendations I have described in Chapter 6, “Fooling Spanning Tree,” apply for these interfaces.

Figure 7-12 depicts other possible straight-through topologies available at the time of this writing.

Tip I usually recommend straight-through topologies for scenarios that require higher access port scalability (because they can deploy a twofold increase in the number of Fabric Extenders that a single parent switch supports).

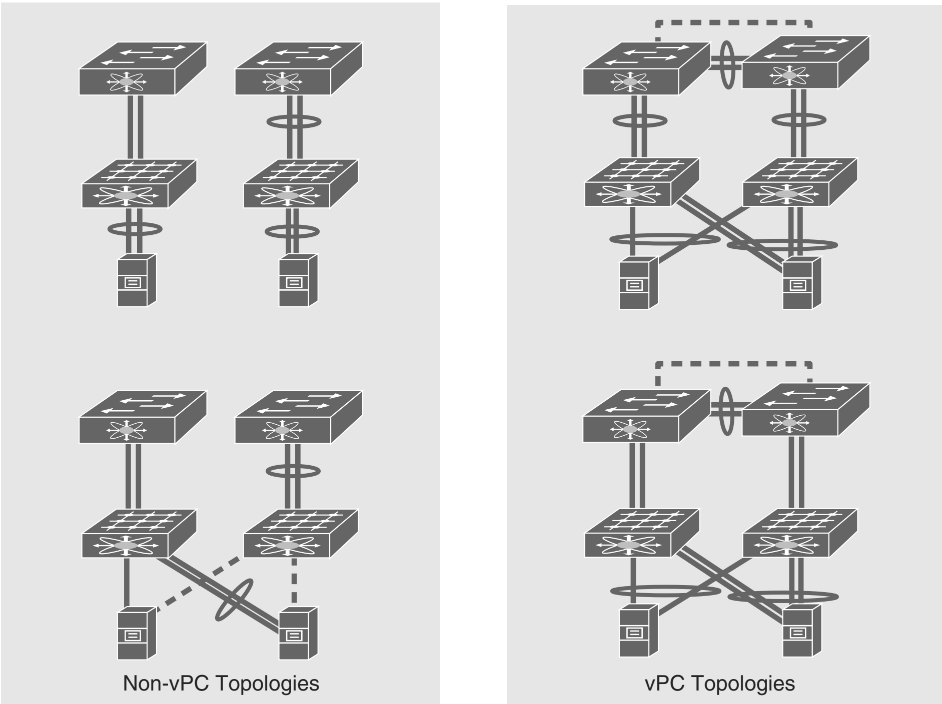


Figure 7-12 *Straight-Through Topologies*

Dual-Homed Topologies

In a classical modular Ethernet switch, redundant supervisor modules can control the installed linecards. In a virtualized access switch, the same redundancy can be achieved with the connection of two parent switches to a single Fabric Extender. However, you should not expect that the straightforward FEX configuration is sufficient to build a dual-homed topology: By default, a Fabric Extender can only be managed by a single parent switch.

To illustrate this behavior, consider the topology and configuration detailed in Figure 7-13. There, the Fabric Extender is connected and registered to Nexus1, and afterward, connected to Nexus2.

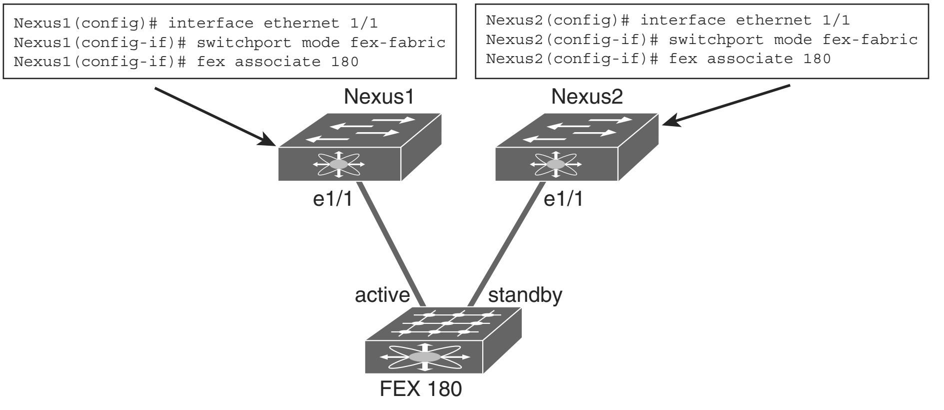


Figure 7-13 *Active-Standby Dual-Homed Topology*

Examples 7-14 and 7-15 demonstrate how each switch detects FEX180.

Example 7-14 *Fabric Extender Connected on Two Parent Switches*

```
! Verifying FEX status in Nexus1
Nexus1# show fex
  FEX         FEX           FEX               FEX
Number    Description      State             Model            Serial
----------------------------------------------------------------------
180         FEX0180         Online    N2K-C2232PP-10GE   SSI150606J3
```

Example 7-15 *Single FEX Connected on Two Parent Switches*

```
! Verifying FEX status in Nexus2
Nexus2# show fex
  FEX         FEX           FEX               FEX
Number    Description      State             Model            Serial
----------------------------------------------------------------------
---        --------      Connected    N2K-C2232PP-10GE   SSI150606J3
```

The Fabric Extender remains in the **Connected** state (and not **Online**) in Nexus2 because it is already registered to Nexus1. Therefore, Nexus2 and the Fabric Extender only discover each other, without advancing to the registration phase.

If the connection to Nexus1 fails, the Fabric Extender registers itself to Nexus2 and remains in that state (even if the connection to Nexus1 is reactivated).

However, after the parent switch transition, the host interfaces would remain unconfigured. The reason is simple: Before the FEX is completely operational on Nexus2, it is not possible to configure any host interfaces because they simply do not exist yet!

Example 7-16 shows what happens if you try to configure a host interface in Nexus2 before FEX180 is online. The example also details how you can enable *Fabric Extender preprovisioning* to fix this behavior and effectively provide active-standby parent switches to a Fabric Extender.

Example 7-16 *Fabric Extender Preprovisioning*

```
! Trying to configure a host interface in FEX180 (which is not operational in Nexus2)
Nexus2(config)# interface ethernet 180/1/1
                                        ^

Invalid range at '^' marker.
! Pre-provisioning FEX180 in Nexus2
Nexus2(config)# slot 180
Nexus2(config-slot)# provision model N2K-C2232P
! Now you can pre-configure a host interface in FEX180
Nexus2(config-slot)# interface ethernet 180/1/1
Nexus2(config-if)# speed 1000
```

Tip The preprovisioning configuration must be done on both switches if you want the host interfaces to be enabled after a switchover from any parent switch to the other.

From a control plane perspective, it might be enough that a single parent switch manages a Fabric Extender while the other waits for a failure. Nevertheless, there are two drawbacks in an active-standby dual-homed topology that challenge its application in real-world scenarios:

- The connection to the standby parent switch is not used for data traffic.
- The transition from one parent switch to the other must wait almost 40 seconds before the Fabric Extender is online.

Figures 7-14 clarifies how virtual PortChannels can overcome both drawbacks and enable active-active dual-homed topologies for Fabric Extenders.

Example 7-17 depicts the vPC configuration in Nexus1. The same configuration was issued in Nexus2, except for the peer keepalive destination IP address that was pointed to 172.19.17.4 (Nexus1 mgmt0 interface).

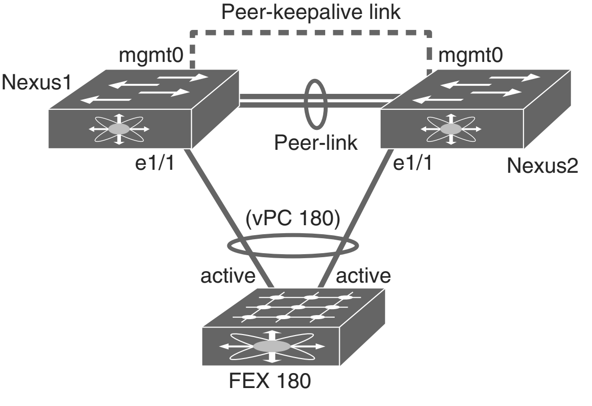


Figure 7-14 *Active-Active Dual-Homed Fabric Extender*

Example 7-17 *Nexus1 vPC Configuration*

```
! Observing all vPC configuration parameters in Nexus1
Nexus1# show running-config vpc
[output suppressed]

feature vpc
! Step 1: Configuring vPC domain
vpc domain 1000
  role priority 1
! Step 2: Configuing peer-keepalive link (IP address 172.19.17.4 in Nexus2)
  peer-keepalive destination 172.19.17.5
[output suppressed]
! Step 3: Configuring peer-link (interfaces e1/31 and e1/32)
Nexus1# show running-config interface port-channel 1000, port-channel 180
 [output suppressed]
interface port-channel1000
  switchport mode trunk
  spanning-tree port type network
  vpc peer-link
! Step 4: Configuring and reusing PortChannels (interface e1/1 on both switches)
interface port-channel180
  switchport mode fex-fabric
  fex associate 180
  vpc 180
```

In active-active dual-homed scenarios, each Fabric Extender is online for both parent switches, and Examples 7-18 and 7-19 portray this status.

Example 7-18 *Verifying Fabric Extender Status in Nexus1*

```
Nexus1# show fex 180 detail
FEX: 180 Description: FEX0180   state: Online
[output suppressed]
  Fabric interface state:
    Po180 - Interface Up. State: Active
    Eth1/1 - Interface Up. State: Active
  Fex Port        State  Fabric Port
! Interface e180/1/1 is configured and operational in Nexus1
       Eth180/1/1    Up       Po180
       Eth180/1/2  Down        None
[output suppressed]
```

Example 7-19 *Verifying Fabric Extender Status in Nexus2*

```
Nexus2# show fex 180 detail
FEX: 180 Description: FEX0180   state: Online
[output suppressed]
  Fabric interface state:
    Po180 - Interface Up. State: Active
    Eth1/1 - Interface Up. State: Active
  Fex Port        State  Fabric Port
! Interface e180/1/1 is configured and operational in Nexus2
       Eth180/1/1    Up       Po180
       Eth180/1/2  Down        None
 [output suppressed]
```

In active-active topologies, a parent switch failure does not affect the host interfaces on the Fabric Extender because both vPC peers manage it simultaneously. However, it is a requirement that the FEX configuration (including the host interfaces) is the same on both switches.

Tip You can use the *configuration synchronization* NX-OS feature to replicate selected parts of a configuration to a peer switch (if it is available in your software and hardware combination).

I usually recommend active-active dual-homed topologies for scenarios that require minimum failure effects on server connectivity (such as servers that have only one Ethernet connection, for example).

During a period, it was not possible to deploy vPCs on host interfaces connected to dual-homed active-active Fabric Extenders. Notwithstanding, Enhanced virtual PortChannel (EvPC) capability surpassed that limitation, maintaining a simple configuration principle: Because in active-active dual-homed topologies, the host interfaces are configurable on both parent switches, you only need to deploy a PortChannel for the server interfaces you want to aggregate.

Figure 7-15 depicts the EvPC scenario derived from our topology with the inclusion of another Fabric Extender (FEX190).

> **Note** Consider that the FEX190 configuration on both switches is exactly the same as the one issued for FEX180.

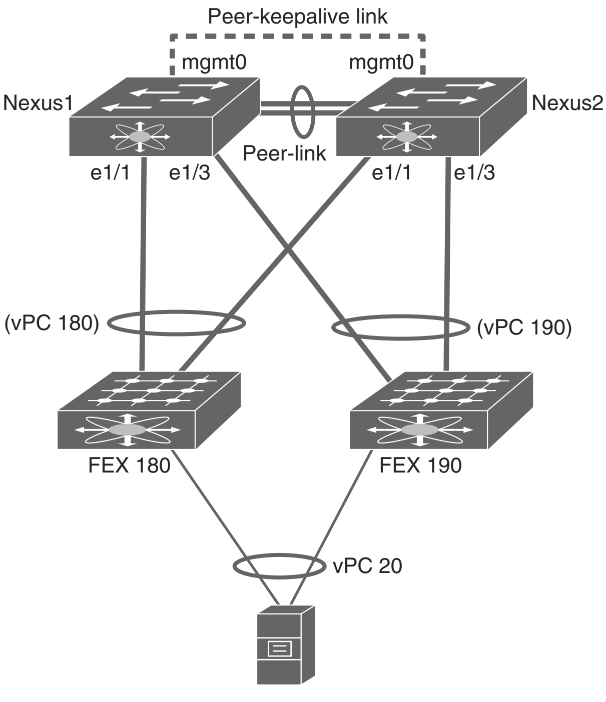


Figure 7-15 *Enhanced vPC*

The EvPC configuration on Nexus1 is detailed in Example 7-20. As mentioned previously, Nexus2 received the same configuration.

Example 7-20 *Enhanced vPC Configuration in Nexus1*

```
Nexus1# show running-config interface port-channel 20
[output suppressed]
! Configuring the EvPC
interface port-channel20
  switchport access vlan 50

Nexus1# show running-config interface ethernet 180/1/1, ethernet 190/1/1
[output suppressed]
interface Ethernet180/1/1
  switchport access vlan 50
  speed 1000
! Including Ethernet 180/1/1 in the EvPC
  channel-group 20 mode active

interface Ethernet190/1/1
  switchport access vlan 50
  speed 1000
! Including Ethernet 190/1/1 in the EvPC
  channel-group 20 mode active
```

Use Case: Mixed Access Data Center

After your imaginary career achieves the sequential successes described in previous chapters, you are requested to design the server access network of a midrange data center.

This data center is acquiring 160 new rack-mountable servers with two 10 Gigabit Ethernet interfaces (for redundancy), where each server consumes approximately 450 watts. But the design must also support 96 legacy servers that will be decommissioned over the next two years. These old servers have only one Gigabit Ethernet interface and very similar power requirements (although only one-third of the performance).

This company's CIO requires that three principles orient your project: scalability, high availability, and physical optimization. He also presents two boundary conditions:

- Only 7.5 kilowatts can be provided per server cabinet.
- The network cannot have an oversubscription higher than 5:1, even in moments of a connection failure.

Believing that an FEX-based virtualized chassis can be customized to support the specific requirements of most server environments, you present the design depicted in Figure 7-16.

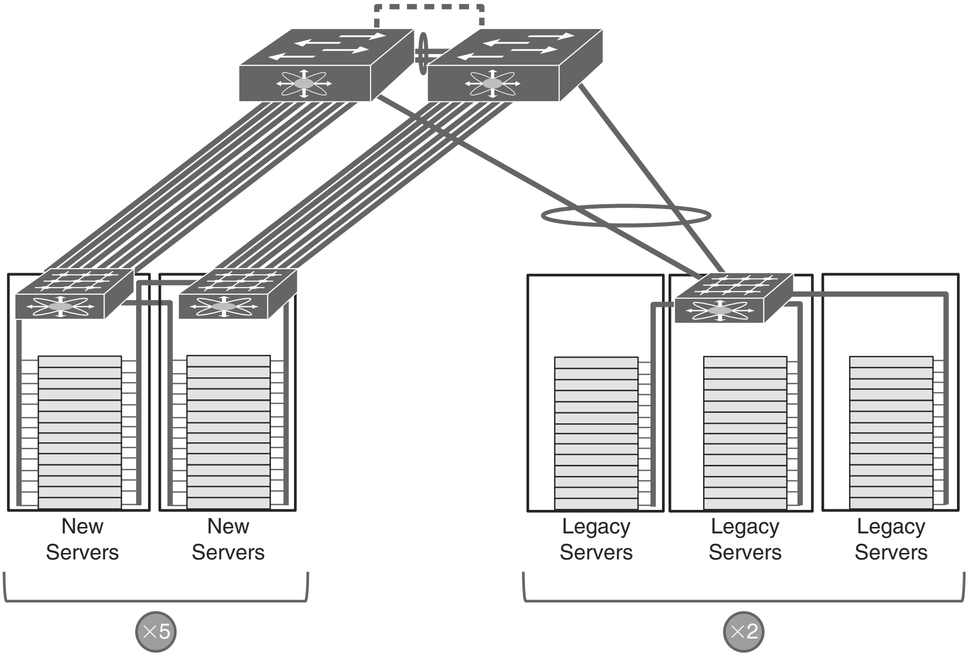


Figure 7-16 *Mixed Access Data Center*

In your design, each server cabinet can accommodate up to 16 servers because these devices consumes together 7.2 kW (leaving 300 W for the Fabric Extender). A pair of Nexus 2232PP Fabric Extenders (with eight 10 Gigabit Ethernet fabric interfaces and 32 10 Gigabit Ethernet host interfaces) can support and optimize the cabling of two server cabinets.

Within these cabinets, each new server is connected to two different Nexus 2232PP Fabric Extenders in a straight-through topology. The fabric interfaces do not use PortChannels to maintain the bandwidth oversubscription of 4:1 if there is a connection failure. Optionally, these servers can leverage vPCs to deploy 20 Gbps of bandwidth with aggregated interfaces.

For the legacy servers, you also consider 16 servers in each cabinet. However, you select one Nexus 2248TP-E Fabric Extender (with four 10 Gigabit Ethernet fabric interfaces and 48 10 Gigabit Ethernet host interfaces) to provide connectivity to three server cabinets and increase high availability through an active-active dual-homed topology. This design assures the network team that, even in the case of a fabric interface failure, the host interface's maximum oversubscription is 4.8:1.

Next, you select a pair of Layer 3 Nexus switches capable of deploying both FEX topologies and acting as the default gateway for the servers. The team is happy to know that only two virtualized switches will be able to manage their entire data center.

During lunch, they reveal that you have just designed their backup data center for the next three years. And later, you are invited to lead their main data center project.

Summary

In this chapter, you have learned that

- Both Top-of-Rack and End-of-Row server access models have advantages and shortcomings. Fabric Extenders leverage the best of both models (cabling optimization and consolidated management, respectively).
- A parent switch connected to one or more Fabric Extenders forms a virtualized access chassis.
- Parent switches and Fabric Extenders use VNTag to deploy virtual interfaces for each FEX host interface.
- FEX host interfaces are RSTP edge ports with BPDU filter and BPDU guard enabled.
- Fabric Extenders can have fabric interface redundancy through static pinning or PortChannels.
- There are two classes of Fabric Extender redundant topologies: straight-through and dual-homed.
- It is possible to provide link aggregation to servers connected to Fabric Extenders in straight-through topologies (standard vPC) and dual-homed topologies (Enhanced vPC).

Figure 7-17 graphically summarizes how Fabric Extenders (connected to a parent switch) can emulate a virtualized modular chassis for a server connection.

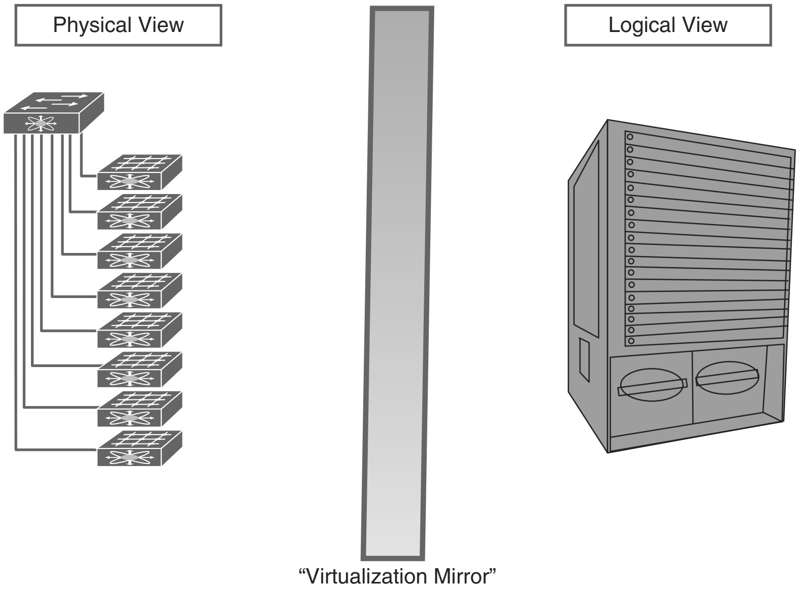


Figure 7-17 *Through the Virtualization Mirror*

Further Reading

Gai, Silvano and DeSanti, Claudio. *I/O Consolidation in the Data Center*. Cisco Press, 2009.

VNTag 101

www.ieee802.org/1/files/public/docs2009/new-pelissier-vntag-seminar-0508.pdf

White Paper: Virtual Machine Networking: Standards and Solutions

www.cisco.com/en/US/prod/collateral/switches/ps9441/ps9902/whitepaper_c11-620065.pdf

IEEE Standards Association

standards.ieee.org/getieee802/download/802.1BR-2012.pdf

This page intentionally left blank

Index

NUMERICS

A

B

C

D

F

G

I

M

N

O

P

Q

R

S

T

U

W

X

Z