Chapter 1

Introduction to Cisco NX-OS

This chapter provides an introduction and overview of NX-OS and a comparison between traditional IOS and NX-OS configurations and terminology. The following sections will be covered in this chapter:

- NX-OS overview
- NX-OS user modes
- Management interfaces
- Managing system files

NX-OS Overview

Cisco built the next-generation data center class operating system designed for maximum scalability and application availability. The NX-OS data center class operating system was built with modularity, resiliency, and serviceability at its foundation. NX-OS is based on the industry-proven Cisco Storage Area Network Operating System (SAN-OS) Software and helps ensure continuous availability to set the standard for mission-critical data center environments. The self-healing and highly modular design of Cisco NX-OS enables for operational excellence, increasing the service levels and enabling exceptional operational flexibility. Several advantages of Cisco NX-OS include the following:

- Unified data center operating system
- Robust and rich feature set with a variety of Cisco innovations
- Flexibility and scalability
- Modularity
- Virtualization
- Resiliency

- IPv4 and IPv6 IP routing and multicast features
- Comprehensive security, availability, serviceability, and management features

Key features and benefits of NX-OS include

- **Virtual device contexts (VDCs):** Cisco Nexus 7000 Series switches can be segmented into virtual devices based on customer requirements. VDCs offer several benefits such as fault isolation, administration plane, separation of data traffic, and enhanced security.
- **Virtual Port Channels (vPCs):** Enables a server or switch to use an EtherChannel across two upstream switches without an STP-blocked port to enable use of all available uplink bandwidth.
- **Continuous system operation:** Maintenance, upgrades, and software certification can be performed without service interruptions because of the modular nature of NX-OS and features such as In-Service Software Upgrade (ISSU) and the capability for processes to restart dynamically.
- **Security:** Cisco NX-OS provides outstanding data confidentiality and integrity, supporting standard IEEE 802.1AE link-layer cryptography with 128-bit Advanced Encryption Standard (AES) cryptography. In addition to CTS, there are many additional security features such as access control lists (ACLs) and port-security, for example.
- **Overlay Transport Virtualization (OTV):** Enables the Layer 2 extension between distributed data centers over any transport Layer 3 network.
- **NX-OS Persistent Storage Service (PSS):** The PSS is a lightweight database that maintains runtime information state. PSS provides reliable persistent storage to the software components to *checkpoint* their internal state and data structures enabling nondisruptive restart. If a fault occurs in a process (such as OSPF), the NX-OS high-availability (HA) manager determines best recovery action:
  - Restart a process.
  - Switch over to a redundant supervisor module.

Note The process restart does not have any impact in the data plane operations; the total control plane recovery is approximately 10 milliseconds.

- **FabricPath:** Enables each device to build an overall view of the topology; this is similar to other link state routing protocols. Each device in the FabricPath topology is identified by a switch-id. The Layer 2 forwarding tables are built based on reachability to each switch-id, not by the MAC address. Eliminates spanning-tree to maximize network bandwidth and flexibility in topological configurations, as well as simplify operational support and configuration. This enables a tremendous amount of flexibility on the topology because you can now build FabricPath topologies for Layer 2-based networks the same as for Layer 3-based networks.

NX-OS Supported Platforms

An NX-OS data center-class operating system, designed for maximum scalability and application availability, has a wide variety of platform support, including the following:

- **Nexus 7000:** Provides an end-to-end data center architecture on a single platform, including data center core, data center aggregation, and data center access layer. The data center access layer could be end-of-row or top-of-rack or a combination of end-of-row and top-of-rack with a Fabric Extender (FEX). Depending on the requirements, the Nexus 7000 has many different form factors; the form factors include the following (note that all the chassis share common supervisor modules, I/O modules, NX-OS software, and power supplies):
  - **Nexus 7018:** An 18-slot chassis that supports 16 I/O modules. Slots 9 and slot 10 are reserved for supervisor modules on the Nexus 7018 chassis. The I/O module slots for the Nexus 7018 chassis are reserved 1 through 8 and 11 through 18. The supervisor module slots (9 and 10) can have only a supervisor module installed in them; I/O modules will not work in the supervisor slots. All I/O module slots have full fabric connections of up to 230 Gbps with Fabric-1 installed or 550 Gbps with Fabric-2 installed. The fabric bandwidth depends on the number of fabric modules installed and the I/O modules installed in any given I/O module slot; the Nexus 7018 chassis is side-to-side airflow.
  - **Nexus 7010:** A 10-slot chassis that supports 8 I/O modules. Slot 5 and slot 6 are reserved for supervisor modules on the Nexus 7010 chassis. Slot 1 through slot 4 and slot 7 through slot 10 are reserved for I/O modules on the Nexus 7010 chassis. The supervisor module slots (5 and 6) can have only a supervisor module installed in them; I/O modules do not work in the supervisor slots. The Nexus 7010 is front-to-back-airflow, to meet hot isle or cold isle data center design. All I/O module slots have full fabric connections of up to 230 Gbps with Fabric-1 installed or 550 Gbps with Fabric-2 installed. The fabric bandwidth depends on the number of fabric modules installed and the I/O modules installed in any given I/O module slot
  - **Nexus 7009:** A 9-slot chassis that supports 7 I/O modules. The Nexus 7009 chassis I/O modules have a horizontal orientation of the line modules and side-to-side airflow. Slot 1 and slot 2 are reserved for supervisor modules in the Nexus 7009 chassis. Slots 3 through 9 are reserved for I/O modules on the Nexus 7009 chassis. The supervisor module slots (1 and 2) can have only a supervisor module installed in them; I/O modules will not work in the supervisor slots. The Nexus 7009 chassis is side-to-side airflow. All I/O module slots have full fabric connections of up to 550 Gbps with Fabric-2 installed. The fabric bandwidth depends on the number of fabric modules installed and the I/O modules installed in any given I/O module slot.
  - **Nexus 7004:** A four-slot chassis that supports two I/O modules. The Nexus 7004 chassis I/O modules have a horizontal orientation of the line modules and side-to-rear airflow. Slot 1 and slot 2 are reserved for supervisor modules in the Nexus

7004 chassis. Slot 3 and slot 4 are reserved for I/O modules on the Nexus 7004 chassis. The supervisor module slots (1 and 2) can have only a supervisor module installed in them; I/O modules do not work in the supervisor slots. The 7004 does not have fabric modules. The I/O modules installed in a 7004 chassis use one of the fabric connections for communications between the modules.

Note The Nexus 7004 chassis is supported only with the Supervisor 2 and Supervisor 2e. In addition, the Nexus 7004 supports M1-XL, F2, M2, and F2e I/O modules.

- **Nexus 5000:** Ideal for the data center server access layer providing architectural support for virtualization and Unified Fabric Environments while maintaining consistent operational models.
  - **Nexus 5010:** Twenty fixed wire-speed 10-Gigabit Ethernet interfaces that support IEEE data center bridging (DCB) and FCoE. In addition to the fixed interfaces, the Nexus 5010 has one expansion module. The expansion module supports Native Fibre Channel, Ethernet, and FCoE interfaces. The first eight interfaces of the Nexus 5010 support 1 GbE and 10 GbE.
  - **Nexus 5020:** Forty fixed wire-speed 10-Gigabit Ethernet interfaces that support IEEE DCB and FCoE. In addition to the fixed interfaces, the Nexus 5010 has one expansion module. The expansion module supports Native Fibre Channel, Ethernet, and FCoE interfaces. The first 16 interfaces of the Nexus 5010 support 1 GbE and 10 GbE.
  - **Nexus 5548P:** Thirty-two fixed 1/10 Gbps fixed SFP+ on the base chassis along with one expansion slot. In addition to the fixed interfaces, the Nexus 5548P has one expansion module. The expansion module supports Native Fibre Channel, Ethernet, and FCoE interfaces, for a total of 48 interfaces. In addition to these expansion modules, the 5548P supports a Layer 3 daughtercard that can be ordered with the system or as a spare.

Note The default airflow for the Nexus 5000/5500 platforms is front-to-back, with the back of the chassis being the network port side of the chassis. The Nexus 5548UP and 5596UP support reversed airflow with power supplies and fan trays with "B" SKU /PID. A sample SKU or PID for the Nexus 5548UP reversed airflow for the power supply and fan tray is N55-PAC-750W-B= and N5548P-FAN-B=.

  - **Nexus 5548UP:** Thirty-two fixed Unified ports 1/10 Gbps fixed SFP+ on the base chassis along with one expansion slot. In addition to the fixed interfaces, the Nexus 5548UP has one expansion module. The expansion module

supports native Fibre Channel, Ethernet, and FCoE interface for a total of 48 interfaces. Unified ports on the Nexus 5500 platforms enable an interface to have one of the following characteristics depending on the licensing and pluggable transceiver installed: traditional Ethernet, Fibre Channel (FC), or FCoE. Depending on the configuration, the interface can have the following physical characteristics: 1-Gigabit Ethernet, 10-Gigabit Ethernet, 10-Gigabit Ethernet with FCoE, and 1/2/4/8-G native Fibre Channel. In addition to these expansion modules, the 5548UP supports a Layer 3 daughtercard that can be ordered with the system or as a spare.

- **Nexus 5596UP:** Forty-eight fixed Unified ports 1/10-Gbps fixed SFP+ on the base chassis along with three expansion slots. The expansion module supports native Fibre Channel, Ethernet, and FCoE interfaces, for a total of 96 interfaces. Another expansion module option is the Layer 3 modules for the 5596UP. Unified ports on the Nexus 5500 platforms enable an interface to have one of the following characteristics depending on the licensing and pluggable transceiver installed: traditional Ethernet or FCoE. Depending on the configuration, the interface can have the following physical characteristics: 1-Gigabit Ethernet, 10-Gigabit Ethernet, 10-Gigabit Ethernet with FCoE, and 1/2/4/8 G native Fibre Channel.

Note The Nexus 5010 and Nexus 5020 do not support the following features:

1. Layer 3 module
2. Reversible airflow
3. FabricPath/TRILL
4. Adapter-FEX
5. VM-FEX

- **Nexus 3000:** Delivers high-performance and high-density switching at ultra-low latencies. The Cisco Nexus 3000 Series switches are positioned for use in environments with ultra-low latency requirements such as financial High-Frequency Trading (HFT), chemical genomics, and automotive crash-test simulation applications. These applications require support for advanced unicast and multicast routing protocol features and ultra-low latency; low latency is measured <1 µsecs (microseconds) where below 1 µsecs is measuring ns (nanoseconds):
  - **Nexus 3064:** Forty-eight fixed 1/10-Gigabit Ethernet and four fixed Quad SFP+ (QSFP+) ports. (Each QSFP+ port is 4 × 10 GbE-capable.)
  - **Nexus 3048:** Forty-eight 1GE and 4 10GE.
  - **Nexus 3016:** Sixteen QSFP+ (40GE) ports.

- **Nexus 3548:** Forty-eight fixed Enhanced Small Form-Factor Pluggable (SFP+) ports (1 Gbps or 10 Gbps).

- **Nexus 2000 Fabric Extenders:** A building block in the architecture for the virtualized data center access layer. The FEX architecture provides flexible data center deployment models to meet growing server demands. FEX can be deployed with end-of-row, middle-of-row, top-of-rack, or in any combination leveraging the Fabric interfaces between the FEX and the parent switch. The parent switch can be a Nexus 5000, Nexus 5500, or Nexus 7000. When a FEX is deployed, all the configuration and management is performed on the parent switch. Think of FEX as a remote line card and module to create a virtual chassis; there is not any spanning tree or FabricPath control plane passed between the FEX fabric interface and the parent switches. FEX has several different hardware SKUs based on server requirements:

  - **Nexus 2148:** FEX, 1000BaseT Host Interfaces (server interfaces), and four 10-Gigabit Ethernet fabric uplinks.

  - **Nexus 2248:** FEX, 48 100/1000BaseT Host Interfaces (server interfaces), and four 10-Gigabit Ethernet fabric uplinks.

  - **Nexus 2248TP-E:** Forty-eight ports 100M/1000BaseT Enhanced Host interfaces (server interfaces) and four 10-Gigabit Ethernet Fabric uplinks. The enhanced FEX is buffer optimized for specific data center workloads such as big data, Hadoop, video applications, and distributed storage.

  - **Nexus 2224:** FEX, 24 100/1000BaseT Host interfaces (server interfaces) and two 10 Gigabit Ethernet fabric uplinks.

  - **Nexus 2232TP:** FEX, 32 1/10 Gigabit Ethernet SFP+ server interfaces and eight 10-Gigabit Ethernet SFP+ Fabric uplinks. The 2232TP supports IEEE DCB to transport FCoE.

  - **Nexus 2232TM:** FEX, 32 1000BaseT/10000BaseT Gigabit Ethernet server interfaces, and eight SFP+ 10-Gigabit Ethernet Fabric uplinks. Today, the industry does not support FCoE over 10GBaseT cabling due to the bit-error rate requirements.

  - **Nexus 2232TM-E:** Thirty-two 1/10GBASE-T host interfaces and uplink module (eight 10-Gigabit Ethernet fabric interfaces [SFP+]). The 2232TM-E enables 10GBASE-T PHY, enabling lower power and improved bit error rate (BER).

- **Nexus 1000v:** An NX-OS Software switch that integrated into VMware hypervisor virtualized platforms. The Nexus 1000v architecture has two components: the Virtual Ethernet Module (VEM) and the Virtual Supervisor Module (VSM). The VEM is a software switch embedded in the hypervisor; the VSM is the control plane and management plane to create policies and quality of service (QoS) for virtual machines for each VEM across multiple physical hypervisor hosts.

- **Cisco MDS 9000:** Multilayer SAN switches running Cisco NX-OS. The MDS 9000 offers director-class platforms and Fabric switches. The MDS 9000 offers native fibre channel, storage services, and FCoE.
- **Cisco Unified Computing System (UCS):** Offers a unified, model-based management, end-to-end provisioning, and migration support that come together in this next-generation data center platform to accelerate and simplify application deployment with greater reliability and security. Cisco UCS integrates the server, network, and I/O resources into a single converged architectural system.
- **Nexus 4000:** A purpose-built blade switch for IBM's BladeCenter H and HT chassis. Nexus 4000 is a line-rate, low-latency, nonblocking 10 Gigabit Ethernet and DCB switch module. The Nexus 4000 has 14 fixed 10-Gigabit Ethernet server-facing downlinks (1/10G) and 6 fixed 10-Gigabit Ethernet SFP+ uplinks (1/10G). The Nexus 4000 is a FIP-Snooping Bridge, meaning that it cannot provide Fibre Channel Forwarder (FCF) functionality. The Nexus 4000 cannot participate in FCoE without a Nexus 5000/5500/7000 FCF.

- **B22HP HP-FEX:** Sixteen × 10GBASE-KR internal host interfaces and 8 × 10-Gigabit Ethernet fabric interfaces (Enhanced Small Form-Factor Pluggable).

NX-OS Licensing

Licensing enables specified features to be on a device after you install the appropriate license.

Nexus 7000

The following list outlines the Nexus 7000 licensing options: feature-based licenses, which make features available to the entire physical device, and module-based licenses, which make features available to one module on the physical device.

- **Base services:** The default license that ships with NX-OS covers Layer 2 protocols including such features such as Spanning Tree, virtual LANs (VLANs), Private VLANS, Unidirectional Link Detection (UDLD), and Cisco Trustsec (CTS).

Note CTS has been moved to base services as of NX-OS 6.1.1.

- **Enterprise Services Package:** Provides Layer 3 protocols such as OSPF, Border Gateway Protocol (BGP), Intermediate System-to-Intermediate System (ISIS), Enhanced Interior Gateway Routing Protocol (EIGRP), Policy-Based Routing (PBR), Protocol Independent Multicast (PIM), and Generic Routing Encapsulation (GRE).

- **Advanced Services Package:** Provides VDC
- **Virtual Device Context:** Provides licensing for four VDCs for Supervisor 2 and Supervisor 2e
- **Transport Services License:** Enables OTV support and in NX-OS 5.2(1) enables LISP
- **DCNM for LAN Enterprise License for one Nexus 7000 Chassis:** Enables data center network manager (DCNM) management support on a per chassis basis.
- **Nexus 7010 Scalable Feature License:** Enables XL capabilities and is enabled on a per-chassis basis
- **Enhanced Layer 2 License:** Enables FabricPath
- **Nexus 7000 MPLS License:** Enables MPLS features, including MPLS forwarding, QoS, L3VPN, 6PE/VPE, and OAM
- **FCoE License:** Enables FCoE features on a per F-Series module basis
- **Storage License:** Enables Inter-VSAN Routing (IVR), fabric binding, and access control for FCoE environments.
- **DCNM for SAN Advanced Edition for Nexus 7000:** Enables DCNM SAN management support on a per-chassis basis

Nexus 5500

The Nexus 5500 offers the following licensing options:

- **Nexus 5500 Storage License, 8 Ports:** The Storage Protocol Services License is required to enable an FC or FCoE operation.
- **Nexus 5000 DCNM SAN:** Fabric Manager is licensed per switch and enforced by Fabric Manager.
- **Layer 3 License for Nexus 5500 Platform:** The Nexus 5500 Layer 3 module has two license types: basic license and enterprise license. The base license includes the following Layer 3 feature support: Connected, Static, RIPv2, OSPF (256 Dynamically Learned Routes), EIGRP-Stub, HSRP, VRRP, IGMPv2/3, PIMv2, RACLs, and uRPF. The Nexus 5500 Layer 3 enterprise license includes the following support: EIGRP, OSPF routes (Unrestricted), BGP, and VRF-Lite.

Nexus 3000

The Nexus 3000 offers the following licensing options:

- **Base Layer-3 Services:** Includes Static Routes, Ripv2, EIGRP Stub, OSPF with limited routes, and PIM
- **Enterprise Layer-3 Services:** Includes OSPF unlimited routes, BGP, VRF-lite, and requires the base license

Nexus 2000

For the Nexus 2000, all the licensing is on the parent switch, Nexus 5010, 5020, 5548, 5596, or Nexus 7000.

Nexus 1000v

The Nexus 1000v offers the following licensing options:

- One 1000V license is needed for each installed server CPU/Socket on every VEM in the distributed architecture. There is no limit to the number of cores per CPU or socket.
- Cisco Virtual Security Gateway (VSG) requires one license for each installed server CPU or socket on every VEM.
- Release of Nexus 1000V v2.1 enables a free Essential licensing mode.
  - **Essential Edition:** Available at no cost, the Nexus 1000V Essential Edition provides all the rich Layer 2 networking features to connect virtual applications to the network and integrate into VMware environments, including VXLAN capability, Cisco vPath service insertion, integration with vCloud Director, and a plug-in for management and monitoring in VMware's vCenter Server.
  - **Advanced Edition:** Priced per CPU, the same price as the current Nexus 1000v 1.5 release, the Advanced Edition includes
    - The Cisco VSG for Nexus 1000V, a virtual firewall with visibility to virtual machine attributes for building sophisticated compliance policies, and logical trust zones between applications
    - Support for advanced security capabilities, such as DHCP snooping, IP Source Guard, Dynamic ARP inspection, and Cisco TrustSec Security Group Access (SGA)

Installing the NX-OS License File

Example 1-1 shows the simplicity of installing the NX-OS license file.

Example 1-1 *Displaying and Installing the NX-OS License File*

```
! Once a license file is obtained from Cisco.com and copied to flash, it can be in-
stalled for the chassis.
! Displaying the host-id for License File Creation on Cisco.com:
congo# show license host-id
License hostid: VDH=TBM14404807
```

```
! Installing a License File:
congo# install license bootflash:license_file.lic
Installing license ..done
congo#
```

Note NX-OS offers feature testing for a 120-day grace period. Here is how to enable a 120-day grace period:

```
congo(config)# license grace-period
```

The feature is disabled after the 120-day grace period begins. The license grace period is enabled only for the default admin VDC, VDC1.

Using the grace period enables customers to test, configure, and fully operate a feature without the need for a license to be purchased. This is particularly helpful for testing a feature prior to purchasing a license.

Cisco NX-OS and Cisco IOS Comparison

If you are familiar with the traditional Cisco IOS command-line interface (CLI), the CLI for NX-OS is similar to Cisco IOS. There are key differences that should be understood prior to working with NX-OS, however:

- When you first log in to NX-OS, you go directly into EXEC mode. Because this is different from IOS, you have the option to configure an operational model similar to 6500 IOS for privilege mode (15).
- NX-OS has a setup utility that enables a user to specify the system defaults, perform basic configuration, and apply a predefined Control Plane Policing (CoPP) security policy.
- NX-OS uses a feature-based license model. This enables flexibility in licensing for uses in different areas of the network in which not all features are required.
- NX-OS has the capability to enable and disable features such as OSPF, BGP, and so on via the **feature** configuration command. Configuration and verification commands are not available until you enable the specific feature.
- Interfaces are labeled in the configuration as Ethernet. There aren't any speed designations in the interface name. Interface speed is dynamically learned and reflected in the appropriate **show** commands and interface metrics.
- NX-OS supports VDCs, which enable a physical device to be partitioned into logical devices. When you log in for the first time, you are in the default VDC.

- By default, Cisco NX-OS has two preconfigured instances of Virtual Routing Forwarding (VRF): management and default-default. All Layer 3 interfaces and routing protocols exist in the default VRF. The mgmt0 interface exists in the management VRF and cannot be moved to another VRF. On the Nexus 7000, mgmt0 is accessible from any VDC. If VDCs are configured, each VDC has a unique IP address for the mgmt0 interface.
- Secure Shell version 2 (SSHv2) is enabled by default. (Telnet is disabled by default.)
- The default login administrator user is predefined as admin; a password must be specified when the system is first powered up. With NX-OS, you must enter a username and password; you cannot disable the username and password login. In contrast, in IOS you can simply type a password; you can optionally set the login to require the use of a username.
- NX-OS uses a kickstart image and a system image. Both images are identified in the configuration file as the kickstart and system boot variables. The first image that boots is the kickstart image, which provides the Linux kernel, basic drivers, and initial file system. The NX-OS system image boots after the kickstart image; the system image provides L2, L3, infrastructure and feature support such as OTV, multicast, FEX, and so on.
- NX-OS removed the **write memory** command; use the **copy running-config startup-config**. The alias command syntax can be used to create an alias for a shortcut.
- The default Spanning Tree mode in NX-OS is Rapid-PVST+.

Caution In NX-OS, you must enable features such as OSPF, BGP, and CTS. If you remove a feature via the **no** feature command, all relevant commands related to that feature are removed from the running configuration.

For example, when configuring vty timeouts and session limits, consider Example 1-2, which illustrates the difference between IOS and NX-OS syntax.

Example 1-2 *vty Configurations and Session Limits, Comparing the Differences Between Traditional IOS and NX-OS*

```
! IOS:
congo#
congo(config)# line vty 0 9
congo(config)# exec-timeout 15 0
congo(config)# login
congo# copy running-config startup-config
-----------------------------------------------------------------
! NX-OS:
congo(config)# line vty
```

```
congo(config)# session-limit 10
congo(config)# exec-timeout 15

congo# copy running-config startup-config
```

NX-OS User Modes

Cisco NX-OS CLI is divided into command modes, which define the actions available to the user. Command modes are "nested" and must be accessed in sequence. As you navigate from one command mode to another, an increasingly more specific set of commands becomes available. All commands in a higher command mode are accessible from lower command modes. For example, the **show** commands are available from any configuration command mode. Figure 1-1 shows how command access builds from the EXEC mode to the global configuration mode.

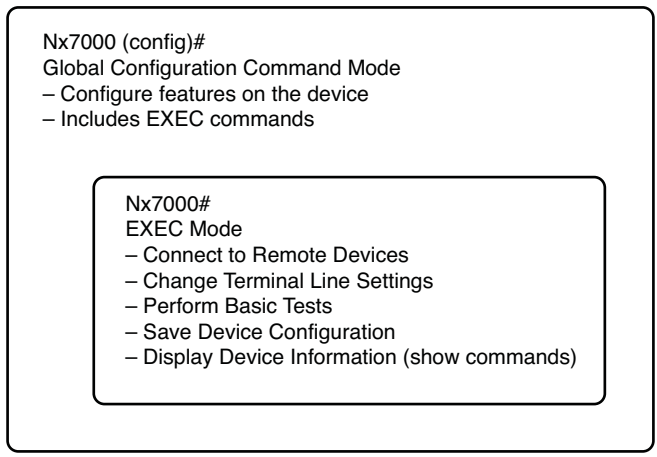

Figure 1-1 *NX-OS Command Access from EXEC Mode to Global Configuration Mode*

EXEC Command Mode

When you first log in, Cisco NX-OS Software places you in EXEC mode. As demonstrated in Example 1-3, the commands available in EXEC mode include the **show** commands that display device status and configuration information, the **clear** commands, and other commands that perform actions that you do not save in the device configuration.

Example 1-3 *Cisco NX-OS EXEC Mode*

```
Congo# show interface ethernet 1/15
Ethernet1/15 is down (SFP not inserted)
  Hardware: 10000 Ethernet, address: 001b.54c2.bbc1 (bia 001b.54c1.e4da)
```

```
  MTU 1500 bytes, BW 10000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA
  auto-duplex, auto-speed
  Beacon is turned off
  Auto-Negotiation is turned off
  Input flow-control is off, output flow-control is off
  Switchport monitor is off
  Last link flapped never
  Last clearing of "show interface" counters never
  30 seconds input rate 0 bits/sec, 0 packets/sec
  30 seconds output rate 0 bits/sec, 0 packets/sec
  Load-Interval #2: 5 minute (300 seconds)
    input rate 0 bps, 0 pps; output rate 0 bps, 0 pps
  L3 in Switched:
    ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
  L3 out Switched:
    ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
! Output omitted for brevity

Congo#
```

Global Configuration Command Mode

Global configuration mode provides access to the broadest range of commands. The term *global* indicates characteristics or features that affect the device as a whole. You can enter commands in global configuration mode to configure your device globally or enter more specific configuration modes to configure specific elements such as interfaces or protocols as demonstrated here:

```
Nx7000# conf t
Nx7000(config)# interface ethernet 1/15
```

Interface Configuration Command Mode

One example of a specific configuration mode that you enter from the global configuration mode is the interface configuration mode. To configure interfaces on your device, you must specify the interface and enter interface configuration mode.

You must enable many features on a per-interface basis. Interface configuration commands modify the operation of the interfaces on the device, such as Ethernet interfaces or management interfaces (mgmt 0).

NX-OS supports different Ethernet interface types such as Gigabit Ethernet and 10-Gigabit Ethernet interfaces. All interfaces are referred to the Ethernet; NX-OS does not designate Gigabit or 10-Gigabit Ethernet interfaces. In Example 1-4, interface 1/15 is a 10-Gigabit Ethernet interface. With NX-OS 5.1(1) and 5.2(1) and later, the default interface is added to NX-OS; this enables the administrator to quickly reset the interface configuration back to the default settings.

Example 1-4 demonstrates moving between the different command modes in NX-OS.

Example 1-4 *Interface Ethernet1/5 Is a 10-Gigabit Ethernet Interface—Show How the Interface Is Designated at the Ethernet and Not the Interface Ten1/15*

```
congo# conf t
congo(config)# interface ethernet 1/15
congo(config-if)# exit
Congo# show interface ethernet 1/15
Ethernet1/15 is down (SFP not inserted)
  Hardware: 10000 Ethernet, address: 001b.54c2.bbc1 (bia 001b.54c1.e4da)
  MTU 1500 bytes, BW 10000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA
  auto-duplex, auto-speed
  Beacon is turned off
  Auto-Negotiation is turned off
  Input flow-control is off, output flow-control is off
  Switchport monitor is off
  Last link flapped never
  Last clearing of "show interface" counters never
  30 seconds input rate 0 bits/sec, 0 packets/sec
  30 seconds output rate 0 bits/sec, 0 packets/sec
  Load-Interval #2: 5 minute (300 seconds)
    input rate 0 bps, 0 pps; output rate 0 bps, 0 pps
  L3 in Switched:
    ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
  L3 out Switched:
    ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes

Congo#
```

Management Interfaces

NX-OS has many different types of management interfaces, all of which the following section covers:

- **Controller Processor (CP)/Supervisor:** Has both the management plane and control plane and is critical to the operation of the network.
- **Connectivity Management Processor (CMP):** Provides a second network interface to the device for use even when the CP is not reachable. The CMP interface is used for out-of-band management and monitoring; the CMP interface is independent from the primary operating system. The CMP interface is only available on the Nexus 7000 Supervisor Module 1.
- **MGMT0:** Provides true out-of-band management through a dedicated interface and VRF to ensure 100 percent isolation from either control plane or data plane. MGMT0 enables you to manage the devices by the IPv4 or IPv6 address on the MGMT0 interface; the mgmt0 interface is a 10/100/1000 Ethernet interface. When implementing vPC, a best practice is to use the MGMT0 interface for the vPC keepalive link.
- **Telnet:** Provides an unsecure management connection to the NX-OS device.
- **SSH:** Provides a secure management connection to the NX-OS device.
- **Extended Markup Language (XML) management interfaces:** Use the XML-based Network Configuration Protocol (NETCONF) that enables management, monitoring, and communication over the interface with an XML management tool or program.
- **Simple Network Management Protocol (SNMP):** Used by management systems to monitor and configure devices via a set of standards for communication over the TCP/IP protocol.

Controller Processor (Supervisor Module)

The Cisco Nexus 7000 series supervisor module is designed to deliver scalable control plane and management functions for the Cisco Nexus 7000 Series chassis. The Nexus 7000 supervisor module is based on an Intel dual-core processor that enables a scalable control plane. The supervisor modules controls the Layer 2 and Layer 3 services, redundancy capabilities, configuration management, status monitoring, power, and environmental management. The supervisor module also provides centralized arbitration to the system Fabric for all line cards. The fully distributed forwarding architecture enables the supervisor to support transparent upgrades to higher forwarding capacity-capable I/O and Fabric modules. Two supervisors are required for a fully redundant system, with one supervisor module running as the active device and the other in hot standby mode, providing exceptional high-availability features in data center class products. Additional features and benefits of the Nexus 7000 supervisor modules to meet demanding data center requirements follow:

- Active and standby supervisor.
- ISSU with dual supervisor modules installed in the Nexus 7000. As of NX-OS 4.2(1)N1 for the Nexus 5000/5500 supports ISSU, the following caveats need to be adhered to for ISSU support:

  - ISSU on the Nexus 5500 is not support with a Layer 3 module installed.
  - ISSU on the Nexus 5000/5500 is not supported if the Nexus 5000/5500 is the STP Root Bridge. The reason is that the root bridge needs to generate BPDUs every hello interval (2 sec). During the ISSU, the process restart time is too long to ensure that the process comes back prior to three times the hello dead interval. Because the Nexus 5000/5500 is positioned for the data center access layer, this typically is not an issue because the STP root would sit at the L2/L3 boundary of the data center aggregation layer.
- Virtual output queuing (VoQ), which is a QoS-aware lossless Fabric, avoids the problems associated with head-of-line blocking.
- USB interfaces that enable access to USB flash memory devices for software image loading and recovery.
- Central arbitration that provides symmetrical control of the flow of traffic through the switch Fabric helps ensure transparent switchover with no losses.
- Segmented and redundant out-of-band provisioning and management paths.
- Virtualization of the management plane via VDC is available on the Nexus 7000.
- Integrated diagnostics and protocol decoding with an embedded control plane packet analyzer; this is based on the Wireshark open source. (No additional licenses are required.) The *EthAnalyzer* provides a real-time, on-the-device protocol analyzer to monitor traffic from inband and mgmt0 interfaces to the Control Processor (supervisor module). The *EthAnalyzer* also provides extensive capture and display options and capturing to a standard .pcap file.
- Fully decoupled control plane and data plane with no hardware forwarding on the module.
- Distributed forwarding architecture, enabling independent upgrades of the supervisor and Fabric.
- With Central arbitration and VoQ, enabling Unified Fabric.
- Transparent upgrade capacity and capability; designed to support 40-Gigabit and 100-Gigabit Ethernet.
- System locator and beacon light-emitting diodes (LEDs) for simplified operations.
- Dedicated out-of-band management processor for lights-out management, the CMP.

Connectivity Management Processor (CMP)

The supervisor incorporates an innovative dedicated CMP to support remote management and troubleshooting of the complete system. The CMP provides a complete out-of-band management and monitoring capability independent from the primary operating

system. The CMP enables lights-out management of the supervisor module, all modules, and the Cisco Nexus 7000 Series system without the need for separate terminal servers with the associated additional complexity and cost. The CMP delivers the remote control through its own dedicated processor, memory, and boot flash memory and a separate Ethernet management port. The CMP can reset all system components and can also reset the host supervisor module to which it is attached, enabling a complete system restart.

Note The CMP interface is only on the Supervisor-1; CMP is not an option on the Supervisor-2 and the Supervisor-2e.

The CMP offer many benefits, including the following:

- Dedicated processor and memory, and boot flash.
- The CMP interface can reset all the system components, which include the supervisor module, and system restart.
- An independent remote system management and monitoring capability enables lights-out management of the system.
- Remote monitoring of supervisor status and initiation of resets that removes the need for separate terminal server devices for out-of-band management.
- System reset while retaining out-of-band Ethernet connectivity, which reduces the need for onsite support during system maintenance.
- Capability to remotely view boot-time messages during the entire boot process.
- Capability to initiate a complete system power shutdown and restart, which eliminates the need for local operator intervention to reset power for devices.
- Login authentication, which provides secure access to the out-of-band management environment.
- Access to supervisor logs that enables rapid detection and prevention of potential system problems.
- Capability to take full console control of the supervisor.

Example 1-5 shows how to connect to the CMP interface and the available **show** commands available from the CMP interface. Also, note the escape sequence of "~," to get back to the main NX-OS interface. You can also connect from the CMP back to the CP module.

Example 1-5 *Connecting to the CMP Interface, Displaying Available show Commands*

```
N7010-1# attach cmp
Connected
Escape character is '~,' [tilde comma]

N7010-1-cmp5 login: admin
Password:
Last login: Tue Aug 11 23:58:12 2009 on ttyS1

N7010-1-cmp5# attach cp
This command will disconnect the front-panel console on this supervisor, and will
clear all console attach sessions on the CP - proceed(y/n)? y
N7010-1#

N7010-1# attach cmp
Connected
Escape character is '~,' [tilda comma]

N7010-1-cmp5 login: admin
Password:
Last login: Wed Aug 12 00:06:12 2009 on ttyS1
N7010-1-cmp5# show ?
  attach          Serial attach/monitor processes
  clock           Display current date
  cores           Show all core dumps for CMP
  cp              Show CP status information
  hardware        Show cmp hardware information
  interface       Display interface information
  line            Show cmp line information
  logging         Show logging configuration and contents of logfile
  logs            Show all log files for CMP
  processes       Show cmp processes information
  running-config  Current operating configuration
  sprom           Show SPROM contents
  ssh             SSH information
  system          Show system information
  users           Show the current users logged in the system
  version         Show cmp boot information
```

Telnet

NX-OS provides support for the Telnet server and client. The Telnet protocol enables TCP/IP terminal connections to a host. Telnet enables a user at one site to establish a

TCP connection to a login server at another site and then passes the keystrokes from one device to the other. Telnet can accept either an IP address or a domain name as the remote device address. Telnet sessions are not encrypted, and SSH is recommended instead.

Note Remember that the Telnet server is disabled by default in NX-OS.

Example 1-6 demonstrates how to enable a Telnet server in NX-OS.

Example 1-6 *Enabling a Telnet Server in NX-OS*

```
N7010-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
N7010-1(config)# feature telnet
N7010-1(config)# show telnet server
telnet service enabled
N7010-1(config)# copy running-config startup-config
[########################################] 100%
```

SSH

NX-OS supports an SSH server and SSH client. Use an SSH server to enable an SSH client to make a secure, encrypted connection to a Cisco NX-OS device; SSH uses strong encryption for authentication. The SSH server in Cisco NX-OS Software can interoperate with publicly and commercially available SSH clients. The user authentication mechanisms supported for SSH are Remote Authentication Dial-In User Service (RADIUS), Terminal Access Controller Access Control System Plus (TACACS+), and the use of locally stored usernames and passwords.

The SSH client application enables the SSH protocol to provide device authentication and encryption. The SSH client enables a Cisco NX-OS device to make a secure, encrypted connection to another Cisco NX-OS device or to any other device that runs the SSH server.

SSH requires server keys for secure communications to the Cisco NX-OS device. You can use SSH server keys for the following SSH options:

- SSH version 2 using Rivest, Shamir, and Adelman (RSA) public-key cryptography
- SSH version 2 using the Digital System Algorithm (DSA)

Be sure to have an SSH server key-pair with the appropriate version before allowing the SSH service. You can generate the SSH server key-pair according to the SSH client version used. The SSH service accepts two types of key-pairs for use by SSH version 2:

- The *dsa* option generates the DSA key-pair for the SSH version 2 protocol.
- The *rsa* option generates the RSA key-pair for the SSH version 2 protocol.

By default, Cisco NX-OS Software generates an RSA key using 1024 bits.

SSH supports the following public key formats:

- OpenSSH
- IETF Secure Shell (SECSH)

Example 1-7 demonstrates how to enable an SSH server and configure the SSH server keys.

Example 1-7 *Enabling an SSH Server and Configuring SSH Server Keys*

```
N7010-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
N7010-1(config)# no feature ssh
XML interface to system may become unavailable since ssh is disabled
N7010-1(config)# ssh key rsa 2048
generating rsa key(2048 bits).....
..
generated rsa key
N7010-1(config)# feature ssh
N7010-1(config)# exit
N7010-1# show ssh key
*************************************
rsa Keys generated:Thu Aug 13 23:33:41 2009
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEA6+TdX+ABH/mq1gQbfhhsjBmm65ksgfQb3Mb3qbwUbNlc
Aa6fjJCGdHuf3kJox/hjgPDChJOdkUXHjESlV59OhZP/NHlBrBq0TGRr+hfdAssD3wG5oPkywgM4+bR/
ssCzoj6jVG41tGmfPip4pr3dqsMzR21DXSKK/tdj7bipWKy1wSkYQzZwatIVPIXRqTJY7L9a+JqVIJEA
0QlJM1l0wZ5YbxccB2GKNKCM2x2BZl4okVgl80CCJg7vmn+8RqIOQ5jNAPNeb9kFw9nsPj/r5xFC1RcS
KeQbdYAjItU6cX1TslRnKjlWewCgIa26dEaGdawMVuftgu0uM97VCOxZPQ==

bitcount:2048
fingerprint:
1f:b7:a3:3b:f5:ca:a6:36:19:93:98:c7:37:ba:27:db
*************************************
could not retrieve dsa key information
*************************************
N7010-1# show ssh server
ssh version 2 is enabled
N7010-1(config)# username nxos-admin password C1sc0123!

N7010-1(config)# username nxos-admin sshkey ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAQEA6+TdX+ABH/mq1gQbfhhsjBmm65ksgfQb3Mb3qbwUbNlcAa6fjJCG-
```

```
dHuf3kJox/hjgP
DChJOd-
kUXHjESlV59OhZP/NHlBrBq0TGRr+hfdAssD3wG5oPkywgM4+bR/ssCzoj6jVG41tGmfPip4pr3dqsMz-
R21DXSKK/tdj7b
ip-
WKy1wSkYQzZwatIVPIXRqTJY7L9a+JqVIJEA0QlJM1l0wZ5YbxccB2GKNKCM2x2BZl4okVgl80CCJg7vmn+8
RqIOQ5jNAP
Neb9kFw9nsPj/r5xFC1RcSKeQbdYAjItU6cX1TslRnKjlWewCgIa26dEaGdawMVuftgu0uM97VCOxZPQ==
N7010-1(config)# show user-account
user:admin
        this user account has no expiry date
        roles:network-admin
user:nxos-admin
        this user account has no expiry date
        roles:network-operator
        ssh public key: ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEA6+TdX+ABH/mq1gQbfhh-
sjBmm65ksgfQb3Mb3qbwUbNlcAa6fjJCGdHuf3kJox/hjgP
DChJOd-
kUXHjESlV59OhZP/NHlBrBq0TGRr+hfdAssD3wG5oPkywgM4+bR/ssCzoj6jVG41tGmfPip4pr3dqsMz-
R21DXSKK/tdj7b
ip-
WKy1wSkYQzZwatIVPIXRqTJY7L9a+JqVIJEA0QlJM1l0wZ5YbxccB2GKNKCM2x2BZl4okVgl80CCJg7vmn+8
RqIOQ5jNAP
Neb9kFw9nsPj/r5xFC1RcSKeQbdYAjItU6cX1TslRnKjlWewCgIa26dEaGdawMVuftgu0uM97VCOxZPQ==
N7010-1(config)#
N7010-1# copy running-config startup-config
[########################################] 100%
N7010-1#
```

NX-OS has a robust XML management interface, which can be used to configure the entire switch. The interface uses the XML-based NETCONF that enables you to manage devices and communicate over the interface with an XML management tool or a program. NETCONF is based on RFC 4741, and the NX-OS implementation requires you to use an SSH session for communication with the device.

NETCONF is implemented with an XML Schema (XSD) that enables you to enclose device configuration elements within a remote procedure call (RPC) message. From within an RPC message, you select one of the NETCONF operations that matches the type of command that you want the device to execute. You can configure the entire set of CLI commands on the device with NETCONF.

The XML management interface does not require any additional licensing. XML management is included with no additional charge.

XML/NETCONF can be enabled via a web2.0/ajax browser application that uses XML/NETCONF to pull all statistics off all interfaces on the Nexus 7000 running NX-OS in a dynamically updating table.

Figures 1-2, 1-3, and 1-4 demonstrate sample output from the XML/NETCONF interface.

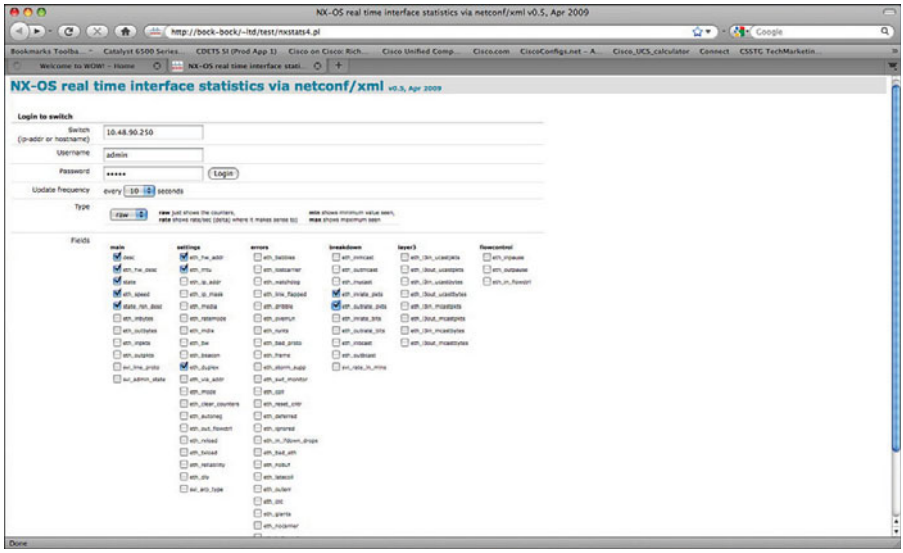

Figure 1-2 *Obtaining NX-OS Real-Time Interface Statistics via NETCONF/XML; the IP Address Entered Is the NX-OS mgmt0 Interface*

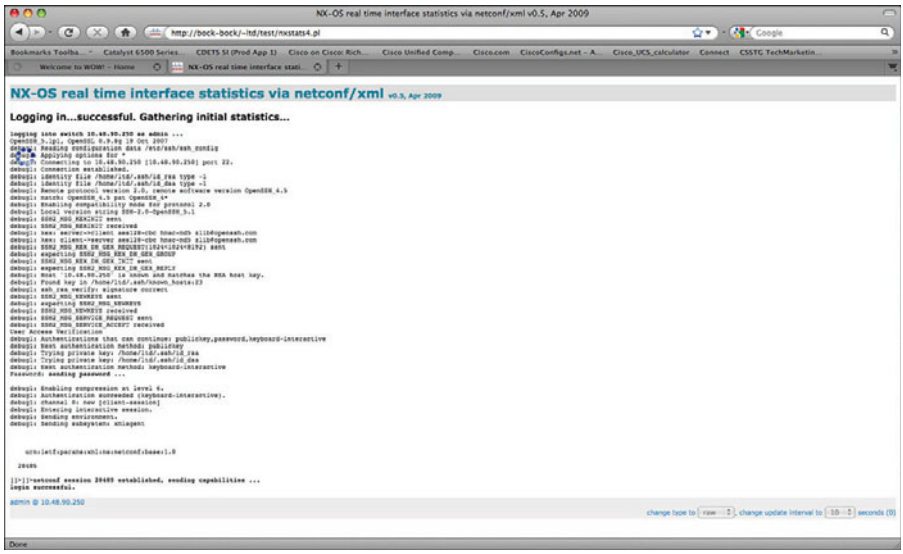

Figure 1-3 *Login Results to the NX-OS Devices via NETCONF/XML*

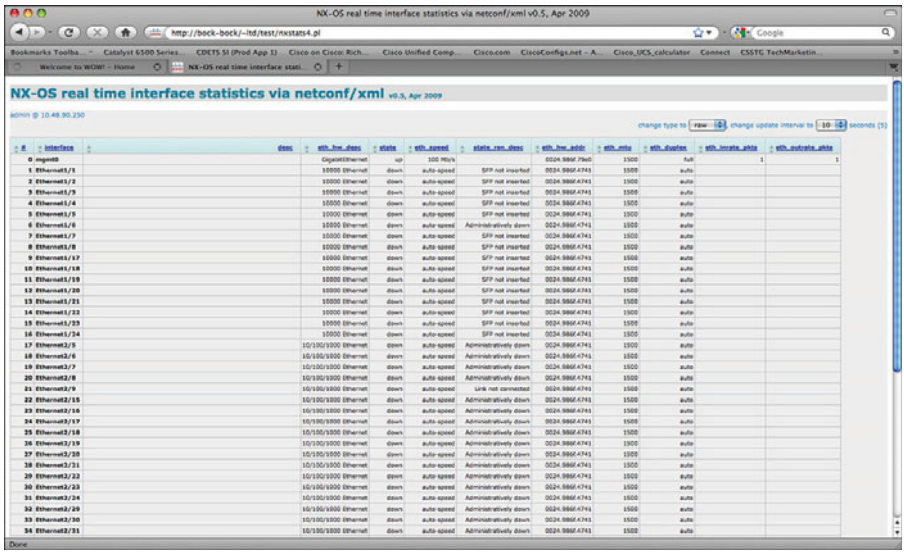

Figure 1-4 *Results of the Selected Attributes, Such as Speed, Duplex, Errors, Counters, MAC Address; the Page Refreshes Every 10 Seconds*

SNMP

The Simple Network Management Protocol is an application layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

SNMP has different versions such as SNMPv1, v2, and v3. Each SNMP version has different security models or levels. Most Enterprise customers want to implement SNMPv3 because it offers encryption to pass management information (or traffic) across the network. The security level determines if an SNMP message needs to be protected and authenticated. Various security levels exist within a security model:

- **noAuthNoPriv:** Security level that does not provide authentication or encryption
- **authNoPriv:** Security level that provides authentication but does not provide encryption
- **authPriv:** Security level that provides both authentication and encryption

Cisco NX-OS supports the following SNMP standards:

- **SNMPv1:** Simple community-string based access.
- **SNMPv2c:** RFC 2575-based group access that can be tied into RBAC model.

- **SNMPv3:** Enables for two independent security mechanisms, authentication (Hashed Message Authentication leveraging either Secure Hash Algorithm [SHA-1] or Message Digest 5 [MD5] algorithms) and encryption (Data Encryption Standard [DES] as the default and AES), to ensure secure communication between NMS station and NX-OS. Both mechanisms are implemented, as shown in Example 1-8.

Because NX-OS is truly modular and highly available, the NX-OS implementation of SNMP supports stateless restarts for SNMP. NX-OS has also implemented virtualization support for SNMP; NX-OS supports one instance of SNMP per VDC. SNMP is also VRF-aware, which enables you to configure SNMP to use a particular VRF to reach the network management host.

Example 1-8 demonstrates how to enable SNMPv3 on NX-OS.

Example 1-8 *Enabling SNMPv3 on NX-OS*

```
N7010-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
N7010-1(config)# snmp-server user NMS auth sha Cisc0123! priv Cisc0123! engineID
00:00:00:63:00:01:00:10:20:15:10:03
N7010-1(config)# snmp-server host 10.100.22.254 informs version 3 auth NMS
N7010-1(config)# snmp-server community public ro
N7010-1(config)# snmp-server community nxos rw
N7010-1(config)# show snmp
sys contact:
sys location:
0 SNMP packets input
        0 Bad SNMP versions
        0 Unknown community name
        0 Illegal operation for community name supplied
        0 Encoding errors
        0 Number of requested variables
        0 Number of altered variables
        0 Get-request PDUs
        0 Get-next PDUs
        0 Set-request PDUs
        0 No such name PDU
        0 Bad value PDU
        0 Read Only PDU
        0 General errors
        0 Get Responses
45 SNMP packets output
        45 Trap PDU
        0 Too big errors
        0 No such name errors
```

```
        0 Bad values errors
        0 General errors
        0 Get Requests
        0 Get Next Requests
        0 Set Requests
        0 Get Responses
        0 Silent drops
Community            Group / Access      context    acl_filter
---------            --------------      -------    ----------
nxos                  network-admin
public                network-operator

_____________________________________________________________
                  SNMP USERS
_____________________________________________________________
User                          Auth  Priv(enforce) Groups
____                          ____  _____________ ______
admin                         md5   des(no)       network-admin

nxos-admin                    sha   des(no)       network-operator
_____________________________________________________________
 NOTIFICATION TARGET USERS (configured  for sending V3 Inform)
_____________________________________________________________
User                          Auth  Priv
____                          ____  ____
NMS                           sha   des
(EngineID 0:0:0:63:0:1:0:10:20:15:10:3)
SNMP Tcp Authentication Flag : Enabled.
-------------------------------------------------------------------------------
Port Monitor : enabled
-------------------------------------------------------------------------------
Policy Name  : default
Admin status : Not Active
Oper status  : Not Active
Port type    : All Ports
-------------------------------------------------------------------------------
Counter          Threshold  Interval Rising Threshold event Falling Threshold  event
In Use
-------          ---------  -------- ---------------- ----- ------------------ --
Link Loss        Delta      60       5                4     1         4     Yes
Sync Loss        Delta      60       5                4     1         4     Yes
Protocol Error   Delta      60       1                4     0         4     Yes
Signal Loss      Delta      60       5                4     1         4     Yes
Invalid Words    Delta      60       1                4     0         4     Yes
Invalid CRC's    Delta      60       5                4     1         4     Yes
RX Performance   Delta      60       2147483648       4     524288000  4     Yes
```

```
TX Performance   Delta      60       2147483648       4     524288000  4     Yes
-------------------------------------------------------------------------------------
SNMP protocol : Enabled
-----------------------------------------------------------------
Context                          [Protocol instance, VRF, Topology]

N7010-1# show snmp user
_____________________________________________________________
                  SNMP USERS
_____________________________________________________________

User                           Auth  Priv(enforce) Groups
____                           ____  ____________  ______
admin                          md5   des(no)       network-admin

nxos-admin                     sha   des(no)       network-operator


_____________________________________________________________
 NOTIFICATION TARGET USERS (configured  for sending V3 Inform)
_____________________________________________________________

User                           Auth  Priv
____                           ____  ____
NMS                            sha   des
(EngineID 0:0:0:63:0:1:0:10:20:15:10:3)
N7010-1(config)# exit
N7010-1# copy running-config  startup-config
 [########################################] 100%
N7010-1#
```

DCNM

Cisco Data Center Network Manager is a management solution that supports NX-OS devices. DCNM maximizes the overall data center infrastructure uptime and reliability, which improves service levels. Focused on the operational management requirements of the data center, DCNM provides a robust framework and rich feature set that fulfills the switching, application, automation, provisioning, and services needs of today's data centers and tomorrow's data center requirements.

DCNM is a client-server application supporting a Java-based client-server application. The DCNM client communicates with the DCNM server only, never directly with managed Cisco NX-OS devices. The DCNM server uses the XML management interface of Cisco NX-OS devices to manage and monitor them. The XML management interface is a programmatic method based on the NETCONF protocol that complements the CLI functionality.

DCNM has a robust configuration and feature support on the NX-OS platform. The following features can be configured, provisioned, and monitored through DCNM enterprise management:

- Physical ports.
- Port channels and vPCs.
- Loopback and management interfaces.
- VLAN network interfaces (sometimes referred to as switched virtual interfaces [SVI]).
- VLAN and private VLAN (PVLAN).
- Spanning Tree Protocol, including Rapid Spanning Tree (RST) and Multi-Instance Spanning Tree Protocol (MST).
- Virtual Device Contexts.
- Gateway Load Balancing Protocol (GLBP) and object tracking.
- Hot Standby Router Protocol (HSRP).
- Access control lists.
- IEEE 802.1X.
- Authentication, authorization, and accounting (AAA).
- Role-based access control.
- Dynamic Host Configuration Protocol (DHCP) snooping.
- Dynamic Address Resolution Protocol (ARP) inspection.
- IP Source Guard.
- Traffic storm control.
- Port security.
- Hardware resource utilization with Ternary Content Addressable Memory (TCAM) statistics.
- Switched Port Analyzer (SPAN).
- Network Path Analysis (PONG); the PONG feature enables you to trace the path latency between two nodes at a given time interval and to monitor the latency information in the form of statistics, based on the polling frequency.
- MDS 9000 support.
- Nexus 1000v Support.
- Nexus 4000 Support.
- Nexus 3000 Support.

- Catalyst 6500 support.
- Nexus 7000.
- Nexus 5000.
- Nexus 5500.
- UCS Support.
- FCoE Provisioning and Management—the wizard-based provisioning enables for simple configuration of Fibre Channel and FCoE interfaces. The performance monitoring of the FCoE path can show how much Fibre Channel versus Ethernet traffic moves through the path. The topology view can display the FCoE path through SAN or LAN switches out to the VMware virtual infrastructure.
- Port Profile Support enables the administrator to create, delete, and modify the Layer 2 parameters of port profiles on Cisco Nexus 7000 Series devices. You can also view the interface types that inherit a given profile.
- FabricPath monitoring enables the administrator to view the switch ID for devices enabled for FabricPath and view the conflicts between switches.
- FEX Layer 3 Routed Port for HIF Ports enables the administrator to configure and manage Layer 3 interfaces for FEX interfaces.
- Shared interfaces enable the administrator to create a new storage VDC, share ports across the Ethernet and storage VDC, and configure allowed VLANs from the Ethernet VDC to the storage VDC.

DCNM also includes end-end enterprise visibility including topology views, event browsers, configuration change management, device operating system management, hardware asset inventory, logging, and statistical data collection management.

Managing System Files

Directories can be created on bootflash: and external flash memory (slot0:, usb1:, and usb2:); you can also navigate through these directories and use them for files. Files can be created and accessed on bootflash:, volatile:, slot0:, usb1:, and usb2: file systems. Files can be accessed only on the system: file systems. A debug file system can be used for debug log files specified in the **debug** *logfile* command. System image files, from remote servers using FTP, Secure Copy (SCP), Secure Shell FTP (SFTP), and TFTP, can also be downloaded.

File Systems

Table 1-1 outlines the parameters for the syntax for specifying a local file system, which is

filesystem:[//module/]

Example 1-9 demonstrates some file system commands and how to copy a file.

Table 1-1 *Syntax for Specifying a Local File System*

File System Name	Module	Description
Bootflash	sup-active sup-local sup-1 sup-2	Internal CompactFlash memory located on the active supervisor module used for storing image files, configuration files, and other miscellaneous files. The initial default directory is bootflash.
Bootflash	sup-standby sup-remote sup-1 sup-2	Internal CompactFlash memory located on the standby supervisor module used for storing image files, configuration files, and other miscellaneous files.
slot0	sup-standby sup-remote sup-1 sup-2	External CompactFlash memory installed in a supervisor module used for storing system images, configuration files, and other miscellaneous files.
volatile	Not applicable	Volatile random-access memory (VRAM) located on a supervisor module used for temporary or pending changes.
Nvram	Not applicable	Nonvolatile random-access memory (NVRAM) located on a supervisor module used for storing the startup-configuration file.
Log	Not applicable	Memory on the active supervisor that stores logging file statistics.
system	Not applicable	Memory on a supervisor module used for storing the running-configuration file.
debug	Not applicable	Memory on a supervisor module used for debug logs.
usb1	Not applicable	External USB flash memory installed in a supervisor module used for storing image files, configuration files, and other miscellaneous files.

File System Name	Module	Description
usb2	Not applicable	External USB flash memory installed in a supervisor module used for storing image files, configuration files, and other miscellaneous files.

Example 1-9 *File System Commands/Copying a File*

```
N7010-1# dir bootflash:
        43032    Jul 26 17:20:27 2011  .sksd_crypt_service_1
      49445    Jul 26 16:48:31 2011  .sksd_crypt_service_2
      39969    Jul 26 16:48:32 2011  .sksd_crypt_service_3
      30317    Jul 26 16:48:32 2011  .sksd_crypt_service_4
        315    Oct 04 11:19:14 2010  Advanced.lic
       7615    Jul 29 13:10:08 2011  DubPoc-7K1-PreConfig.txt
        308    Oct 04 11:19:30 2010  Enhancedl2.lic
        317    Oct 04 11:19:00 2010  Enterprise.lic
        257    Aug 29 14:43:49 2011  N7K-AIDA-FCoE
        257    Aug 29 14:45:24 2011  N7K-AIDA-FCoE.lic
        257    Aug 29 14:44:01 2011  N7K-AIDA-FCoE2
        257    Aug 29 14:45:34 2011  N7K-AIDA-FCoE2.lic
        300    Aug 23 00:36:46 2011  N7K1-FCOE1.lic
        300    Aug 23 00:37:04 2011  N7K1-FCOE2.lic
        298    Aug 23 00:37:17 2011  N7K1-MPLS.lic
        301    Aug 23 00:37:33 2011  N7K1-SAN.lic
        311    Oct 04 11:17:57 2010  Otv.lic
      33766    Jul 26 16:48:32 2011  dana1.txt
        309    Mar 21 15:43:51 2011  dc1-fp.lic
       4096    Aug 05 17:04:31 2011  lost+found/
  146701191    Jul 28 18:49:31 2011  n7000-s1-dk9.5.1.3.bin
  146247835    Jul 01 15:26:11 2011  n7000-s1-dk9.5.1.4.bin
  161980383    Aug 04 13:48:52 2011  n7000-s1-dk9.5.2.1.bin
   13564350    Oct 26 12:04:14 2010  n7000-s1-epld.5.1.1.img
   13574595    Aug 04 13:56:45 2011  n7000-s1-epld.5.2.1.img
   30674944    Jul 28 18:50:31 2011  n7000-s1-kickstart.5.1.3.bin
   30691328    Jul 26 16:49:37 2011  n7000-s1-kickstart.5.1.4.bin
   29471232    Aug 04 13:52:46 2011  n7000-s1-kickstart.5.2.1.bin
       4096    Jul 29 13:39:58 2011  vdc_2/
       4096    Jul 29 13:40:39 2011  vdc_3/
       4096    Aug 12 13:01:23 2011  vdc_4/

Usage for bootflash://sup-local
  705400832 bytes used
```

```
 1104498688 bytes free
 1809899520 bytes total
CMHLAB-DC1-SW2-OTV1#
Usage for bootflash://sup-local
  982306816 bytes used
  827592704 bytes free
 1809899520 bytes total
N7010-1# dir bootflash://sup-remote
      12349     Dec 05 02:15:33 2008  7k-1-vdc-all.run
       4096     Apr 04 06:45:28 2009  eem/
      18180     Apr 02 23:47:26 2009  eem_script.cfg
   99851395     Aug 03 05:20:20 2009  congo-s1-dk9.4.2.0.601.bin
  100122301     Aug 12 04:46:18 2009  congo-s1-dk9.4.2.1.bin
      19021     Apr 03 21:04:50 2009  eem_script_counters.cfg
      19781     Apr 05 23:30:51 2009  eem_script_iptrack.cfg
      29104     Jun 19 22:44:51 2009  ethpm_act_logs.log
          0     Jun 19 22:44:51 2009  ethpm_syslogs.log
        175     Jun 20 04:14:37 2009  libotm.log
      49152     Jun 19 22:38:45 2009  lost+found/
   87755113     Apr 07 23:54:07 2009  congo-s1-dk9.4.0.4.bin
   92000595     Apr 16 21:55:19 2009  congo-s1-dk9.4.1.4.bin
   92645614     Apr 08 06:08:35 2009  congo-s1-dk9.4.1.5.bin
   92004757     Jun 02 04:29:19 2009  congo-s1-dk9.4.1.5E2.bin
   10993389     Mar 22 04:55:13 2009  congo-s1-epld.4.1.3.33.img
   23785984     Apr 07 23:47:43 2009  congo-s1-kickstart.4.0.4.bin
   24718848     Apr 16 21:52:40 2009  congo-s1-kickstart.4.1.4.bin
   25173504     Apr 08 06:00:57 2009  congo-s1-kickstart.4.1.5.bin
   23936512     Jun 02 04:26:35 2009  congo-s1-kickstart.4.1.5E2.bin
   25333248     Aug 03 05:19:26 2009  congo-s1-kickstart.4.2.0.601.bin
   25234944     Aug 12 04:45:24 2009  congo-s1-kickstart.4.2.1.bin
        310     Sep 19 03:58:55 2008  n7k-rhs-1.lic
      12699     Jan 23 14:02:52 2009  run_vpc_jan22
      11562     Mar 13 07:52:42 2009  startup-robert-cfg
      16008     Mar 12 02:02:40 2009  startup-vss-cfg
      17315     Mar 19 06:24:32 2009  startup-vss-cfg_roberto_mar18
         99     Apr 04 06:51:15 2009  test1
       9991     Jun 19 23:12:48 2009  vdc.cfg
       4096     Jan 22 13:37:57 2009  vdc_2/
       4096     Jan 22 00:40:57 2009  vdc_3/

       4096     Sep 11 12:54:10 2008  vdc_4/
     111096     Dec 20 04:40:17 2008  vpc.cap
          0     Feb 03 08:02:14 2009  vpc_hw_check_disable
      18166     Apr 03 03:24:22 2009  vpc_vss_apr02
      18223     Apr 02 22:40:57 2009  vss_vpc_apr2
```

```
Usage for bootflash://sup-remote
  863535104 bytes used
  946364416 bytes free
 1809899520 bytes total
N7010-1# copy bootflash://sup
bootflash://sup-1/        bootflash://sup-active/   bootflash://sup-remote/
bootflash://sup-2/        bootflash://sup-local/    bootflash://sup-standby/

N7010-1# copy bootflash://sup-local/congo-s1-epld.4.0.4.img bootflash://sup-
remote/congo-s1-epld.4.0.4.img
N7010-1# dir bootflash://sup-remote
      12349     Dec 05 02:15:33 2008  7k-1-vdc-all.run
       4096     Apr 04 06:45:28 2009  eem/
      18180     Apr 02 23:47:26 2009  eem_script.cfg
      19021     Apr 03 21:04:50 2009  eem_script_counters.cfg
      19781     Apr 05 23:30:51 2009  eem_script_iptrack.cfg
      29104     Jun 19 22:44:51 2009  ethpm_act_logs.log
          0     Jun 19 22:44:51 2009  ethpm_syslogs.log
        175     Jun 20 04:14:37 2009  libotm.log
      49152     Jun 19 22:38:45 2009  lost+found/
   87755113     Apr 07 23:54:07 2009  congo-s1-dk9.4.0.4.bin
   92000595     Apr 16 21:55:19 2009  congo-s1-dk9.4.1.4.bin
   92645614     Apr 08 06:08:35 2009  congo-s1-dk9.4.1.5.bin
   92004757     Jun 02 04:29:19 2009  congo-s1-dk9.4.1.5E2.bin
   99851395     Aug 03 05:20:20 2009  congo-s1-dk9.4.2.0.601.bin
  100122301     Aug 12 04:46:18 2009  congo-s1-dk9.4.2.1.bin
    9730124     Aug 12 22:02:57 2009  congo-s1-epld.4.0.4.img
   10993389     Mar 22 04:55:13 2009  congo-s1-epld.4.1.3.33.img
   23785984     Apr 07 23:47:43 2009  congo-s1-kickstart.4.0.4.bin
   24718848     Apr 16 21:52:40 2009  congo-s1-kickstart.4.1.4.bin
   25173504     Apr 08 06:00:57 2009  congo-s1-kickstart.4.1.5.bin
   23936512     Jun 02 04:26:35 2009  congo-s1-kickstart.4.1.5E2.bin
   25333248     Aug 03 05:19:26 2009  congo-s1-kickstart.4.2.0.601.bin
   25234944     Aug 12 04:45:24 2009  congo-s1-kickstart.4.2.1.bin
        310     Sep 19 03:58:55 2008  n7k-rhs-1.lic
      12699     Jan 23 14:02:52 2009  run_vpc_jan22
      11562     Mar 13 07:52:42 2009  startup-robert-cfg
      16008     Mar 12 02:02:40 2009  startup-vss-cfg
      17315     Mar 19 06:24:32 2009  startup-vss-cfg_roberto_mar18
         99     Apr 04 06:51:15 2009  test1

       9991     Jun 19 23:12:48 2009  vdc.cfg
       4096     Jan 22 13:37:57 2009  vdc_2/
       4096     Jan 22 00:40:57 2009  vdc_3/
```

```
        4096     Sep 11 12:54:10 2008  vdc_4/
      111096     Dec 20 04:40:17 2008  vpc.cap
           0     Feb 03 08:02:14 2009  vpc_hw_check_disable
       18166     Apr 03 03:24:22 2009  vpc_vss_apr02
       18223     Apr 02 22:40:57 2009  vss_vpc_apr2

Usage for bootflash://sup-remote
  873283584 bytes used
  936615936 bytes free
 1809899520 bytes total
N7010-1#
```

Configuration Files: Configuration Rollback

The configuration rollback feature enables you to take a snapshot, or *checkpoint*, of the Cisco NX-OS configuration and then reapply that configuration to your device at any point without reloading the device. Rollback enables any authorized administrator to apply this checkpoint configuration without requiring expert knowledge of the features configured in the checkpoint.

You can create a checkpoint copy of the current running configuration at any time. Cisco NX-OS saves this checkpoint as an ASCII file that you can use to roll back the running configuration to the checkpoint configuration at a future time. You can create multiple checkpoints to save different versions of your running configuration.

When you roll back the running configuration, you can trigger the following rollback types:

- **Atomic:** Implement the rollback only if no errors occur. This is the default rollback type.
- **Best-effort:** Implement a rollback and skip any errors.
- **Stop-at-first-failure:** Implement a rollback that stops if an error occurs.
- **Verbose mode:** Shows the execution log and enables the administrator to see what the switch does during a configuration rollback.

When you are ready to roll back to a checkpoint configuration, you can view the changes that will be applied to your current running configuration before committing to the rollback operation. If an error occurs during the rollback operation, you can choose to cancel the operation or ignore the error and proceed with the rollback. If you cancel the operation, Cisco NX-OS provides a list of changes already applied before the error occurred. You need to clean up these changes manually.

Configuration rollback limitations are as follows:

- You are allowed to create up to 10 checkpoint copies per VDC.
- You are not allowed to apply a checkpoint file of one VDC into another VDC.
- You are not allowed to apply a checkpoint configuration in a nondefault VDC if there is a change in the global configuration portion of the running configuration compared to the checkpoint configuration.
- The checkpoint filenames must be 75 characters or less.
- You are not allowed to start a checkpoint filename with the word *auto*.
- You cannot name a checkpoint file with *summary* or any abbreviation of the word *summary*.
- Only one user can perform a checkpoint or rollback or copy the running configuration to the startup configuration at the same time in a VDC.
- After execution of the **write erase** and **reload** commands, checkpoints are deleted. You can use the **clear checkpoint database** command to clear out all checkpoint files.
- Rollback fails for NetFlow if during rollback you try to modify a record that is programmed in the hardware.
- Although rollback is not supported for checkpoints across software versions, users can perform rollback at their own discretion and can use the best-effort mode to recover from errors.
- When checkpoints are created on bootflash, differences with the running-system configuration cannot be performed before performing the rollback and the system reports No Changes.

Example 1-10 demonstrates how to create a configuration rollback.

Note You need to make sure you are in the correct VDC. If you need to change VDCs, use the **switchto vdc** syntax.

Example 1-10 *Creating a Configuration Rollback*

```
N7010-1# checkpoint changes
...........Done
N7010-1# show diff rollback-patch checkpoint changes running-config
Collecting Running-Config
Generating Rollback Patch
Rollback Patch is Empty
N7010-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
```

```
N7010-1(config)# no snmp-server user nxos-admin
N7010-1(config)# exit
N7010-1# show diff rollback-patch checkpoint changes running-config
Collecting Running-Config
Generating Rollback Patch
!!
no username nxos-admin sshkey ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEA6+TdX+ABH/mq1gQbf-
hhsjBmm65ksgfQb3Mb3qbwUbNlcAa6fjJCGdHuf3kJ
ox/hjgPDChJOd-
kUXHjESlV59OhZP/NHlBrBq0TGRr+hfdAssD3wG5oPkywgM4+bR/ssCzoj6jVG41tGmfPip4pr3dqsMz-
R21DXSK
K/tdj7bipWKy1wSkYQzZwatIVPIXRqTJY7L9a+JqVIJEA0QlJM1l0wZ5YbxccB2GKNKCM2x2BZl4okVgl80C
CJg
7vmn+8RqIOQ5jNAPNeb9kFw9nsPj/r5xFC1RcSKeQbdYAjItU6cX1TslRnKjlWewCgIa26dEaGdawMVuft-
gu0uM
97VCOxZPQ==
no username nxos-admin
N7010-1# rollback running-config checkpoint changes
Note: Applying config in parallel may fail Rollback verification
Collecting Running-Config
Generating Rollback Patch
Executing Rollback Patch
Generating Running-config for verification
Generating Patch for verification
N7010-1# show snmp user nxos-admin
_____________________________________________________________

                   SNMP USER

_____________________________________________________________

User                           Auth  Priv(enforce) Groups
____                           ____  _____________ ______
nxos-admin                     sha   des(no)       network-operator

You can also enable specific SNMP traps:
N7010-1(config)# snmp-server enable traps eigrp
N7010-1(config)# snmp-server enable traps callhome
N7010-1(config)# snmp-server enable traps link
N7010-1(config)# exit
N7010-1#
```

Operating System Files

Cisco NX-OS Software consists of three images:

- **The kickstart image:** Contains the Linux kernel, basic drivers, and initial file system.
- **The system image:** Contains the system software, infrastructure, and Layers 4 through 7.

- **The Erasable Programmable Logic Device (EPLD) image:** EPLDs are found on the Nexus 7000 currently shipping I/O modules. EPLD images are not released frequently; even if an EPLD image is released, the network administrator is not forced to upgrade to the new image. EPLD image upgrades for I/O modules disrupt traffic going through the I/O module. The I/O module powers down briefly during the upgrade. The EPLD image upgrades are performed one module at a time. Starting with NX-OS 5.2.1 and higher, ELPD images can be installed in parallel upgrade on all I/O modules.

On the Nexus 7000 with dual-supervisor modules installed, NX-OS supports ISSU. NX-OS ISSU upgrades are performed without disrupting data traffic. If the upgrade requires EPLD to be installed onto the line cards that causes a disruption of data traffic, the NX-OS software warns you before proceeding so that you can stop the upgrade and reschedule it to a time that minimizes the impact on your network.

NX-OS ISSU updates the following images:

- Kickstart image
- System image
- Supervisor module BIOS
- Data module image
- Data module BIOS
- SUP-1 CMP image
- SUP-1 CMP BIOS

The ISSU process performs a certain sequence of events, as outlined here:

1. Upgrade the BIOS on the active and standby supervisor modules and the line cards (data cards or nonsupervisor modules).
2. Bring up the standby supervisor module with the new kickstart and system images.
3. Stateful Switchover (SSO) from the active supervisor module to the upgraded standby supervisor module.
4. Bring up the old active supervisor module with the new kickstart image and the new system image.
5. Upgrade the CMP on both supervisor modules.
6. Perform a nondisruptive image upgrade for the line card (data cards or nonsupervisor modules), one at a time. With NX-OS 5.2.1 and higher, I/O modules on the Nexus 7000 can be upgraded in parallel.
7. ISSU upgrade is complete.

Virtual Device Contexts

The Nexus 7000 NX-OS software supports Virtual Device Contexts (VDC), which enable the partitioning of a single physical Nexus 7000 device into multiple logical devices. This logical separation provides the following benefits:

- Administrative and management separation
- Change and failure domain isolation from other VDCs
- Address, VLAN, VRF, and vPC isolation

Each VDC appears as a unique device and enables separate Roles-Based Access Control Management (RBAC) per VDC. This enables VDCs to be administered by different administrators while still maintaining a rich, granular RBAC capability. With this functionality, each administrator can define VRF names and VLAN IDs independent of those used in other VDCs safely with the knowledge that VDCs maintain their own unique software processes, configuration, and data plane forwarding tables.

Each VDC also maintains an individual high-availability (HA) policy that defines the action that the system takes when a failure occurs within a VDC. Depending on the hardware configuration of the system, there are various actions that can be performed. In a single supervisor system, the VDC can be shut down or restarted or the supervisor can be reloaded. In a redundant supervisor configuration, the VDC can be shut down or restarted or a supervisor switchover can be initiated.

Example 1-11 shows how to monitor VDC resources.

Example 1-11 *How to Monitor VDC Resources*

```
egypt(config)# show resource

     Resource          Min       Max       Used      Unused    Avail
     --------          ---       ---       ----      ------    -----
     vlan              16        4094      28        0         4066
     monitor-session   0         2         0         0         2
     monitor-session-erspan-dst 0         23        0         0         23

     vrf               2         1000      3         0         997
     port-channel      0         768       1         0         741
     u4route-mem       96        96        1         95        95
     u6route-mem       24        24        1         23        23
     m4route-mem       58        58        1         57        57
     m6route-mem       8         8         1         7         7
egypt#
```

The output shows how much shared memory in a specific VDC is used for a specific type of routes; the **u4route-mem** row indicates memory usage for unicast IPv4 routes. The first five items up to the port-channel are in numbers; the remaining –mem information is in MBs.

Note Refer to Chapter 6, "High Availability," for additional details.

Components are shared between VDCs, which include the following:

- A single instance of the kernel which supports all the processes and VDCs
- Supervisor modules
- Fabric modules
- Power supplies
- Fan trays
- System fan trays
- CMP
- CoPP
- Hardware SPAN resources

Figure 1-5 shows the logical segmentation with VDCs on the Nexus 7000. A common use case is horizontal consolidation to reduce the quantity of physical switches at the data center aggregation layer. There are two physical Nexus 7000 chassis; the logical VDC layout is also shown.

The default VDC is a fully functional VDC with all capabilities. The default VDC has special tasks that are unique to the default VDC. Tasks unique to the default VDC are

- VDC creation/deletion/suspend
- Resource allocation: interfaces and memory
- NX-OS Upgrade across all VDCs
- EPLD Upgrade (for new hardware features)
- Ethanalyzer captures: control plane/data plane (with ACL) traffic
- Feature-set installation for Nexus 2000, FabricPath and FCoE
- CoPP
- Port Channel load balancing
- Hardware IDS checks control

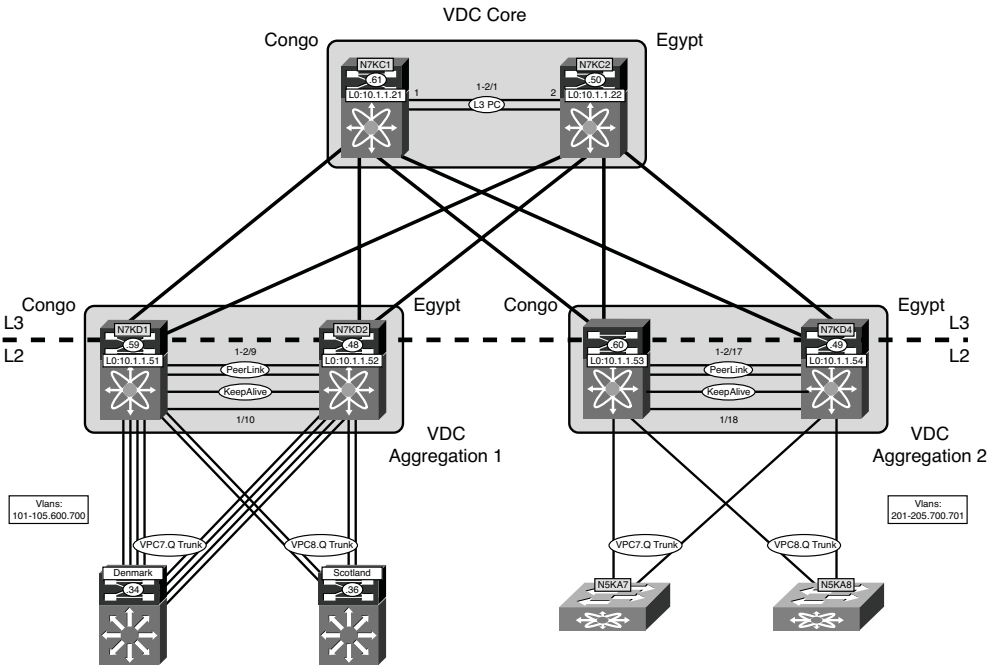

Figure 1-5 *Logical Segmentation with VDCs on the Nexus 7000*

If the operational and administrative requirements or tasks are met, the default VDC can be used for production traffic with no issues; some customers may choose to reserve it for administrative functions.

The non-default VDC is also a fully functional VDC with all capabilities and scale. The VDC feature offers a superset of functionality; the following features are a subset of VDC functionality:

- Changes in nondefault VDC affect only that particular VDC.
- Independent processes started for each protocol in each VDC.
- Discrete configuration file per VDC.
- Discrete checkpoints per VDC.
- Discrete RBAC, TACACS, SNMP, and so on.
- Discrete VLAN, VRF, Spanning-tree control-plane or topology, routing protocols, private-VLANs, and so on.

In NX-OS release 5.1, a module-type parameter that defines the behavior for each VDC was introduced. There are five different I/O module types that can be specified:

- **m1:** Specifies that VDC can contain only M1 modules
- **m1-xl:** Specifies that VDC can contain only M1-XL modules

- **f1:** Specifies that VDC can contain only F1 modules
- **f2:** Specifies that VDC can contain only F2 modules
- **m2xl:** Enables m2 type modules in this VDC

The default VDC is **limit-resource module-type f1 m1 m1-xl m2-xl (default):** It enables a mix of M1, M1-XL, and F1 modules in the VDC. Example 1-12 shows how to create a VDC and limit the resources module type to F1 modules only.

Example 1-12 *Creating VDC Module Type*

```
For an F1-only VDC
egypt# conf t
Egypt (config)# vdc egypt-dc1-fcoe
Egypt(config-vdc)# limit-resource module-type f1
Egypt(config-vdc)# end
egypt#
For an M1/M1-XL-only VDC:
egypt# conf t
Egypt (config)# vdc egypt-dc1-core
Egypt(config-vdc)# limit-resource module-type m1 m1-xl
Egypt(config-vdc)# end
egypt#
For an M1-XLwith F1 Modules for L3 proxy-mode
egypt# conf t
Egypt (config)# vdc egypt-dc1-agg
Egypt(config-vdc)# limit-resource module-type m1-xl f1
Egypt(config-vdc)# end
egypt#
```

Note When configuring these VDC types, the following results will occur based on the following conditions. Conflicting modules are placed in a“suspended” state. With online insertion and removal (OIR), power is supplied, the module is in ok status but the interfaces are not available for configuration. Only VDC allocation is allowed for such interfaces, meaning, to move F1 interfaces from an M1-only VDC to an F1 or mixed-mode VDC.

The introduction of NX-OS 5.2.1 for the Nexus 7000, enables the creation of an FCoE Storage VDC with F1series I/O modules. For FCoE support on F2 and F2-e, a Supervisor-2 or Supervisor-2e must be deployed with NX-OS 6.1.1 and higher. The storage VDC enables traditional Fibre Channel SANs topologies: Fabric A and Fabric B separation. Today, there is support only for one Storage VDC; the default VDC cannot be the

storage VDC. The storage VDC enables separation of job functions for LAN and SAN administrators to preserve current operational models. The storage VDC creates a "virtual" MDS within the Nexus 7000 with the following feature sets:

- Participates as a full FCF in the network.
- Zoning, FC alias, fcdomains, IVR, Fabric Binding, and so on.
- FCoE target support.
- FCoE ISLs to other switches: Nexus 7000, 5000, and MDS.
- Only one storage VDC per chassis.
- Does not require Advanced License (VDC).
- Does count toward total VDC count; currently support for four per Nexus 7000 with Supervisor 1, 4+1 for Supervisor 2, and 8+1 for Supervisor 2e.
- Shared interfaces, exception to the rule allowing an interface to exist in only one VDC. The shared interface concepts are for F1, F2, and F2e modules with a CNA installed in a server. The traffic is based on the L2 Ethertype; Ethernet VDC "owns" the interface, and the storage VDC sees the interface as well.

The shared interface is the only exception to have an interface shared between two VDCs. The shared interface is supported when an F1 interface has a Converged Network Adapter (CNA) attached running FCoE. Depending on the Layer 2 Ethertype, the traffic is separated. There are two components to FCoE. The two Ethertypes are directed only to the storage VDC running FCoE; all other Ethertypes are directed to the Ethernet VDC (non-storage VDC):

1. Control plane, FCoE Initialization Protocol (FIP) Ethertype 0x8914
2. Data plane, FCoE 0x8906

Figure 1-6 shows the shared interface concept with a CNA installed in a server connected to the Nexus 7000 F-Series modules.

Following are the requirements for a shared interface on the Nexus 7000:

- Minimum of NX-OS 5.2(1).
- The interfaces must be on F1, F2, or F2e I/O modules.
- Shared between Default VDC and Storage VDC.
- Shared between nondefault VDC and Storage VDC.
- Ethernet VDC is where the interface is allocated.
- Must be configured as an 802.1q trunk in the Ethernet VDC.
- Both ports on the ASIC must be configured for sharing. Storage VDC is allocated shared interfaces.

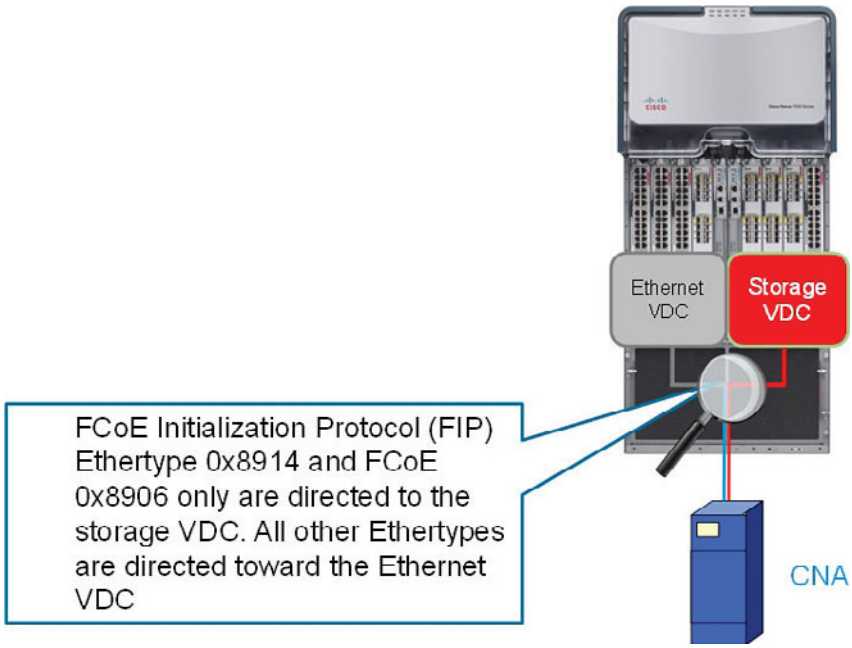

Figure 1-6 *Shared Interface Concept with a CNA Installed in a Server Connected to the Nexus 7000 F-Series Modules*

Example 1-13 shows how to configure shared interfaces on the Nexus 7000.

Example 1-13 *Configuring Shared Interfaces on the Nexus 7000*

```
N7K1-VDC1# config
N7K1-VDC1(config)# vdc fcoe
N7K1-VDC1(config-vdc)# allocate fcoe-vlan-range 2000-2100 from vdc N7K1-VDC1
N7K1-VDC1(config-vdc)# allocate shared interface e3/25-26
Ports that share the port group of the interfaces you have specified will be affected
as well. Continue (y/n)? [yes] yes
N7K1-VDC1(config-vdc)# end
N7K1-VDC1# switchto vdc fcoe
FCoE# show int brief
Eth3/25       1       eth  trunk  down    Administratively down      auto(D) --
Eth3/26       1       eth  trunk  down    Administratively down      auto(D) --
FCoE#
```

Because each VDC is its own switch, to communicate between VDCs, the following criteria must be met:

- Must use front panel port to communicate between VDCs; today there are not soft cross-connect or backplane inter-VDC communications.

- Storage shared ports.
- Front panel ports align security models; ensure QoS, ACL, NetFlow, and so on resources.
- No restrictions on L2/L3 or line card models.
- When using vPC or vPC+ between VDCs, ensure domain IDs are unique.

VDC Configuration

This section shows the required steps to creating a VDC; after the VDC is created, you will assign resources to the VDC. VDCs are always created from the default admin VDC context, VDC context 1.

Note The maximum number of VDCs that can be configured per Nexus 7000 chassis is four with Supervisor-1: the default VDC (VDC 1) and three additional VDCs. Additional VDCs can be configured with the Supervisor-2 and Supervisor-2e. The Supervisor 2 supports four VDC + the admin VDC. The Supervisor-2e supports eight VDCs + the admin VDC. The admin VDC cannot have any data-plane interfaces allocated; only mgmt0 is allowed in the admin VDC.

Example 1-14 shows how to configure the VDC core on Egypt.

Example 1-14 *Creating a VDC Core on Egypt*

```
egypt(config)# vdc core
Note:  Creating VDC, one moment please ...
egypt# show vdc
vdc_id  vdc_name                          state               mac
------  --------                          -----               ----------
1       egypt                             active         00:1b:54:c2:38:c1
2       core                               active        00:1b:54:c2:38:c2

egypt# show vdc core detail
vdc id: 2
vdc name: core
vdc state: active
vdc mac address: 00:1b:54:c2:38:c2
vdc ha policy: RESTART
vdc dual-sup ha policy: SWITCHOVER
vdc boot Order: 2
vdc create time: Mon Feb 22 13:11:59 2010
vdc reload count: 1
vdc restart count: 0
egypt#
```

After the VDC is created, you must assign physical interfaces to the VDC. Depending on the Ethernet modules installed in the switch, interface allocation is supported as follows.

For the 32-port 10-Gigabit Ethernet module (N7K-M132XP-12 and N7K-M132XP-12L) interfaces can be allocated on a per-port-group basis; there are eight port-groups. For example, port-group 1 interfaces are e1, e3, e5, e7; port-group 2 interfaces e2, e4, e6, and e8.

Figure 1-7 shows the Nexus 7000 M1-32-port 10-Gb Ethernet Module.

Figure 1-7 *Nexus 7000 M1-32-Port 10-Gb Ethernet Module*

Figure 1-8 shows the 48-port 10/100/1000 I/O module (N7K-M148GT-11 and N7K-M148GT-11S) can be allocated on a per-port basis.

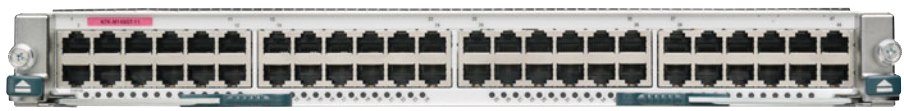

Figure 1-8 *Nexus 7000 M1-48 10/100/1000 Ethernet Module*

Figure 1-9 shows the 48-port 1000BaseX I/O module (N7K-M148GS-11 and N7K-M148GS-11S) can be allocated on a per-port basis.

Figure 1-9 *Nexus 7000 M1 48-Port 1000BaseX Ethernet Module*

Figure 1-10 shows the N7K-F132XP-15, 32-port 1G/10G L2 Only Ethernet module; SFP/SFP+ interfaces will be allocated per 2 ports per VDC. (1–2, 3–4, 5–6...).

Figure 1-10 *Nexus 7000 F1 32-Port L2 Ethernet Module*

Figure 1-11 shows the N7K-M108X2-12L, eight-port 10GbE with an XL option does not have port allocation requirements.

Figure 1-11 *Nexus 7000 M1-08 10-Gb Ethernet Module with Two EARL-8 Forwarding Engines*

Figure 1-12 shows the N7K-F248XP-25 - F2-series I/O module 48-port L2/L3 1/10GE SFP+ Module (req. SFP).

Figure 1-12 *N7K-F248XP-25 - F2-series I/O Module 48-Port L2/L3 1/10GE SFP+ Module (req. SFP)*

Figure 1-13 shows the N7K-F248XT-25e - F2-series I/O module 48-Port L2/L3 1/10-GBase-T RJ45 Module.

Figure 1-13 *Nexus 7000 F2e 48-Port L2/L3 1/10GBase-T Ethernet Module RJ45*

The F2/F2e module is 12 groups of four. The ports will be in sequence (1–4, 5–8, and so on). The F2 module will operate only in a VDC or chassis with other F2 modules. So you could have a chassis composed of only F2 modules in the default VDC and not need the advanced license, or you could have F2 modules in their own VDC in a chassis with other M1/M2 or F1 modules in their own VDC. The F2/F2e module will provide for 550 Gbps of Fabric bandwidth with the Fabric-2 modules, and provide line-rate L2/L3 performance.

Figure 1-14 shows the N7K-M206FQ-23L – M2-Series I/O modules 6-port 40-G Module L2/L3 QSFP.

Figure 1-14 *Nexus 7000 M2-Series I/O modules 6-Port 40Gbe Ethernet Module*

Figure 1-15 shows the N7K-M224XP-23L – M2-Series I/O modules 24-port 10-G Module L2/L3 SFP+.

Figure 1-15 *N7K-M224XP-23L – M2-Series I/O modules 24-port 10G Module L2/L3 SFP+*

Figure 1-16 shows the N7K-M202CF-22L – M2 Series 2-port 100-G Module CFP Optics.

Figure 1-16 *N7K-M202CF-22L – M2 Series 2-Port 100-G Module CFP Optics*

The M2 modules can operate in a VDC or a chassis with M1/F1.

VDC Interface Allocation

Depending on the hardware modules installed in the Nexus 7000 chassis, the interface allocation can vary. The following sections provide the details for each hardware module and port-allocation VDC allocation.

Interface Allocation: N7K-M132XP-12 and L

Interfaces are assigned on a per-VDC basis and cannot be shared across VDCs. After an interface has been assigned to a VDC, all subsequent configuration is done from within that VDC. The N7K-M132XP-12 and L require allocation in port groups of four to align ASIC resources:

- Ports are assigned on a per-VDC basis and cannot be shared across VDCs.
- After a port has been assigned to a VDC, all subsequent configuration is done from within that VDC.
- N7K-M132XP-12 and L require allocation in port groups of four to align ASIC resources.

Figure 1-17 shows the interface allocation for the N7K-M132XP-12 and L modules.

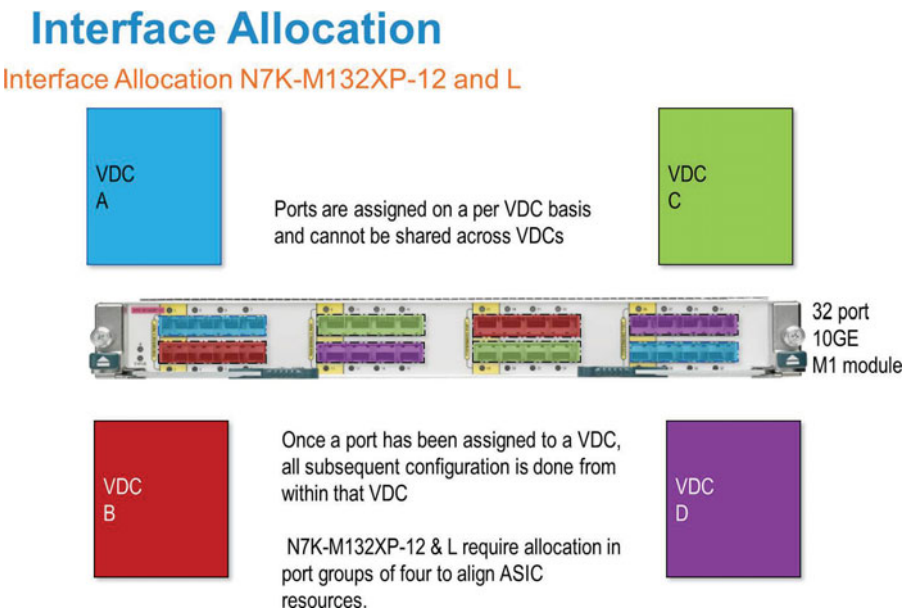

Figure 1-17 *N7K-M132XP-12 and L Module Interface Allocation*

Interface Allocation: N7K-F132XP-15

Interfaces are assigned on a per-VDC basis and cannot be shared across VDCs unless using FCoE. When an interface has been assigned to a VDC, all subsequent configuration is done from within that VDC. The N7K-F132XP-15 requires allocation in port groups of two to align ASIC resources:

- Ports are assigned on a per-VDC basis and cannot be shared across VDCs unless using FCoE.
- After a port has been assigned to a VDC, all subsequent configuration is done from within that VDC.
- N7K-F132XP-15 requires allocation in port groups of two to align ASIC resources.

Figure 1-18 shows the interface allocation for the N7K-F132XP-15 modules.

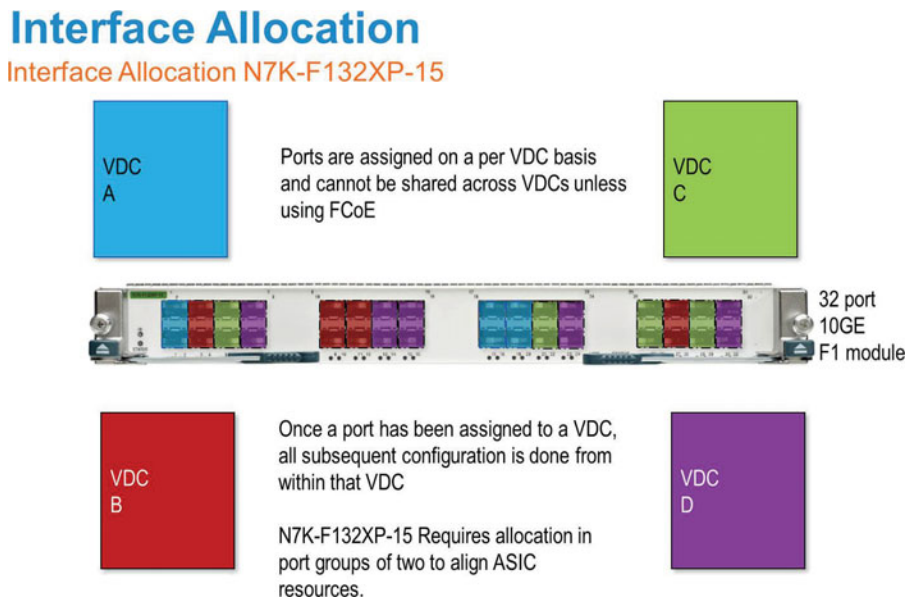

Figure 1-18 *N7K-F132XP-15 Module Interface Allocation*

Interface Allocation: N7K-M108X2-12L

Interfaces are assigned on a per-VDC basis and cannot be shared across VDCs. When a port has been assigned to a VDC, all subsequent configuration is done from within that VDC. Each port on a N7K-M108X2-12L has its own ASIC:

- Ports are assigned on a per-VDC basis and cannot be shared across VDCs.
- After a port has been assigned to a VDC, all subsequent configuration is done from within that VDC.
- Each port on a N7K-M108X2-12L has its own ASIC.

Figure 1-19 shows the interface allocation for the N7K-M108X2-12L modules.

Interface Allocation: 10/100/1000 Modules

Interfaces are assigned on a per VDC basis and cannot be shared across VDCs. After a port has been assigned to a VDC, all subsequent configuration is done from within that VDC. The M1 48 port line cards have four port groups of 12 ports:

- Ports are assigned on a per-VDC basis and cannot be shared across VDCs.
- After a port has been assigned to a VDC, all subsequent configuration is done from within that VDC.
- The M1 48-port line cards have four port groups of 12 ports.
- The recommendation is to have all members of a port group in the same VDC.

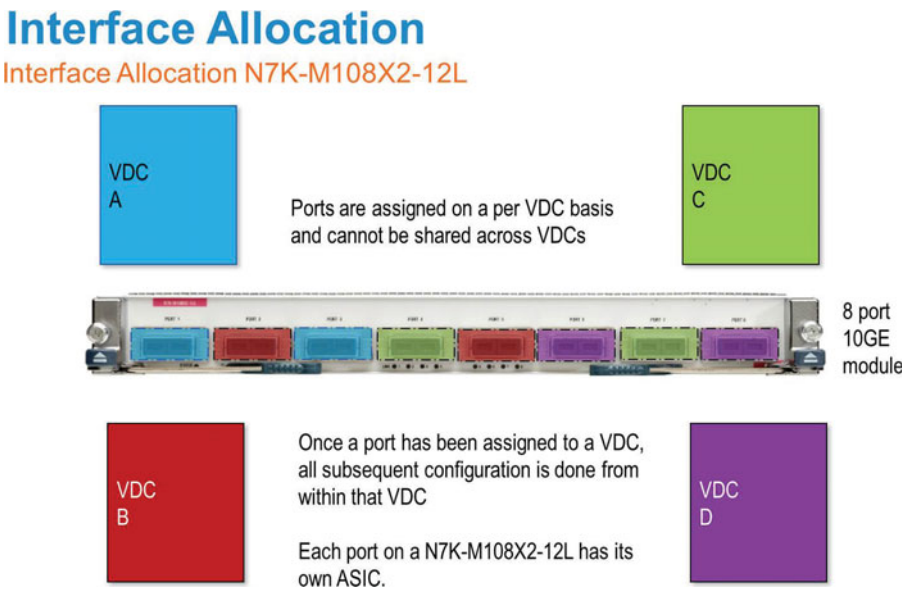

Figure 1-19 *N7K-M108X2-12L Module Interface Allocation*

Figure 1-20 shows the interface allocation for the N7K-M148GS-11 and L and N7K-M148GT-11 and L modules.

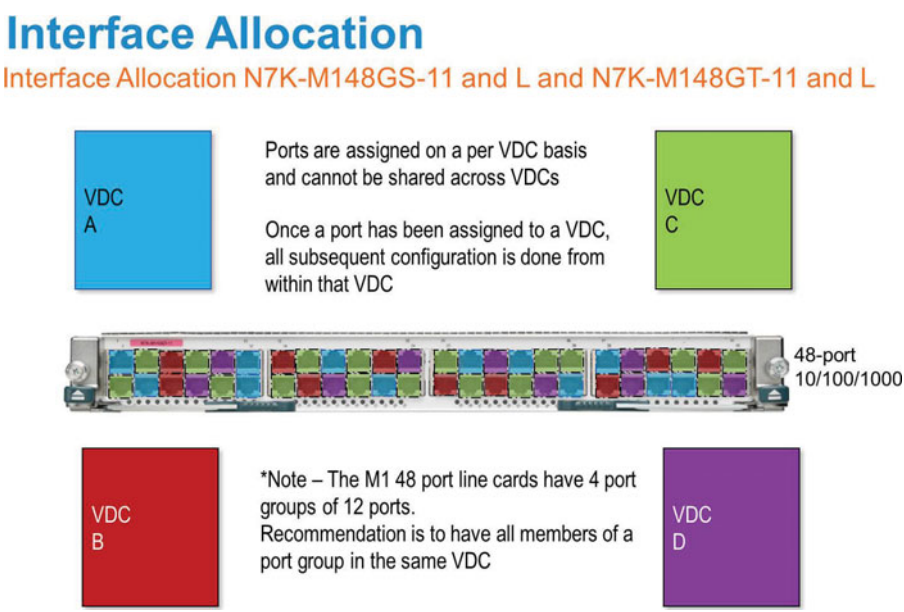

Figure 1-20 *N7K-M148GS-11 and L Module, N7K-M148GT-11 and L Module Interface Allocation*

Figure 1-21 shows the interface allocation for the N7K-F248XP-25 I/O modules.

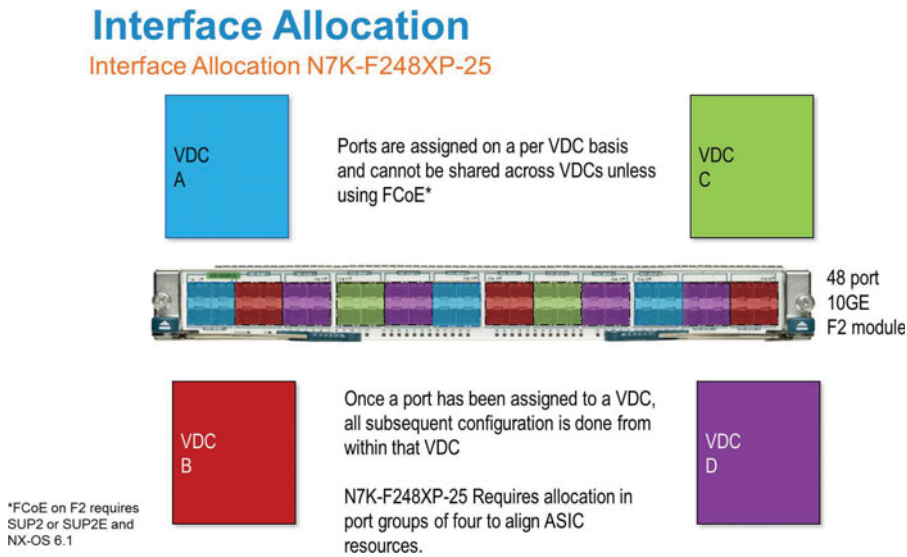

Figure 1-21 *N7K-F248XP-25 I/O Modules Interface Allocation*

Figure 1-22 shows the interface allocation for the N7K-M224XP-23L I/O modules.

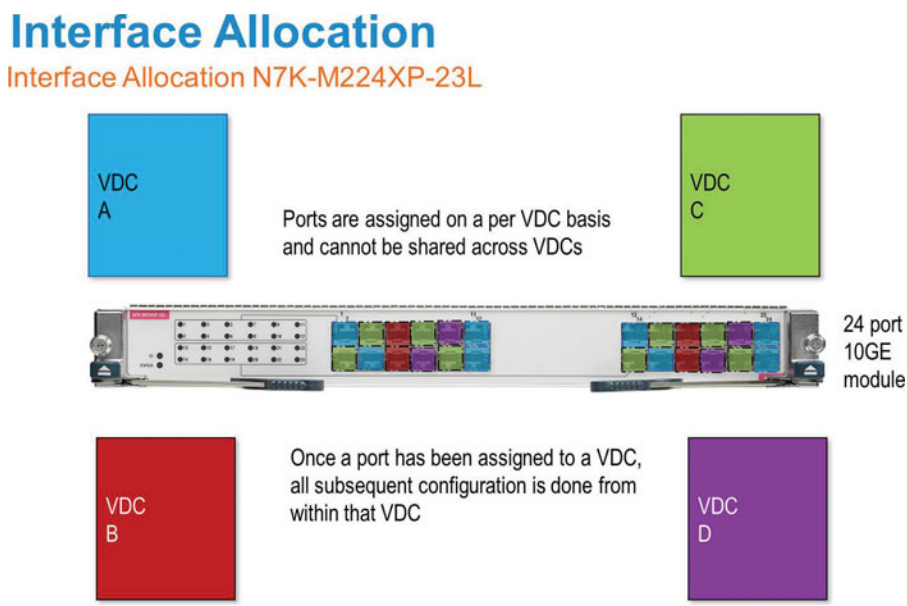

Figure 1-22 *N7K-M224XP-23L I/O Module Interface Allocation*

Figure 1-23 shows the interface allocation for the N7K-M206QF-23L I/O modules.

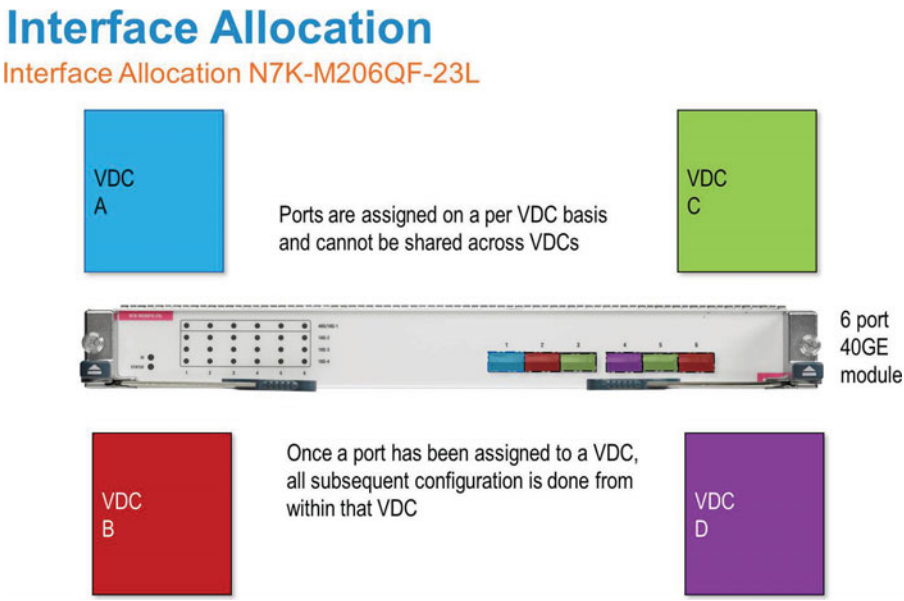

Figure 1-23 *N7K-M206QF-23L I/O Module Interface Allocation*

Figure 1-24 shows the interface allocation for the N7K-M202CF-22L I/O modules.

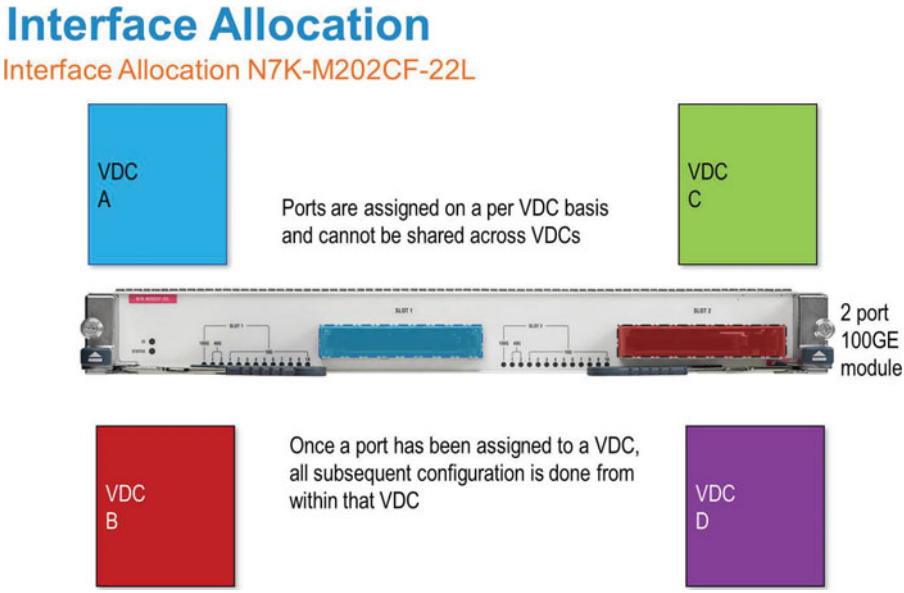

Figure 1-24 *N7K-M202CF-22L I/O Module Interface Allocation*

A common question is, "Can you explain the Nexus 7000 I/O module part number?" For example:

Part Number: N7K-M108X2-12L:

- **N7K:** Nexus 7000 i/o module
- **M1:** M1 forwarding engine
- **08:** Number of interfaces on the module
- **X2:** Optics Interface Type
- **1:** Module h/w version
- **2:** Requires two fabric without N+1 Fabric redundancy
- **L:** XL version and requires XL License

Part Number: N7K-F132XP-15:

- **N7K:** Nexus 7000 i/o module
- **F1:** Fabric module
- **32:** Number of interfaces on the module
- **XP:** Optics Interface Type SFP+
- **1:** Module h/w version
- **5:** Requires five fabric without N+1 Fabric redundancy

Interface Allocation on M2 Modules

On M2 modules, ports are assigned on a per VDC basis and cannot be shared across VDCs.

Note You cannot virtualize a physical interface and associate the resulting logical interfaces to different VDCs. A supported configuration is to virtualize a physical interface and associate the resulting logical interfaces with different VRFs or VLANs. By default, all physical ports belong to the default VDC.

Example 1-15 demonstrates how to allocate interfaces to a VDC.

Example 1-15 *Allocating Interfaces to a VDC*

```
egypt(config)# vdc core
egypt(config-vdc)# allocate interface Ethernet1/17
egypt(config-vdc)# allocate interface Ethernet1/18
```

To verify the interfaces allocation, enter the **show vdc membership** command, as demonstrated in Example 1-16.

Example 1-16 *Verifying Interface Allocation to a VDC*

```
egypt(config-vdc)# show vdc membership

vdc_id: 1 vdc_name: egypt interfaces:
        Ethernet1/26          Ethernet1/28          Ethernet1/30
        Ethernet1/32          Ethernet2/2           Ethernet2/4
        Ethernet2/6           Ethernet2/8           Ethernet2/26
        Ethernet2/28          Ethernet2/30          Ethernet2/32
        Ethernet3/4           Ethernet3/5           Ethernet3/6
        Ethernet3/7           Ethernet3/8           Ethernet3/9
        Ethernet3/11          Ethernet3/12          Ethernet3/13
        Ethernet3/14          Ethernet3/15          Ethernet3/16
        Ethernet3/17          Ethernet3/18          Ethernet3/19
        Ethernet3/20          Ethernet3/21          Ethernet3/22
        Ethernet3/23          Ethernet3/24          Ethernet3/25
        Ethernet3/26          Ethernet3/27          Ethernet3/28
        Ethernet3/29          Ethernet3/30          Ethernet3/31
        Ethernet3/32          Ethernet3/33          Ethernet3/34
        Ethernet3/35          Ethernet3/36          Ethernet3/39
        Ethernet3/40          Ethernet3/41          Ethernet3/42
        Ethernet3/43          Ethernet3/44          Ethernet3/45
        Ethernet3/46          Ethernet3/47          Ethernet3/48

vdc_id: 2 vdc_name: core interfaces:
        Ethernet1/17          Ethernet1/18          Ethernet1/19
        Ethernet1/20          Ethernet1/21          Ethernet1/22
        Ethernet1/23          Ethernet1/24          Ethernet1/25
        Ethernet1/27          Ethernet1/29          Ethernet1/31
        Ethernet2/17          Ethernet2/18          Ethernet2/19
        Ethernet2/20          Ethernet2/21          Ethernet2/22
        Ethernet2/23          Ethernet2/24          Ethernet2/25
        Ethernet2/27          Ethernet2/29          Ethernet2/31
        Ethernet3/1           Ethernet3/2           Ethernet3/3
        Ethernet3/10
```

In addition to interfaces, other physical resources can be allocated to an individual VDC, including IPv4 route memory, IPv6 route memory, port-channels, and SPAN sessions. Configuring these values prevents a single VDC from monopolizing system resources. Example 1-17 demonstrates how to accomplish this.

Example 1-17 *Allocating System Resources*

```
egypt(config)# vdc core
egypt(config-vdc)# limit-resource port-channel minimum 32 maximum equal-to-min
egypt(config-vdc)# limit-resource u4route-mem minimum 32 maximum equal-to-min
egypt(config-vdc)# limit-resource u6route-mem minimum 32 maximum equal-to-min
egypt(config-vdc)# limit-resource vlan minimum 32 maximum equal-to-min
egypt(config-vdc)# limit-resource vrf minimum 32 maximum equal-to-min
```

Defining the VDC HA policy is also done within the VDC configuration sub-mode. Use the ha-policy command to define the HA policy for a VDC as demonstrated in Example 1-18.

Example 1-18 *Changing the HA Policy for a VDC*

```
egypt(config)# vdc core
egypt(config-vdc)# ha-policy dual-sup bringdown
```

The HA policy will depend on the use case or VDC role. For example, if you have dual-supervisor modules in the Nexus 7000 chassis or if the VDC role is development/test, the VDC HA policy may be to just shut down the VDC. If the VDC role is for the core and aggregation use case, the HA policy would be switchover.

Troubleshooting

The troubleshooting sections introduce basic concepts, methodology, and general troubleshooting guidelines for problems that might occur when configuring and using Cisco NX-OS.

show Commands

Table 1-2 lists sample EXEC commands showing the differences between IOS and NX-OS.

Table 1-2 *Sample EXEC Commands Showing the Differences Between IOS and NX-OS*

Operation	IOS	NX-OS
Displays the running configuration	**show running-config**	**show running-config**
Displays the startup configuration	**show startup-config**	**show startup-config**
Displays the status of a specified port-channel interface	**show etherchannel #**	**show port channel #**

Operation	IOS	NX-OS
Displays the current boot variables	**show boot**	**show boot**
Displays all environmental parameters	**show environment**	**show environment**
Displays the percentage of Fabric used per module	**show fabric utilization**	**show hardware fabric-utilization [detail]**
Displays the supervisors high-availability status	**show redundancy**	**show system redundancy status**
Displays CPU and memory usage data	**show process cpu**	**show system resources**
Displays specific VRF information	**show ip vrf** *name*	**show vrf** *name*

debug Commands

Cisco NX-OS supports an extensive debugging feature set for actively troubleshooting a network. Using the CLI, you can enable debugging modes for each feature and view a real-time updated activity log of the control protocol exchanges. Each log entry has a timestamp and is listed chronologically. You can limit access to the debug feature through the CLI roles mechanism to partition access on a per-role basis. Although the **debug** commands show real-time information, you can use the **show** commands to list historical and real-time information.

Caution Use the **debug** commands only under the guidance of your Cisco technical support representative because **debug** commands can impact your network/device performance.

Save **debug** messages to a special log file, which is more secure and easier to process than sending the **debug** output to the console.

By using the **?** option, you can see the options available for any feature. A log entry is created for each entered command in addition to the actual **debug** output. The **debug** output shows a timestamped account of the activity that occurred between the local device and other adjacent devices.

You can use the **debug** facility to track events, internal messages, and protocol errors. However, you should be careful when using the **debug** utility in a production environment because some options might prevent access to the device by generating too many messages to the console or creating CPU-intensive events that could seriously affect network performance.

You can filter out unwanted **debug** information by using the **debug-filter** command. The **debug-filter** command enables you to limit the **debug** information produced by related **debug** commands.

Example 1-19 limits EIGRP hello packet **debug** information to Ethernet interface 1/1.

Example 1-19 *Filtering debug Information*

```
switch# debug-filter ip eigrp interface ethernet 1/1
switch# debug ip eigrp  packets hello
```

Topology

Throughout the book, you see a common topology for demonstration purposes. Figure 1-25 depicts the physical topology.

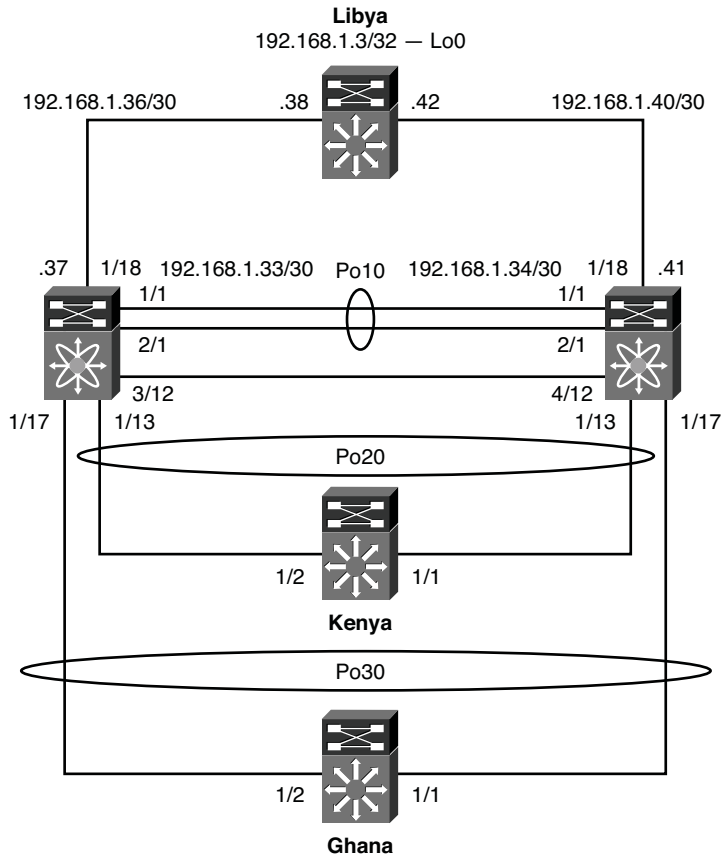

Figure 1-25 *Physical Topology for Book Demonstration Purposes*

Further Reading

NX-OS Nexus 7000 Supported MIB List: ftp://ftp-sj.cisco.com/pub/mibs/supportlists/nexus7000/Nexus7000MIBSupportList.html

- NX-OS Nexus 4000 Support MIB List: ftp://ftp.cisco.com/pub/mibs/supportlists/nexus4000/Nexus4000MIBSupportList.html
- NX-OS Nexus 5000/5500 Supported MIB List: ftp://ftp-sj.cisco.com/pub/mibs/supportlists/nexus5000/Nexus5000MIBSupportList.html
- NX-OS Nexus 1000V Supported MIB List: ftp://ftp.cisco.com/pub/mibs/supportlists/nexus1000v/Nexus1000VMIBSupportList.html
- IOS to NX-OS Conversion tool on CCO: http://tools.cisco.com/nxmt/

NX-OS is a full-featured, modular, and scalable network operating system that enables the entire Cisco Data Center switching portfolio. NX-OS has a modular building-block approach to quickly integrate new innovations and evolving industry standards.

NX-OS helps ensure continuous availability and sets the standard for mission-critical environments. Its self-healing, highly modular design makes zero-impact operations a reality and provides you with exceptional operational flexibility and scalability. Delivering the critical features for next-generation networks, NX-OS is designed with the following requirements: resiliency, virtualization, efficiency, and extensibility.

NX-OS resiliency delivers highly secure, continuous operations, with failure detection, fault isolation, self-healing features, and hitless ISSU that helps reduce maintenance outages.

NX-OS virtualization enhances virtual machine portability and converges multiple services, platforms, and networks to reduce infrastructure sprawl and total cost of ownership.

NX-OS efficiency, operational tools, and clustering technologies reduce complexity and offer consistent features and operations without compromising functionality.

NX-OS extensibility is designed to scale current and future multi-processor hardware platforms and offers easy portability across varying platforms with consistent features. It facilitates the integration of new innovations and evolving standards, delivering long-term feature extensibility.