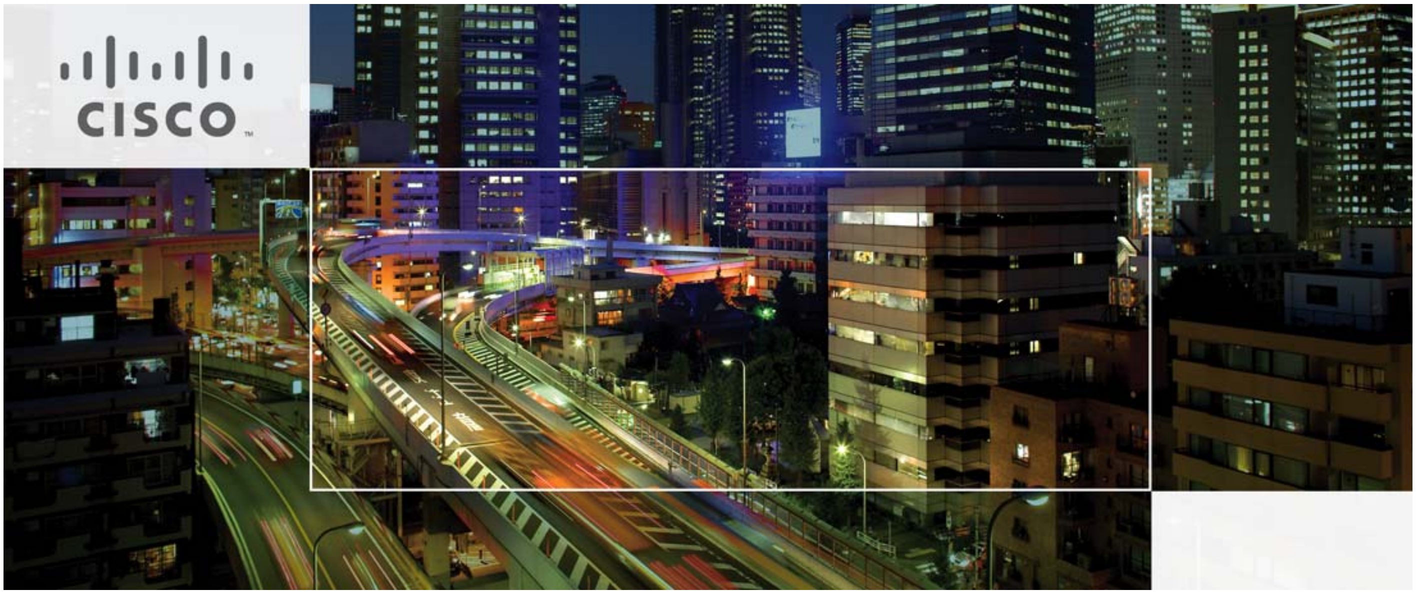


# Cisco Virtualized Multiservice Data Center 2.3 Design Guide

September 30, 2013

Cisco Validated Design

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS” WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

*Cisco Virtualized Multiservice Data Center 2.3 Design Guide*

# CONTENTS

# Preface

The Cisco® Virtualized Multiservice Data Center (VMDC) solution provides design and implementation guidance for Enterprises deploying private cloud services and Service Providers building virtual private and public cloud services. The Cisco VMDC solution integrates various Cisco and third-party products that are part of the cloud computing ecosystem.

Product screen shots and other similar material in this document are used for illustrative purposes only and are VMAX (EMC Corporation), NetApp FAS3240 (NetApp), vSphere (VMware, Inc.), respectively. All other marks and names mentioned herein may be trademarks of their respective companies. The use of the word “partner” or “partnership” does not imply a legal partnership relationship between Cisco and any other company.

## Introduction

Interest in cloud computing over the last several years has been phenomenal. For cloud providers, public or private, it will transform business and operational processes, streamlining customer on-ramping and Time to Market (TTM), facilitating innovation, providing cost efficiencies, and enabling the ability to scale resources on demand.

Infrastructure as a Service (IaaS) simplifies application development and implementation by virtualizing underlying hardware resources and operating systems. This allows IaaS users to significantly cut development and deployment times by cloning the environments best suited for an application without having to factor in the underlying hardware environment. Units of this infrastructure, including compute, storage, and networks, collectively form a cloud infrastructure.

This guide describes design details for a reference architecture that brings together core products and technologies from Cisco, NetApp, EMC, BMC, and VMware to deliver a comprehensive end-to-end cloud solution. Focused on IaaS cloud deployment, the Cisco VMDC solution provides customers with robust, scalable, and resilient options for cloud Data Center (DC) deployments.

Cisco’s VMDC system defines an end-to-end architecture, which an organization may reference for the migration or build out of virtualized, multitenant data centers for new cloud-based service models such as Infrastructure as a Service (IaaS).

The system builds upon these foundational pillars in terms of architectural approach:

- **Secure Multitenancy—**Leveraging traditional security best practices in a multilayered approach to secure the shared physical infrastructure and those logical constructs that contain tenant-specific resources, while applying new technologies to provide security policy and policy mobility to the Virtual Machine (VM) level ensures the continued ability to enforce and comply with business and regulatory policies, even in a highly virtualized multitenant environment.
- **Modularity—**A pod-based modular design approach mitigates the risks associated with unplanned growth, providing a framework for scalability that is achievable in manageable increments with predictable physical and cost characteristics, and allowing for rapid time-to-market through streamlined service instantiation processes.
- **High Availability—**Building for carrier-class availability through platform, network, and hardware and software component level resiliency minimizes the probability and duration of service-affecting incidents, meaning that Private IT and Public Cloud administrators can focus on supporting the bottom line rather than fighting fires.
- **Differentiated Service Support—**Defining logical models around services use cases results in a services-oriented framework for systems definition, ensuring that resources can be applied and tuned to meet tenant requirements.
- **Service Orchestration—**Dynamic application and re-use of freed resources is a key aspect of a Cloud-based operations model, thus the ability to properly represent abstractions of the underlying tenant-specific resources and services is a fundamental requirement for automated service orchestration and fulfillment. This is accomplished in the VMDC architecture through continued evolution of network container definitions which can be leveraged by in-house middleware and partner management solutions.

# Intended Audience

This guide is intended for, but not limited to, system architects, network/compute/storage design engineers, systems engineers, field consultants, advanced services specialists, and customers who want to understand how to deploy a public or private cloud DC infrastructure. This guide assumes that the reader is familiar with the basic concepts of IP protocols, Quality of Service (QoS), High Availability (HA), Layer 4 (L4) - Layer 7 (L7) services, DC platforms and technologies, SAN and VMware hypervisor. This guide also assumes that the reader is aware of general system requirements and has knowledge of Enterprise or Service Provider network and DC architectures and platforms and virtualization technologies.

# Document Organization

Table 1 provides the organization of this guide.

*Table 1 Document Organization*

| Topic | Description |
|---|---|
| Chapter 1, “Design Overview” | This chapter provides an overview of this solution. |

*Table 1 Document Organization*

| | |
|---|---|
| Chapter 2, “Design Details” | This chapter provides the design details of this solution. |
| Glossary | This glossary provides a list of acronyms. |

# Related Documents

The VMDC design recommends that general Cisco DC design best practices be followed as the foundation for IaaS deployments. The following Cisco Validated Design (CVD) companion documents provide guidance on such a foundation:

- VMDC 2.0 Solution Overview
- VMDC 2.0 Solution White Paper
- VMDC 2.1 Design Guide
- VMDC 2.1 Implementation Guide
- VMDC 2.2 Design Guide
- VMDC 2.2 Implementation Guide
- VMDC 2.2 EoMPLS DCI for Hybrid Cloud with vCloud Director
- VMDC 2.3 Implementation Guide
- VMDC 3.0 Design Guide
- VMDC 3.0 Implementation Guide
- Previous VMDC System Releases
- Data Center Designs: Data Center Interconnect
- VMDC Hybrid Cloud with vCloud Director Design and Implementation Guide
- Data Center Design - IP Network Infrastructure
- Data Center Service Patterns
- Data Center Interconnect
- Security and Virtualization in the Data Center
- Vblock Infrastructure Solutions

Cloud Enablement Services from Cisco Advanced Services and partners can help customers realize the full business value of their IT investments faster. Backed by our networking and security expertise, an architectural approach, and a broad ecosystem of partners, these intelligent services enable customers to build a secure, agile, and highly automated cloud infrastructure.

# About Cisco Validated Designs

The Cisco Validated Design Program consists of systems and solutions designed, tested, and documented to facilitate faster, more reliable, and more predictable customer deployments. For more information visit http://www.cisco.com/go/validateddesigns.

CHAPTER 1

# Design Overview

This chapter provides an overview of the Virtualized Multiservice Data Center (VMDC) solution and contains the following topics:

- Introduction
- Cloud Data Center
- Multitenancy Architecture
- Cloud Services

## Introduction

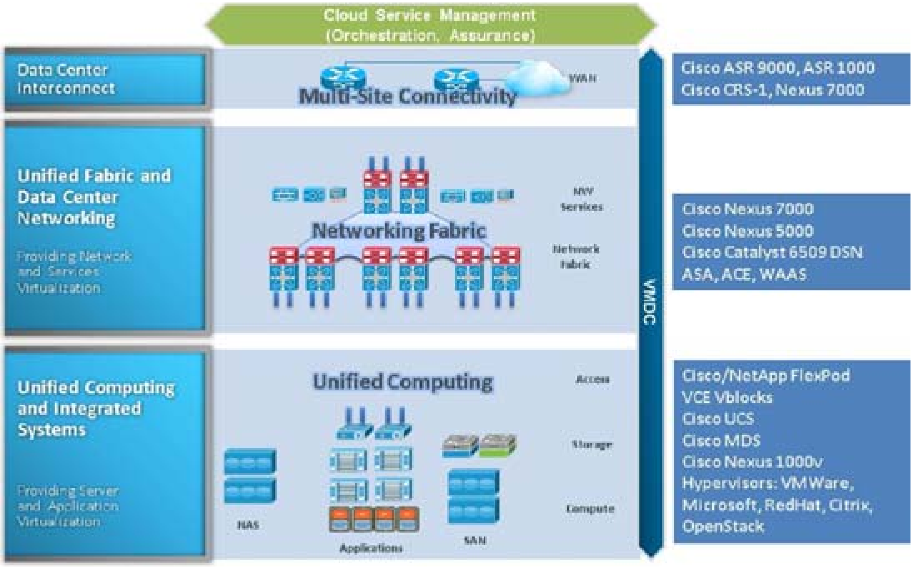
The cloud provides highly scalable, efficient, and elastic services accessed on-demand over the Internet or intranet. In the cloud, compute, storage, and network hardware are abstracted and delivered as a service. End users enjoy the functionality and value provided by the service without the need to manage or be aware of the underlying technology. A cloud deployment model differs from traditional deployments in its ability to treat the Data Center (DC) as a common fabric of resources. A portion of these resources can be dynamically allocated and deallocated when they are no longer in use.

The VMDC solution is the Cisco reference architecture for Infrastructure as a Service (IaaS) cloud deployments. This Cisco cloud architecture is designed around a set of modular DC components consisting of building blocks of resources called pods. These pods are comprised of shared resource pools of network, storage, and compute. Each of these components is virtualized and used by multiple tenants securely, so that each cloud tenant appears to have its own set of physical resources. Cloud service orchestration tools automate the resource provisioning workflow within the cloud DC.

The VMDC solution is targeted towards Enterprises building private clouds and Service Providers building public clouds. In the public cloud case, the tenant would typically be located remotely and have their own DC resources on site in addition to resources within the cloud. In the private case, the tenant could reside locally in another organizational unit logically separated from the IT DC or be located at another facility.

The VMDC system is built around the Cisco Unified Compute System (UCS), Nexus 1000V virtual switches, Multilayer Director Switch (MDS) storage switches, SAN and NAS storage arrays such as NetApp FAS arrays offering NetApp Unified Storage Architecture, Nexus 7000 Aggregation (switching and routing) and Nexus 5000 Access (switching) layers connecting into the Catalyst 6500 Data Center Service Node (DSN), Adaptive Security Appliance (ASA), Application Control Engine (ACE)-based Layer 4 (L4) - Layer 7 (L7) Services layer, and the ASR 9000 and 1000 WAN routers. Cloud orchestration is provided by the BMC Cloud Lifecycle Management (CLM) suite, and cloud assurance by the Zenoss Cloud Service Assurance (CSA) suite. Figure 1-1 shows the functional components of the VMDC solution:

*Figure 1-1 VMDC System Overview*



**Note** Data Center Interconnect (DCI), Cloud Orchestration, and Cloud Assurance are not covered in this document. Please refer to the following documentation for those aspects: Cisco VMDC Documentation on Cisco.com Design Zone.

There have been several iterations of the VMDC solution, with each phase encompassing new platforms, versions, and technologies. The previously released VMDC 2.2 and VMDC 3.0 solutions utilize end-to-end VRF-Lite for tenant segmentation within the cloud DC. The VMDC 2.2 solution utilizes a Virtual Port-channel (vPC)-based Layer 2 (L2) design in the Nexus DC fabric, whereas the VMDC 3.0 solution utilizes a FabricPath-based L2 design in the Nexus DC fabric.

**Note** For more information about previous versions of the VMDC solution, refer to the following documentation:
Cisco VMDC 2.2 Design Guide
Cisco VMDC 3.0 Design Guide

This document focuses on design considerations specific to aspects of the VMDC 2.3-based DC. The VMDC 2.3 solution forms the basis for the Service Provider Cloud Smart Solutions Standard-offer for the Cloud Ready Infrastructure (CRI). The VMDC 2.3 architecture is based on the prior VMDC 2.2 architecture, with some design changes. The key changes in the VMDC 2.3 solution, as compared to the VMDC 2.2 solution, are listed below.

- VMDC 2.2 is built for high server and VM density with up to 3000 servers and 70,000 VMs across six pods.
- VMDC 2.3 is built for small-to-medium server and VM density with up to 768 servers and 24,000 VMs across four pods.
- VMDC 2.3 design has been optimized from the VMDC 2.2 design to achieve higher tenant density with up to 2000 tenants across four pods.
- The Cisco ASR 1000 replaces the Cisco ASR 9000 as the DC WAN router (MPLS-PE).
- VMDC 2.3 does not use the DC Core Nexus 7000 layer.

- Instead of the Nexus 7018 with M1/M2 line cards, VMDC 2.3 uses the Nexus 7004 with F2 line cards as the Aggregation layer.
- VMDC 2.3 does not use the Catalyst 6500 DSN. Instead, services are provided by ASA and ACE appliances directly connecting to the Nexus 7000 Aggregation layer.
- VMDC 2.3 includes optimized tenancy models for Expanded Gold, Silver, and Bronze containers.
- VMDC 2.3 includes a new Copper container for Internet-based cloud access.

# Cloud Data Center

The VMDC-based cloud DC consists of network, storage, and compute resources. The data centers are typically interconnected and provide access to the WAN, IP/Next Generation Network (NGN), or the public Internet. The DC provides multitenancy and multiservices, and also includes management elements for administrative functions, orchestration (cloud portals, Service Catalog, workflow automation), and assurance.

This section discusses the following aspects of the cloud DC:

- Hierarchical Network Architecture
- VMDC Layers
- Modular Building Blocks
- SAN and NAS Architecture
- Compute Architecture

## Hierarchical Network Architecture

Typical DC designs are based upon the classic, multilayer hierarchical network model. In general, such a model implements three layers of hierarchy:

1. A DC Core layer, characterized by a high degree of redundancy and bandwidth capacity, and thus, optimized for availability and performance. The Core layer connects to the DC WAN Edge router.
2. A DC Aggregation layer, characterized by a high degree of high-bandwidth port density capacity, and thus, optimized for traffic distribution and link fan-out capabilities to Access layer switches. Functionally, the nodes in the Aggregation layer typically serve as the L2/L3 boundary. Multiple Aggregation layers can connect to the Core layer, providing for increased density and east-west traffic within the DC.
3. A DC Access layer, serving to connect hosts to the infrastructure, and thus, providing network access, typically at L2 (VLANs).

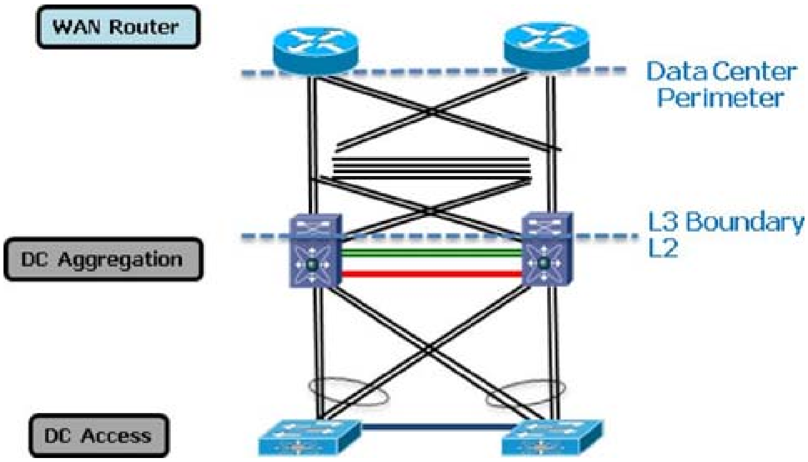
The previous VMDC 2.2 architecture utilized such a three-layer design within the DC, however, for the VMDC 2.3 architecture, the primary goals are to:

1. Increase tenancy scale
2. Reduce cost of the solution
3. Require fewer VMs per tenant (typically 1 VLAN and 4-5 VMs per tenant), targeting the Small, Medium Business (SMB) market for the public or virtual private cloud
4. Require less VM density and east-west bandwidth within a tenant

Based on these requirements, the VMDC 2.3 architecture has been optimized by eliminating the Core layer, as the Core layer adds to cost and reduces tenancy scale due to control plane (Border Gateway Protocol (BGP) and Virtual Routing and Forwarding (VRF)) limits on the platforms. Further, since most tenants are contained within the Aggregation layer (fewer VMs and VLANs per tenant), there is no need to have a Core layer that can provide for routing capabilities between multiple Aggregation layers. Instead, the WAN Edge router can provide the same functionality if needed.

Figure 1-2 illustrates these two layers of the hierarchical model.

*Figure 1-2 VMDC 2.3 Two-Layer Hierarchical Model*

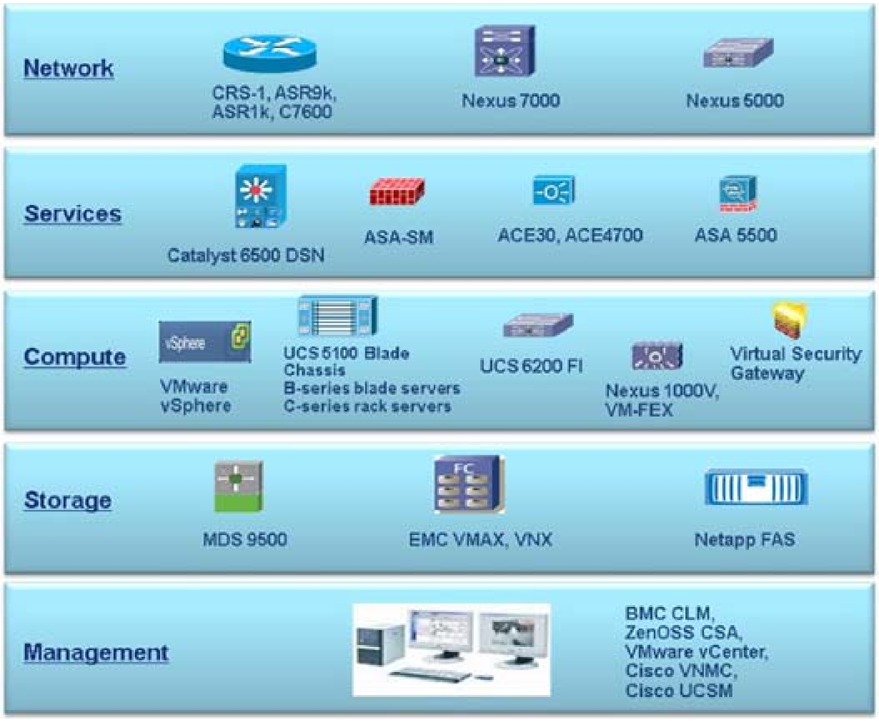


Benefits of such a hierarchical model include scalability, resilience, performance, maintainability, and manageability. The hierarchical design represents a structured approach to building the infrastructure, allowing for relatively easy expansion in modular increments. Redundant nodes and links at each level ensure no single point of failure, while link aggregation can be engineered for optimal bandwidth and performance through the Aggregation and Core layers. Devices within each layer perform the same functions, and this consistency simplifies troubleshooting and configuration. The effect is ease of maintenance at lower operational expense.

# VMDC Layers

Figure 1-3 illustrates the functional layers within the VMDC architecture.

*Figure 1-3 Functional Layers within the VMDC Data Center*



The Network layer includes the WAN/PE router, which forms the DC perimeter to the Enterprise wide area or provider IP/NGN backbone and to the public Internet. These perimeter nodes may be dedicated to L3 routing functions or may be multiservice in nature, providing L2 interconnects between data centers, as well as L3 services. WAN/PE routers validated within the VMDC reference system architecture include the Cisco CRS-1, Cisco ASR 9000, ASR 1000, Cisco 7600, and Cisco Catalyst 6500 platforms. The Network layer also includes the aforementioned, two-layer hierarchy of switching nodes. Within the VMDC reference architecture, this portion of the infrastructure is comprised of Nexus 7000 systems serving as the aggregation nodes, and the Nexus 5000 systems as the access nodes. These systems allow for fine-tuning of port capacity and bandwidth to the level of aggregation or access density required to accommodate current and anticipated scale requirements. In the VMDC 2.3 architecture, the ASR 1000 is used as the WAN/PE router, the Nexus 7004 is used as the aggregation device, and the Nexus 5548 is used as the access device.

The Services layer comprises network and security services such as firewalls, server load balancers, SSL offload, intrusion prevention, network analysis, etc. A distinct difference arises between the conventional DC Services layer and cloud DC Services layer in that the solution set for the latter must support application of L4 - L7 services at a per-tenant level, through logical abstraction of the physical resources. Centralized services are most useful in applying policies that are broadly applicable across a range of tenants (or workgroups in the private case). This layer also serves as the termination point for remote access IPsec or SSL VPNs. Within the VMDC reference architecture, the Catalyst 6500 DSN can provide firewalling and server load-balancing services, in a service module form factor (i.e., the ACE30 and ASASM service modules); alternatively, these are available in appliance form-factors. In the VMDC 2.3 architecture, to keep to smaller footprint and cost, the ASA and ACE appliances serve as the Services layer. Specifically, the ASA 5585-X60 is utilized for firewall services, the ASA 5555X for IPsec/SSL VPN remote access, and the ACE 4710 for Server Load Balancing (SLB).

The Compute layer includes several sub-systems. The first is a virtual Access switching layer, which allows for extension of the L2 network across multiple physical compute systems. This virtual Access switching layer is of key importance in that it also logically extends the L2 network to individual VMs within physical servers. The feature-rich Cisco Nexus 1000V fulfills this role within the architecture. Depending on the level of software functionality (i.e., Quality of Service (QoS) or security policy) or scale required, the Cisco UCS VM-FEX may be a hardware-based alternative to the Nexus 1000V. A second sub-system is that of virtual (i.e., vApp-based) services. These may include security, load balancing, and optimization services. Services implemented at this layer of the infrastructure will complement more centralized service application, with unique applicability directly to a specific tenant or workgroup and their applications or VMs. Specific vApp-based services validated within the VMDC 2.3 architecture include the Cisco Virtual Security Gateway (VSG), which provides security policy enforcement point within the tenant Virtual Data Center (vDC) or Virtual Private Data Center (VPDC). The third sub-system within the Compute layer is the computing resource. This includes physical servers, hypervisor software providing compute virtualization capabilities, and the VMs thus enabled. The UCS, featuring redundant 6200 Fabric Interconnects, UCS 5108 Blade Chassis, and B-Series Blade or C-Series RackMount servers, comprise the compute resources utilized within the VMDC reference architecture.

The Storage layer provides storage resources. Data stores will reside in SAN (block-based) or NAS (file-based) storage systems. SAN switching nodes (MDS) implement an additional level of resiliency, interconnecting multiple SAN storage arrays to the compute resources, via redundant Fibre Channel (FC) (or Fibre Channel over Ethernet (FCoE)) links. The VMDC architecture has been validated with both EMC and NetApp storage arrays and will also work with any other storage vendors.

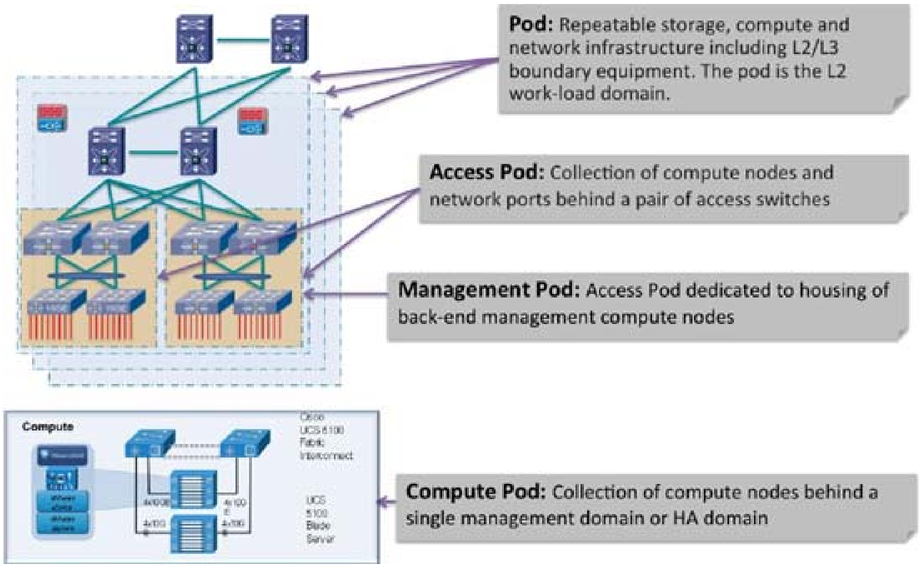
The Management layer consists of the hardware and software resources required to manage the multitenant infrastructure. These include domain element management systems, as well as higherlevel service orchestration systems. The domain management systems that have been validated within VMDC include the UCS Manager (UCSM), VMware vCenter and vCloud Director for compute resource allocation; EMC Unified Infrastructure Manager (UIM), NetApp OnCommand Unified Manager and OnCommand System Manager, NetApp VSC (Virtual Storage Console - a vCenter plugin that provides end-to-end virtual machine (VM) monitoring, provisioning, B&R and management for VMware vSphere environments running on NetApp storage) and Cisco Fabric Manager for storage administration; and the Cisco Virtual Supervisor Module (VSM) and Virtual Network Management Center (VNMC) for virtual access and virtual services management. Automated service provisioning, including cross-resource clooud service orchestration functions, are provided by BMC's Cloud Lifecycle Management (CLM) system; while cloud service assurance is provided by Zenoss Cloud Service Assurance (CSA). Typically, these management systems are hosted in a separate Management pod, so as to isolate the failure domains between the production and management systems.

# Modular Building Blocks

## The Pod

Previous iterations of the VMDC reference architecture defined resource containers called “pods” that serve as the basis for modularity within the cloud DC. As a homogenous modular unit of network, compute, and storage resources, the pod concept allows one to address environmental, physical, logical, and application-level requirements in a consistent way. The pod serves as a blueprint for incremental build-out of the cloud DC in a structured fashion. When resource utilization within a pod reaches a pre-determined threshold (i.e., 70-80%), the idea is that one simply deploys a new pod. From a service fulfillment and orchestration perspective, a pod represents a discrete resource management domain.

*Figure 1-4 Pod Concept*



**Pod:** Repeatable storage, compute and network infrastructure including L2/L3 boundary equipment. The pod is the L2 work-load domain.

**Access Pod:** Collection of compute nodes and network ports behind a pair of access switches

**Management Pod:** Access Pod dedicated to housing of back-end management compute nodes

**Compute Pod:** Collection of compute nodes behind a single management domain or HA domain

In general practice, the pod concept may serve simply as a framework, with designers defining their own variants tuned to specific environmental or performance characteristics. As Figure 1-4 illustrates, a pod can be defined at different levels of modularity, supporting growth in differing increments. Within the VMDC reference architecture, however, a general purpose utility compute pod extends from the Compute and Storage layers to the L2 ports on the aggregation nodes serving as the L2/L3 boundary, and up to and including components within the network Services layer. The port and MAC address capacity of the aggregation nodes are thus key factors in determining how many pods a single pair of aggregation nodes will support within the cloud DC.

## Special Purpose Pods

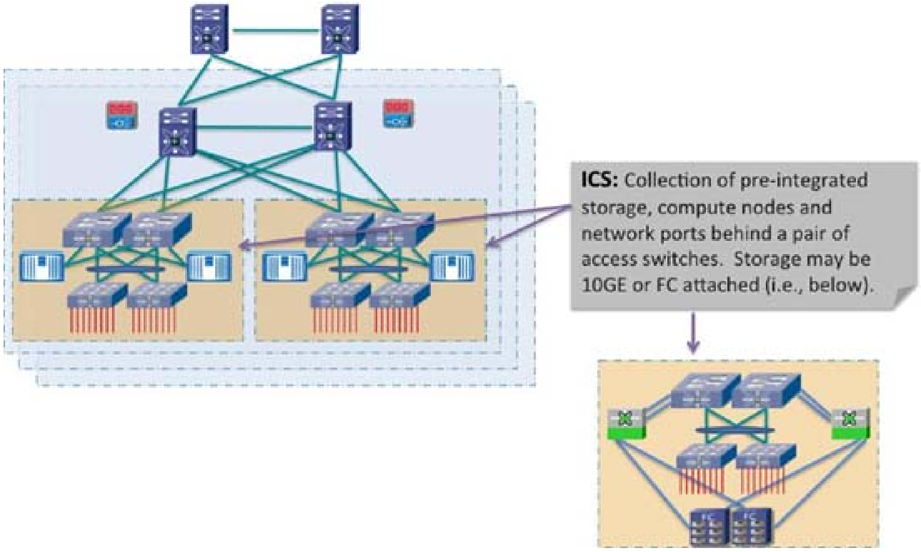
A major premise behind building general purpose homogeneous compute pods and applying logical segmentation overlays to meet business or security policy requirements is that this maximizes utilization of resources, however, in some cases there may be a unique requirement - for ease of operation, special performance tuning, or to meet special security objectives - to physically separate some of the compute nodes out from a general purpose pod and place them in a dedicated, perhaps application-specific pod. The VMDC architecture provides the flexibility to build special purpose pods, and such is the case with the management pod concept.

Back-end management compute nodes may be placed within a general purpose compute pod, and logically isolated and firewalled from production hosts. For smaller, less complex or more streamlined environments, this is an excellent option, however, in larger environments, a separate pod dedicated to back-end management servers (i.e., bare metal and virtualized) is recommended. In the various VMDC 2.X releases, the as-tested systems have in fact included a separate access pod in which servers are dedicated to back-end infrastructure management functions. The benefits of this option include creation of a more discrete troubleshooting domain in the event of instability or failures. The architecture flexibility allows for logical isolation and firewalling or for dedicated firewalls (physical or in vApp form) to be placed on the perimeter of the management container. In practice, Role-based Access Controls (RBAC) tied to directory services would be applied to categorize and limit user access and change control authority as per their functional roles within the organization.

## The Integrated Compute and Storage Stack

An Integrated Compute and Storage (ICS) stack represents another potential unit of modularity within the VMDC cloud DC, representing a sub-component within the pod. An ICS is a pre-integrated collection of storage, compute, and network resources, up to and including L2 ports on a pair of access switching nodes. Figure 1-5 illustrates the location of the ICS within a pod. Multiples instances of an ICS are deployed like building blocks to fill the capacity of a pod.

*Figure 1-5 ICS Concept*


ICS: Collection of pre-integrated storage, compute nodes and network ports behind a pair of access switches. Storage may be 10GE or FC attached (i.e., below).

Working with eco-system partners, Cisco currently supports two ICS stack options, a Vblock and a FlexPod. A Vblock comprises UCS and EMC storage systems, offered in several combinations to meet price, performance, and scale requirements. Similarly, a FlexPod unit combines UCS compute and storage resources, however in this case, NetApp storage systems are used. The VMDC reference architecture will accommodate more generic units of compute and storage, including storage from other third-party vendors, however, the business advantage of an ICS stack is that pre-integration takes the guesswork out of balancing compute processing power with storage Input/Output Operations Per Second (IOPS) to meet application performance requirements.

The Cisco/Netapp FlexPod units are offered in a range of sizes designed to achieve specific workload requirements. The FlexPod architecture is highly modular or "podlike'. Each of the component families of the FlexPod can be scaled both up (adding resources to a FlexPod unit) for greater performance and capacity, and out (adding more FlexPod units) for environments that require consistent, multiple deployments while supporting the same feature-sets and functionality as the base FlexPod.

Some of the key benefits of FlexPod with clustered Data ONTAP are:

- **Non-Disruptive Operations**—Customers never have to deal with a storage outage again. Storage services are always available, even while systems or software are being upgraded or replaced. Immortal data storage infrastructure is now a reality.
- **On-demand Flexibility—**Businesses can scale their storage (both up and out and for performance and capacity), compute, and network resources almost without limits to keep up with today's monumental data growth, all without an interruption in service.

- **Operational Efficiency and Multi-Tenancy**—Organizations can operate more efficiently and become agile by managing multiple systems as a single entity. One storage pool supports a diverse set of applications that companies use to run the business. Storage-related services required to support and protect the business are all automated by using storage service catalogs.

# SAN and NAS Architecture

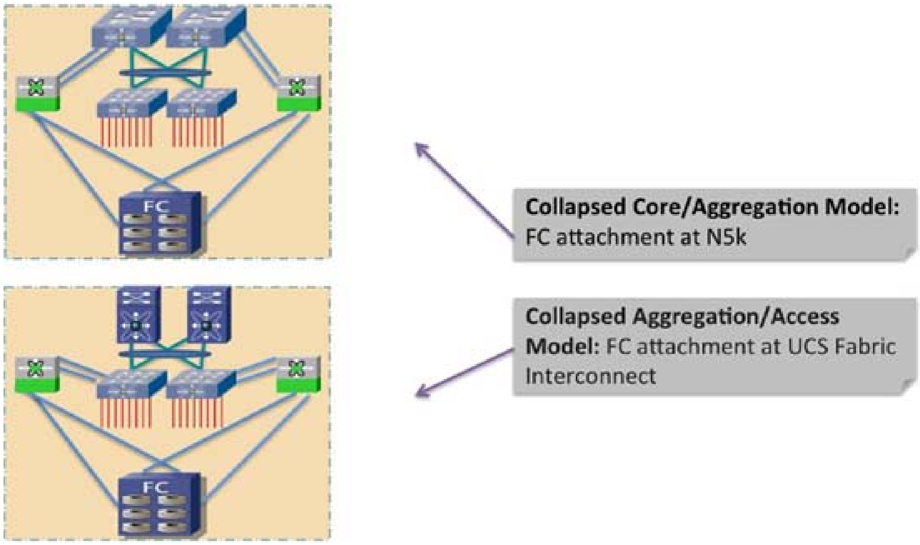
### SAN Architecture

The VMDC 2.3 SAN architecture remains unchanged from previous (2.x) designs. It follows current best practice guidelines for scalability, high availability, and traffic isolation. Key design aspects of the architecture include:

- Leverage of Cisco Data Center Unified Fabric to optimize and reduce LAN and SAN cabling costs
- High availability through multi-level redundancy (link, port, fabric, Director, Redundant Array of Independent Disks (RAID))
- Risk mitigation through fabric isolation (multiple fabrics, Virtual SANs (VSANs))
- Data store isolation through N-Port Virtualization (NPV)/N-Port Identifier Virtualization (NPIV) virtualization techniques, combined with zoning and Logical Unit Number (LUN) masking

The hierarchical, pod-based infrastructure model described in this document lends itself to two possible attachment points for storage, within the pod and/or at the aggregation nodes - i.e., distributed or centralized. In practice, which option is most suitable for a particular deployment will depend on application characteristics and anticipated traffic patterns for interactions involving data store access. Companies often employ both options in order to satisfy specific application requirements and usage patterns. In terms of the VMDC validation work, the focus to date has been on consideration of storage as a distributed, pod-based resource. This is based on the premise that in a hierarchical, cloud-type DC model, it is more efficient in terms of performance and traffic flow optimization to locate data store resources as close to the tenant hosts and vApps as possible. In this context, there are two methods of attaching FC storage components into the infrastructure. The first method follows the ICS model of attachment via the Nexus 5000, and the second method provides for attachment at the UCS Fabric Interconnect. Both methods are illustrated in Figure 1-6.

*Figure 1-6 FC SAN Attachment Options*


FC
Collapsed Core/Aggregation Model: FC attachment at N5k
Collapsed Aggregation/Access Model: FC attachment at UCS Fabric Interconnect
FC

In both scenarios, Cisco's unified fabric capabilities are leveraged with Converged Network Adapters (CNAs) providing "SAN-ready" servers, and N-Port Virtualizer on the UCS Fabric Interconnect or Nexus 5000 Top-of-Rack (ToR) switches enabling each aggregated host to be uniquely identified and managed through the fabric and over uplinks to the SAN systems. In order to match the current maximum processing capability of the SAN system, and thus, eliminate lack of bandwidth between the SAN components and their point of attachment to the network infrastructure as a potential bottleneck, multiple FC links are used from each (redundant) Nexus 5000 or UCS Fabric Interconnect to the MDS SAN switches.

In the FlexPod aligned VMDC 2.3 validation, the NetApp FAS arrays were FC attached to the Nexus 5000 access switch in the FlexPod. For more information on the FlexPod architecture, refer to http://www.netapp.com/us/media/tr-4036.pdf. From a storage/FlexPod automation perspective, OnCommand Workflow Automation (WFA), NetApp's storage automation product, makes common storage management processes simple and easy. Storage experts can easily define common storage management processes like provisioning, setup, migration, and decommissioning, and make them available for execution by approved users. WFA can leverage the current automation policies to demonstrate the value of a "Storage Service Catalog" and can also integrate with the existing orchestration systems. More details can be found at https://communities.netapp.com/community/products_and_solutions/storage_management_software/workflow-automation

Although Figure 1-6 shows a very simplistic SAN switching topology, it is important to note that if greater SAN port switching capacity is required, the architecture supports (and has been validated with) more complex, two-tier core-edge SAN topologies, as documented in the VMDC 2.0 Compact Pod Implementation Guide, and more generally in the Cisco SAN switching best practice guides, available at http://www.cisco.com/en/US/prod/collateral/ps4159/ps6409/ps5990/white_paper_C11-515630.html.

### NAS Architecture

The VMDC 2.3 NAS architecture follows current best practice guidelines for scalability, high availability, and traffic isolation. Key design aspects of the FlexPod architecture include:

- Infrastructure resiliency through multi-level redundancy of FRU components, multipath HA controller configurations, RAID-DP, and software enhancements that help with failures from a software perspective and a hardware perspective.

- Risk mitigation through fabric isolation and multi-level redundancy of connections (multiple fabrics, vPCs or port-channels, interface groups at the storage layer).
- Cisco virtual Port Channels (vPC) address aggregate bandwidth, link, and device resiliency. The Cisco UCS fabric interconnects and NetApp FAS controllers benefit from the Cisco Nexus vPC abstraction, gaining link and device resiliency as well as full utilization of a nonblocking Ethernet fabric. From a storage perspective, both standard LACP and the Cisco vPC link aggregation technologies play an important role in the FlexPod design.
- Network redundancy in clustered Data ONTAP is supported by both the interconnect and the switching fabric, permitting cluster and data and management network interfaces to fail over to different nodes in the cluster, which extends beyond the HA pair.

## Compute Architecture

The VMDC compute architecture is based upon the premise of a high degree of server virtualization, driven by DC consolidation, the dynamic resource allocation requirements fundamental to a cloud model, and the need to maximize operational efficiencies while reducing Capital Expense (CAPEX). The architecture is based upon three key elements:

1. **Hypervisor-based virtualization—**In this release as well as previous system releases, VMware's vSphere plays a key role, enabling the creation of VMs on physical servers by logically abstracting the server environment in terms of CPU, memory, and network touch points into multiple, virtual software containers.
2. **Unified Computing System (UCS)—**Unifying network, server and I/O resources into a single, converged system, the CS provides a highly resilient, low-latency unified fabric for the integration of lossless 10-Gigabit Ethernet and FCoE functions with x-86 server architectures. The UCS provides a stateless compute environment that abstracts I/O resources and server personality, configuration and connectivity, facilitating dynamic programmability. Hardware state abstraction makes it easier to move applications and operating systems across server hardware.
3. **Nexus 1000V—**The Nexus 1000V provides a feature-rich alternative to VMware's Distributed Virtual Switch (DVS), incorporating software-based VN-link technology to extend network visibility, QoS, and security policy to the VM level of granularity.

This system release utilizes VMware's vSphere 5.0 as the compute virtualization operating system. A complete list of new enhancements available with vSphere 5.0 is available online. Key baseline vSphere functionality leveraged by the system includes ESXi boot from SAN, Auto Deploy, VMware High Availability (VMware HA), and Distributed Resource Scheduler (DRS).

Fundamental to the virtualized compute architecture is the notion of clusters. A cluster consists of two or more hosts with their associated resource pools, virtual machines, and data stores. Working in conjunction with vCenter as a compute domain manager, vSphere's more advanced functionality, such as HA and DRS, is built around the management of cluster resources. vSphere supports cluster sizes of up to 32 servers when HA and/or DRS features are utilized. In general practice, however, the larger the scale of the compute environment and the higher the virtualization (VM, network interface, and port) requirement, the more advisable it is to use smaller cluster sizes in order to optimize performance and virtual interface port scale. Therefore, in large VMDC deployments, cluster sizes are limited to eight servers; in smaller deployments, cluster sizes of 16 or 32 can be utilized. As in the VMDC 2.2 release, three compute profiles (Gold, Silver, and Bronze) are created to represent Large, Medium, and Small workload types. Gold has 1 vCPU/core and 16G RAM, Silver has .5 vCPU/core and 8G RAM, and Bronze has .25 vCPU/core and 4G of RAM. The above sizing characteristics are provided as generic reference points in the VMDC architecture and have no bearing to Application or IOPS requirements.

While the VMDC 2.3 architecture works with Vblocks and FlexPods, the system has been validated with FlexPod ICS, which has the following characteristics:

- The architecture comprises multiple UCS 5100 series chassis (5108s), each populated with eight (half-width) server blades.
- Each server has dual 10GigE attachment, i.e., to redundant A and B sides of the internal UCS fabric.
- The UCS is a fully redundant system, with two 2208XP Series Fabric Extenders per chassis connecting up to a pair of UCS 6248UP Fabric Interconnects.
- Internally, four uplinks per Fabric Extender feed into dual Fabric Interconnects to provide the maximum bandwidth possible per server. This means that for server-to-server traffic within the UCS fabric, each server will have 10GigE bandwidth.
- Each UCS 6248 Fabric Interconnect aggregates via redundant 10GigE EtherChannel connections into the access switch (Nexus 5548UP). The number of uplinks provisioned will depend upon traffic engineering requirements. For example, in order to provide an eight-chassis system with an 8:1 oversubscription ratio for internal fabric bandwidth to aggregation bandwidth, a total of 80G (8x10G) of uplink bandwidth capacity must be provided per UCS system.
- For FC connectivity, eight ports on the Nexus 5548 provide 8Gig FC direct connectivity to the NetApp FAS storage arrays. In order to maximize IOPS, the aggregate link bandwidth from the Nexus 5548 to the FAS should match the processing capability of the storage controllers.
- The Nexus 1000V functions as the virtual Access switching layer, providing per-VM policy and policy mobility.

The current version of the CVD is "VMWare vSphere 5.1 on FlexPod Clustered Data ONTAP" which can be found at FlexPod Animal release.

For NAS connectivity, the FlexPod architecture leverages both the Unified Target Adapter (UTA) and the traditional 10GbE Ethernet adapter for storage connectivity. The UTA provides the greatest flexibility when migrating to an end-to-end FCoE design, however, a standard 10GbE can be used for IP-based storage designs. The vPC links between the Nexus 5548 switches and NetApp storage controllers' UTA are converged, supporting both FCoE and traditional Ethernet traffic at 10Gb providing a robust connection between initiator and target. The UTAs installed in each NetApp storage controller use FCoE to send and receive FC traffic to and from the Cisco Nexus switches over 10GbE. The Cisco UCS system also uses FCoE to send and receive FC traffic to and from the various Cisco UCS components (for example, the Cisco UCS B-Series blade servers and Cisco UCS C-Series servers). The FlexPod Animal topology is the first to leverage true end-to-end FCoE, which significantly simplifies the network design and therefore reduces application time to market.

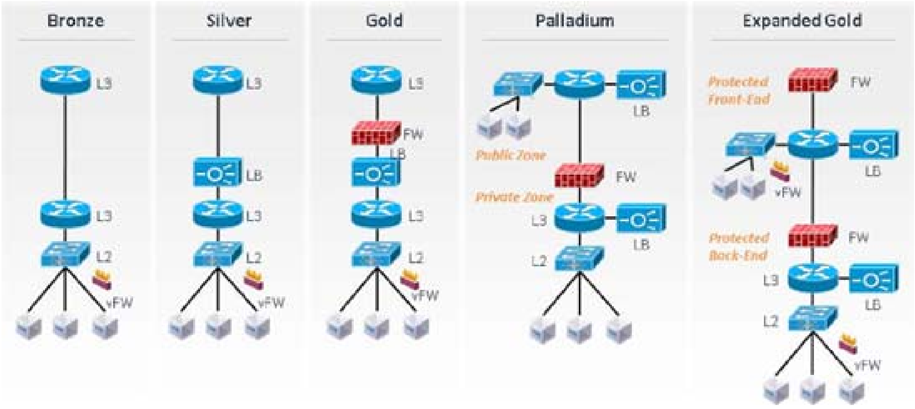
# Multitenancy Architecture

Virtualization of compute and storage resources enables sharing across an organizational entity. In contrast, virtualized multitenancy, a concept at the heart of the VMDC reference architecture, refers to the logical isolation of shared virtual compute, storage, and network resources. In essence, this is “bounded” or compartmentalized sharing. A tenant is a user community with some level of shared affinity. For example, within an Enterprise, a tenant may be a business unit, department, or workgroup. Depending upon business requirements or regulatory policies, a tenant “compartment” may stretch across physical boundaries, organizational boundaries, and even between corporations. A tenant container may reside wholly within their private cloud or may extend from the tenant's Enterprise to the provider's facilities within a public cloud. The VMDC architecture addresses all of these tenancy use cases through a combination of secured datapath isolation and a tiered security model, which leverages classical security best practices and updates them for the virtualized multitenant environment.

# Tenancy Models

Earlier VMDC releases presented five tenancy models or containers. High-level, logical depictions of these models are illustrated in Figure 1-7.

*Figure 1-7 Existing VMDC Tenancy Models*



The first three models provide a baseline, simple set of tenant containers, which were combined with different levels of network services in a tiered fashion, hence the Bronze, Silver, and Gold nomenclature. The two most interesting containers from this set are Bronze and Gold. Bronze is seemingly the most basic, but simplicity broadens its applicability. One tends to think of these containers as single tenant in nature, but in practice, a Bronze container may be used to support multiple tenants, with homogenous requirements, i.e., similar workload profiles, QoS, or security policies, or perhaps this is a community of interest using the same application set.

A Gold container, with both firewall and server load balancer services applied, assumes a higher degree of security and availability. As in the Silver container, multiple VLANs support logical segmentation for N-tiered applications. The idea is that one could combine these tenant containers together in various combinations to support more complex scenarios if desired.

The fourth container type (Palladium) demonstrates a further incremental evolution of tenancy models from simple multisegment containers toward logical approximations of a vDC overlay on the physical shared infrastructure. With the notion of a separate front-end and back-end set of zones, each of which may have a different set of network services applied, the Palladium container begins to more closely align with traditional zoning models in use in physical IT deployments.

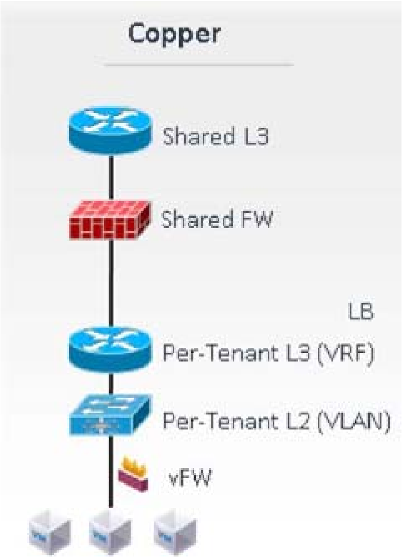
The fifth container type (Expanded Gold container) incrementally evolves the vDC concept, providing more expansion of protected front-end and back-end zones while furthering the notion of separate public (i.e., Internet or Demilitarized Zone (DMZ)) or shared (i.e., campus/inter-organizational) access from private access. It also includes secured remote IPsec or SSL VPN access. In this case, the term “private” can mean that the vDC is routed over the private Enterprise WAN or through the public cloud provider's IP/NGN via a private MPLS VPN. In the public cloud scenario, this type of virtual DC linked to the tenant Enterprise via an L2 or L3 MPLS VPN, is commonly termed a VPDC. MPLS VPNs are often used by public cloud providers as transport for hybrid managed cloud services. Such services may include IP addressing, security (i.e., firewalling, managed DMZ, zoning, secured remote VPN access), and server resiliency solutions.

## New Tenancy Model Introduced in VMDC 2.3

A new tenancy model, the Copper container, is introduced in VMDC 2.3. This tenancy model has been designed to provide higher tenancy scale in VMDC cloud deployments, and is suitable for Internet-based access to cloud resources. The Copper container is relevant to SMBs who require one VLAN and a handful of VMs in the cloud. Such customers require isolation and security, but typically do not wish to pay higher fees for utilizing their own virtual firewall context in the cloud. The Copper container solves this need by utilizing a common firewall shared across such tenants, with each tenant getting their own VLAN and VRF instance for isolation behind the shared firewall.

Figure 1-8 shows the new Copper tenancy model for VMDC 2.3.

*Figure 1-8 New VMDC 2.3 Copper Tenancy Model*



## Storage Multitenancy

From a storage perspective, secure multitenancy is the use of secure virtual partitions within a shared physical storage environment for the purpose of sharing the physical environment among multiple distinct tenants. For instance, a storage SP might configure a storage array in such a way that each of three different customers is provisioned a certain portion of the array's disk capacity and network resources. In a secure multi-tenant environment, each customer would have access only to the resources explicitly provisioned to that customer. The customer would not have access to other customers' data, and not even be aware of the existence of the other customers or the fact that they share a common physical array.

The VMDC 2.3 system has been validated with the FlexPod ICS, with NetApp FAS storage arrays. NetApp FAS arrays can be utilized in clustered Data ONTAP or Data ONTAP 7-Mode configurations. Clustered Data ONTAP is an inherently multi-tenant storage operating system and is architected in such a way that all data access is done through secure virtual storage partitions. It is possible to have a single partition that represents the resources of the entire cluster or multiple partitions that are assigned specific subsets of cluster resources. These secure virtual storage partitions are known as Storage Virtual Machines (SVM). A SVM is effectively isolated from other SVMs that share the same physical hardware. Because it is a secure entity, a SVM is only aware of the resources that have been assigned to it and has no knowledge of other SVMs and their respective resources. Each SVM operates as a separate and distinct entity with its own security domain. Tenants may manage the resources allocated to them through a delegated SVM administration account.

# Cloud Services

Another concept at the heart of the VMDC reference architecture is the notion of differentiated service tiering. Simply put, tenants may have unique requirements in terms of network throughput, compute processing, storage performance, or data store privacy characteristics, and a successful multitenant deployment must be able to address these needs.

## Differentiated Services

By definition, in a cloud-based model, compute, storage, and network infrastructure are abstracted and delivered "as a service." In order to tailor workload characteristics or application performance to specific needs, the cloud administrator has various methods at hand for providing differentiated service tiers and ensuring that tenant privacy and Service Level Agreement (SLA) objectives are met:

- **Tiered Workload Definitions—**The secret to building a cloud-ready infrastructure is in categorizing the set of applications that must be supported and distilling these into their basic workload characteristics. Once these are reasonably understood, they can in most cases be addressed by a set of standard service profiles. For example, characteristics which apply to the ICS include VM attributes (CPU ratio, memory and associated storage capacity), storage attributes (RAID levels, disk types and speeds, and protection mechanisms), and support for various degrees of application tiering.
    - In the context of FlexPod and Netapp FAS storage arrays, the NetApp Virtual Storage Tier is a self-managing data-driven service layer for storage infrastructure that lends itself very well to hosting tiered workloads. VST is natively built into the Data ONTAP operating system and works by leveraging block-sharing technologies such as NetApp primary storage deduplication and file/volume FlexClone to reduce the amount of cache required and eliminate duplicate disk reads. This is extended with Flash Cache and Flash Pool technology, which provides intelligent caching enabling real-time assessment of workloadbased priorities, and enables I/O data requests to be optimized for cost and performance without requiring complex data classification.
- **Availability Mechanisms—**Availability mechanisms may be applied at various layers of the infrastructure to ensure that communication requirements are met. For example, within a vSphere cluster, DRS and vMotion or Fault Tolerance (FT) may be used to provide optimal resource allocation, even in the event of server failure. Similarly, within the SAN, data protection mechanisms such as snapshots, cloning, and backup archiving help to ensure that data store integrity is preserved through various types of failure scenarios. Network services, such as SLB, encryption, advanced routing and redundancy, can further help to achieve availability targets. The larger the shared domain (ICS, pod, or entire DC level), the broader the impact of the availability mechanisms utilized at that particular layer of the hierarchy. As these typically do not come without added cost, the goal would be to ensure that broadly scoped availability methods meet minimum targeted requirements for the entire tenant community.
- **Secure Isolation—**In a multitenant environment, the ability to securely contain and isolate tenant traffic is a fundamental requirement, protecting tenant resources and providing risk mitigation in the event that a specific tenant's privacy is breached. Like availability, isolation mechanisms are applied in a multilayered fashion in order to implement the requisite infrastructure protection and security

zoning policies on a per-tenant basis. In practice, techniques fall into two categories of physical and logical isolation mechanisms, however, VMDC analysis focuses mainly on logical mechanisms. These include various L2 and L3 mechanisms, such as multiple vNICs (i.e., for specific control or data traffic), 802.1q VLANs, MPLS VRF instances, VSANs, combined with access control mechanisms (i.e., RBAC and directory services, IPsec or SSL VPNs), and packet filtering and firewall policies.

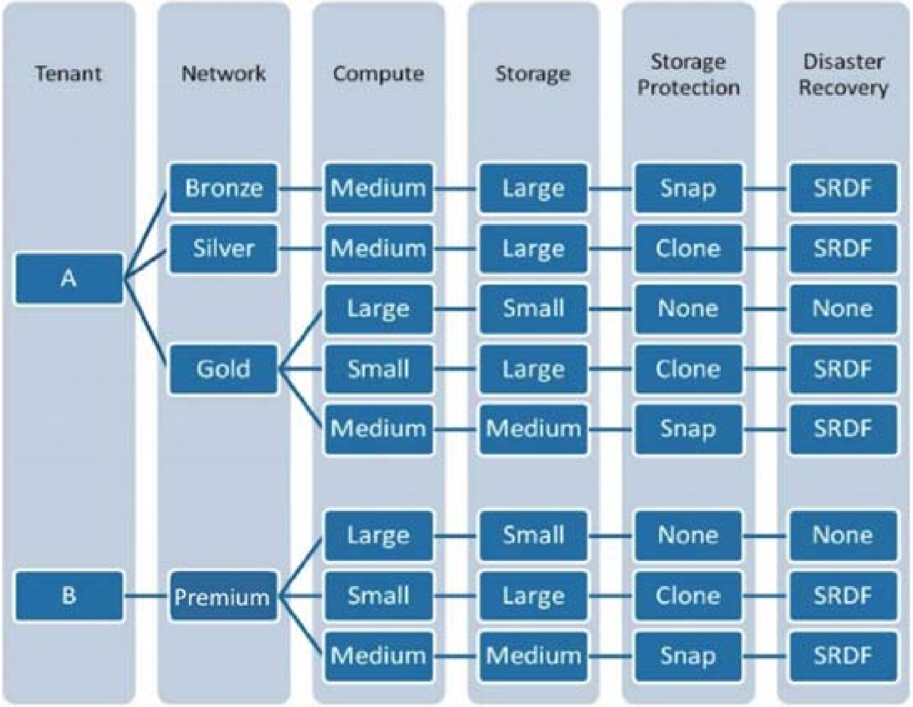
  - From a FlexPod and NetApp storage perspective, clustered Data ONTAP provides a logical mechanism for secure isolation of tenants using Storage Virtual machines (SVMs). Secure multi-tenancy using SVM allows businesses to consolidate tenants into shared resources, and assures that tenants will not have access to resources that are not explicitly assigned to them. Tenants sharing the same physical hardware can operate independently and with the expectation that no single tenant will consume resources unfairly. For example, production and dev/test can run on the same system without the risk of dev/test affecting production workloads. Because it is a secure entity, an SVM is only aware of the resources that have been assigned to it and has no knowledge of other SVMs and their respective resources. Each SVM operates as a separate and distinct entity with its own security domain. Tenants may manage the resources allocated to them through a delegated SVM administration account, and each SVM may connect to unique authentication zones such as Active Directory, LDAP, or NIS.

- **Service Assurance Mechanisms—**Service assurance is a function of availability and QoS policies. The implementation of QoS policies allows for differentiated classification and treatment of traffic flows per tenant, per service tier during periods of congestion.
- **Management—**The ability to abstractly represent per-tenant resources and services in the form of a Service Catalog is a prerequisite for automated service fulfillment and service assurance functions, i.e., the “Day 1” and “Day 2” management tasks which are so essential to operating under an IaaS model. The Service Catalog is effectively the highest level of abstraction for the underlying cloud resources. Accurate representations of these resources as policy-based tenancy models to the service catalog rely on interactions directly with domain element managers or middleware Management layers via standardized interfaces (i.e., APIs, MIBS, etc.). The more intelligent the middleware layer, the less work has to be done at higher levels in the management framework to understand the tenancy models and commission or decommission resources on a per-tenant basis.

## Service Tiering

Previous VMDC releases were modeled based on three baseline categories of tenant network services tiers - Bronze, Silver, and Gold - represented in terms of firewalling, server load balancing, SSL offload, and QoS policy (i.e., three data classes of service), combined with three workload models, each with specific compute attributes, associated storage characteristics, and business continuance services. Figure 1-9 is a high-level conceptual illustration of these models, demonstrating a variety of ways in which these resources and services can be applied in combination to meet business or application requirements in a tiered fashion.

*Figure 1-9 VMDC Service Tiers*



In VMDC 2.3, these definitions are augmented, with the expansion of the Gold service tier to create a premium “Expanded Gold” tier. This premium tier is enabled through a QoS framework which adds SLA support for low latency Voice over IP (VoIP) and multimedia (i.e., video) traffic, in addition to control and differentiated data traffic classes. VMDC 2.3 also introduces an SMB “Copper” tier.

In the context of FlexPod and NetApp storage arrays, refer to the following links for more information on tiering, replication, backup, and DR technologies:

- **Virtual Storage Tiering**: http://www.netapp.com/in/technology/virtual-storage-tier/index.aspx
- **SnapMirror Datasheet**: http://www.netapp.com/in/products/protection-software/snapmirror.aspx
- **SnapMirror Best Practices**: http://www.netapp.com/us/media/tr-4015.pdf
- **SnapVault Datasheet**: http://www.netapp.com/in/products/protection-software/snapvault.aspx
- **SnapVault Best Practices**: http://www.netapp.com/us/media/tr-4183.pdf

CHAPTER 2

# Design Details

The Virtualized Multiservice Data Center (VMDC) 2.3 release continues the end-to-end Virtual Routing and Forwarding (VRF)-Lite and Nexus 7000 Virtual Port-channel (vPC)-based design approach outlined in VMDC 2.2, with some optimizations in physical topology/platforms and tenancy models to achieve higher tenancy scale. This chapter outlines the design details of the VMDC 2.3 solution and consists of the following sections:

- Solution Architecture
- Pod and ICS
- Solution Components
- Secure Tenant Separation
- VMDC 2.3 Containers
- High Availability
- Service Assurance
- Scalability Considerations
- VMDC 2.3 Scale

## Solution Architecture

The VMDC 2.3 system release leverages the end-to-end architecture defined in VMDC 2.0 and 2.2. This document revisits foundational principles of high availability and modular growth, and describes enhancements to the system in the areas of tenant scalability, tenant isolation and security in general, and describes the Quality of Service (QoS) framework for accommodation of multimedia and collaboration applications.

The architecture for this VMDC 2.3 system is based on VMDC 2.2, which utilizes end-end VRF-Lite with per-VRF Border Gateway Protocol (BGP) within the Data Center (DC). VMDC 2.2 defines a hierarchical L3 design with a WAN/PE layer (ASR 9000), Core layer (Nexus 7010), Aggregation layer (Nexus 7018), and a Services layer (Catalyst 6500 with service modules).

To reduce the cost of the overall solution, and to increase scalability (by reducing BGP peering on Nexus 7000) for VMDC 2.3, the following changes have been made:

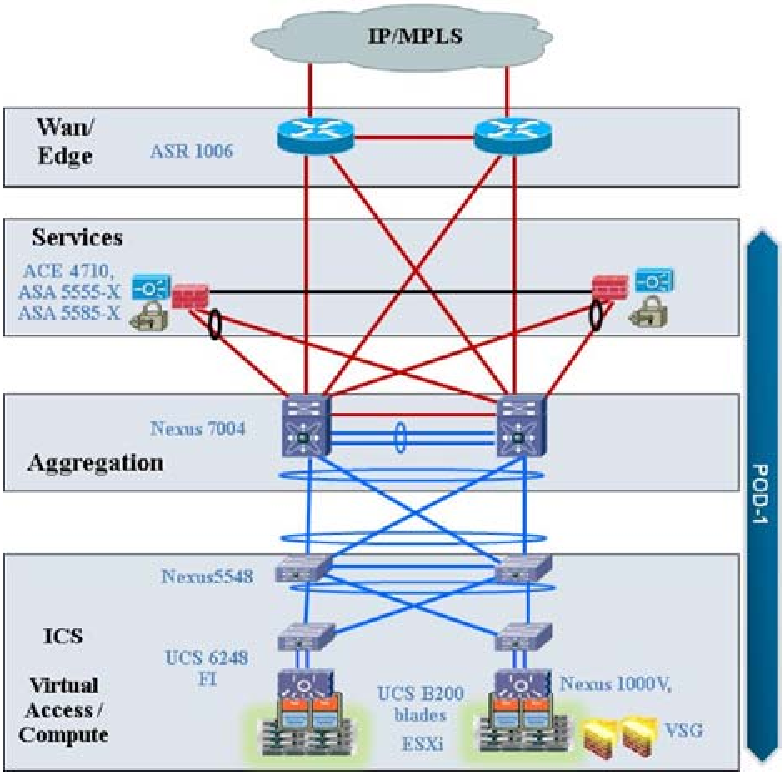
- Utilizes the ASR 1000 as WAN/PE layer
- Eliminates DC Core layer
- Utilizes Nexus 7004 as the Aggregation device

- Uses cheaper F2 line cards on the Nexus 7004
- Eliminates the Catalyst 6500 Data Center Service Node (DSN) and utilizes the Adaptive Security Appliance (ASA) and Application Control Engine (ACE) appliances to connect directly to the Nexus 7004 Aggregation layer

## Physical Topology

Figure 2-1 shows the VMDC 2.3 system architecture from a physical topology perspective. The system consists of 1-3 Integrated Compute and Storage (ICS) stacks (FlexPod or Vblock) connecting to a pair of Nexus 7004 aggregation nodes in a pod. Each ICS is comprised of 1-8 Unified Computing System (UCS) blade systems, a pair of UCS 6248 Fabric Interconnects (FI), and a pair of Nexus 5548UP access switches. There can be 1-4 pods in the VMDC 2.3 DC, with the Nexus 7004 aggregation nodes connecting to an ASR 1006 WAN/MPLS router. Services are provided by security appliances connecting to the Nexus 7004 aggregation nodes. Per-tenant firewall services are provided by firewall contexts on the ASA 5585-X60. Server Load Balancing (SLB) services are provided by ACE 4710 appliances. Remote Access VPN (IPsec and SSL) is provided by ASA 5555X appliances. Compute security is provided by the Virtual Security Gateway (VSG) attached to the Nexus 1000V virtual switch.

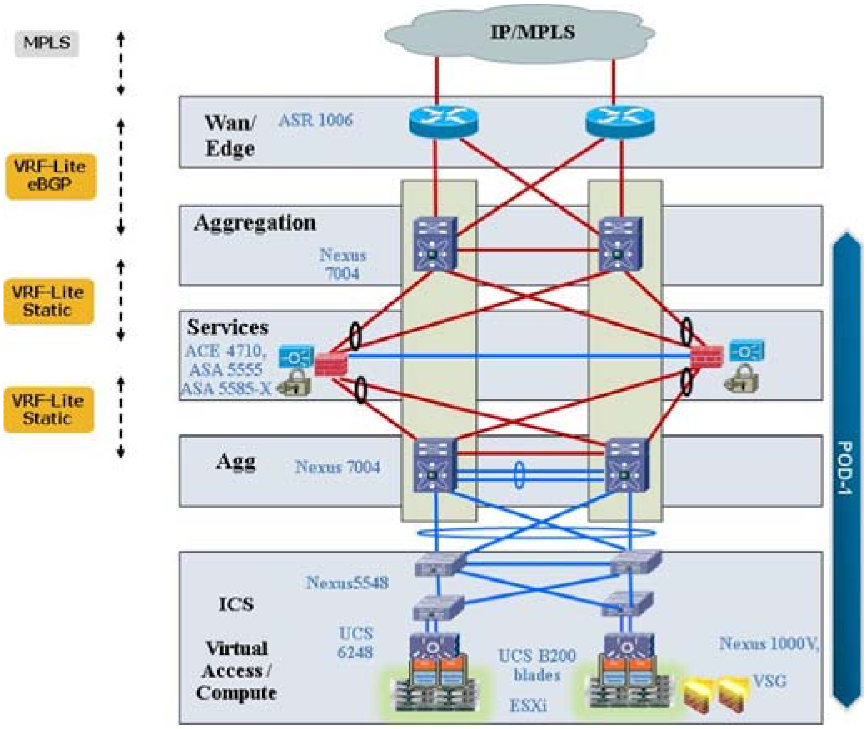
*Figure 2-1 VMDC 2.3 Physical Topology*



# Logical Topology

Figure 2-2 shows the VMDC 2.3 system architecture from a logical topology perspective. When needing to insert an L3 firewall context (on the ASA 5585-X60) to provide perimeter firewall services, the Nexus 7004 aggregation node is logically split into a north VRF and a south VRF instance for each tenant. This applies only to the Gold tenants. For Silver and Bronze tenants, there will no longer be two VRF instances on the Nexus 7004 (to conserve resources on the Nexus and to increase overall solution scalability).

*Figure 2-2 VMDC 2.3 Logical Topology*



the VMDC 2.3 design can be contrasted with the VMDC 2.2 as follows:

- Remains similar to VMDC 2.2, except:
  - Nexus 7004 replaces the Nexus 7018 in the Aggregation layer
  - No Core layer
  - No 6500 DSN
- VMDC 2.3 provides an increased tenancy scale. VMDC 2.2 supports 150 tenants per pod and 150 per DC, while VMDC 2.3 can support 250-500 tenants per pod and 1000-2000 per DC.
- Decreased VM scalability (due to smaller port density Nexus 7000 chassis and lower MAC scale Nexus 7000 line cards). VMDC 2.2 can support 12,000 VMs per pod and 72,000 VMs per DC, while VMDC 2.3 can support up to 6000 VMs per pod and 24,000 VMs per DC.
- Services are provided through appliances connected to the Nexus 7004 instead of modules on the 6500 DSN.
- The ACE 4710 provides SLB (4G). The ASA 5585-X60 provides the firewall (20G multiprotocol). The ASA 5555X provides the RA-VPN service (750 MB).

- Tenancy models remain aligned with VMDC 2.2.
- VMDC 2.3 provides Expanded Gold, Silver, and Bronze network containers.
- VMDC 2.3 introduces a new Copper container to increase tenancy scale and to meet Small, Medium Business (SMB) customer requirements.
- As with previous VMDC releases, the VMDC 2.3 system supports and works with VCE Vblock, Cisco/NetApp FlexPod, or any other ICS stack, but the VMDC 2.3 system was validated with a FlexPod. Previous VMDC releases were validated with Vblock.

# Pod and ICS

## Integrated Compute and Storage Stack

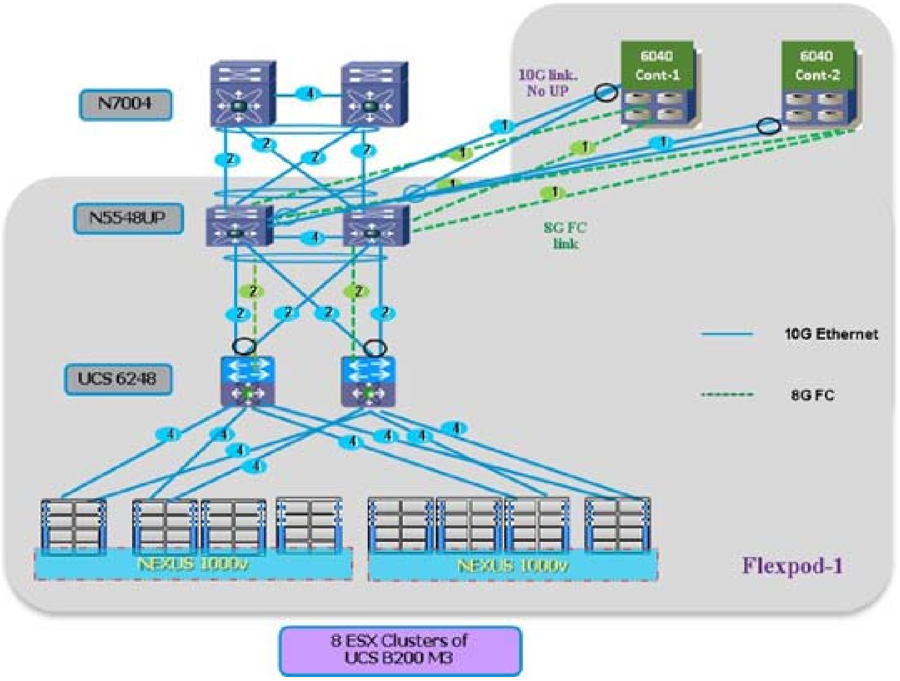
Some of the key design points for the Integrated Compute and Storage (ICS) stack layer of the VMDC 2.3 system architecture are as follows:

- FlexPod consisting of eight UCS chassis, two 6248UP FI, two Nexus 5548UP, and 64 blades of B200 M3.
- VM sizing and distribution will be determined by application requirements. On average, each UCS chassis is sized for 250 VMs (31.5 per blade).
- 2000 VMs per FlexPod (this number is derived by the Nexus 7000 F2 linecard MAC limits, and assuming 2 vNIC per VM).
- Three FlexPods per pod for a total of 6000 VMs in a pod. (The 6000 VM number is derived by the Nexus 7000 F2 linecard MAC limits, and assuming 2 vNIC per VM. The number of FlexPod is shown here as 3, using 64 blades per FlexPod as a reference, however, the number of FlexPods and size of the FlexPod depends on the Application workload.
- NetApp storage array (FAS 6000/6200) self-contained within each FlexPod with NetApp sevenmode configuration.
- Assuming two vNICs per VM for a total of 12,000 vNICs.
- Eight UCS blades in the ESX cluster.
- Two Nexus 1000V virtual switches per FlexPod with six per pod (this number is derived from the Nexus 1000V 2.1 release limit of 2000 vEth ports per VSM).

**Note** For deployment sizing, the application workload and storage requirements need to be considered. The number of VMs supported per FlexPod is determined by the application or workload requirements. A FlexPod unit can be scaled up or scaled out to host all the VMs for a particular Pod depending on the workload. Using a FlexPod at the ICS layers provides the flexibility to scale the ICS layer to a Pod. NetApp storage arrays are self-contained within each FlexPod with clustered Data ONTAP. The array platform will be determined by the type of workloads running on the shared storage platform. Use the following sizing tools available at: FlexPod sizing tool,and NetApp Storage Performance Modeler sizing tool.

Figure 2-3 outlines the ICS (FlexPod) topology and port density/connectivity for the VMDC 2.3 system.

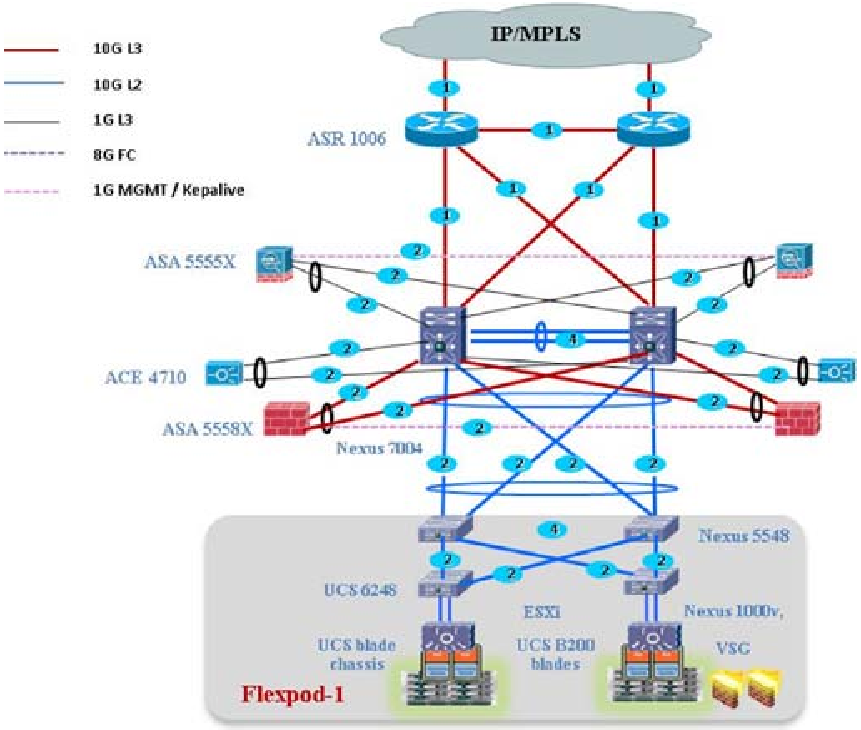
*Figure 2-3 VMDC 2.3 ICS*



## Pod Layout

Figure 2-4 outlines the network topology and port density/connectivity and service appliance attachments within a pod for the VMDC 2.3 system.

*Figure 2-4 VMDC 2.3 Pod - Network and Services Topology*



# Solution Components

The VMDC 2.3 solution consists of several Cisco and third-party hardware and software components. Table 2-1 lists the components that were validated as part of the VMDC 2.3 solution.

*Table 2-1 VMDC 2.3 Solution Components*

| Product | Description | Hardware | Software |
|---|---|---|---|
| Cisco ASR 1000 | WAN (MPLS-PE) Router | ASR 1006 RP-2, ESP-40, SIP-40, SPA-10GE-V2 | IOS XE 3.7.1S |
| Cisco Nexus 7000 | DC Aggregation | Nexus 7004 Sup-2, N7K-F248-12 | NX-OS 6.1(3) |
| Cisco ACE | Application Control Engine (Server Load Balancer) | ACE 4710-MOD-K9 | A 5(2.1) |
| ASA 5555-X | IPsec & SSL VPN remote access | ASA 5555-X | 9.0.1 |
| ASA 5585-X | Adaptive Security Appliance (Firewall Services) | ASA 5585-X60 (with SSP60) | 9.0.1 |
| Cisco Nexus 5548 | Integrated Compute/ Storage Switch | Nexus 5548UP | NX-OS 5.2(1)N1(2) |

*Table 2-1 VMDC 2.3 Solution Components (continued)*

| Product | Description | Hardware | Software |
|---|---|---|---|
| Cisco UCS | Compute System | UCS 5108 blade chassis, UCS 6248 FI, B200-M2 and M3 server blades, Cisco VIC 1240, VIC 1280, M81KR Adapters | 2.0(4b) |
| Cisco Nexus 1010 | Virtual Service Appliance | | NX-OS 4.2(1)SP1(5.1) |
| Cisco Nexus 1000V | Distributed Virtual Switch | | NX-OS<br>4.2(1)SV2(1.1) |
| Cisco VSG | Nexus 1000V Virtual Security Gateway | | 4.2(1)VSG1(4.1) |
| Cisco VNMC | Virtual Network Management Center | | 2.0(3f) |
| NetApp FAS | Unified Storage Array | FAS6040 (Production Pod)<br>FAS3240 (Management Pod) | ONTAP 8.1.1 |
| VMware vSphere ESXi | Hypervisor | | 5.0.0 Build 804277 |
| VMware vSphere vCenter | Virtualization Manager | | 5.0.0 Build 821926 |
| VMware vSphere Auto Deploy | | | 5.0.0.3392 |

**Note** The VMDC 2.3 solution was validated with the ASA 55555-X for IPsec and SSL VPN remote access. For higher performance and throughput, you can also use the ASA 5585-X with SSP-60.

**Note** The NetApp FAS6040 is used as the SAN/NAS storage array in the VMDC 2.3 compute pod to host production (data) VMs. The NetApp FAS3240 is used in the VMDC 2.3 management pod to host management VMs (VMware Virtual Center, Nexus 1000V VSM, VNMC, test tools, BMC Cloud Lifecycle Manager (CLM) orchestration suite, and other management applications).

# Secure Tenant Separation

Traditionally, IT administrators deployed a dedicated infrastructure for their tenants. Deploying multiple tenants in a shared, common infrastructure optimizes resource utilization at lower cost, but requires designs that address secure tenant separation to ensure end-to-end path isolation and meet tenant security requirements. The following design considerations provide secure tenant separation and path isolation:

- Network Separation

- Compute Separation
- Storage Separation
- Application Tier Separation
- Perimeter Security
- DMZ Zones

## Network Separation

In order to address the need to support multitenancy while providing the same degree of tenant isolation as a dedicated infrastructure, the VMDC reference architecture uses path isolation techniques to logically divide a shared infrastructure into multiple (per-tenant) virtual networks. These rely on both data path and device virtualization, implemented in end-to-end fashion across the multiple hierarchical layers of the infrastructure and include:

- **Network L3 separation (Core/Aggregation layers)—**VRF-Lite implemented at the Core and Aggregation layers provides per tenant isolation at L3, with separate dedicated per-tenant routing and forwarding tables ensuring that no inter-tenant (server-to-server) traffic within the DC will be allowed, unless explicitly configured. A side benefit of separated routing and forwarding instances is the support for overlapping IP addresses. This is a required feature in the public cloud case or in merger, or other situations involving IP addressing transitions in the private Enterprise case.
- **Network L2 separation (Access, virtual Access layers)—**VLAN IDs and the 802.1q tag provide isolation and identification of tenant traffic across the L2 domain, and more generally, across shared links throughout the infrastructure.
- **Network services separation (services Core, Compute layers)—**On physical appliance or service module form factors, dedicated contexts or zones provide the means for virtualized security, load balancing, NAT, and SSL offload services and the application of unique per-tenant policies at the VLAN level of granularity. Similarly, dedicated virtual appliances (i.e., in vApp form) provide for unique per-tenant services within the Compute layer of the infrastructure at the VM level of granularity.

## Compute Separation

Traditionally, security policies were implemented at the physical server level, however, server virtualization and mobility introduces new security challenges and concerns. In effect, in order to meet these challenges, policy must be implemented at the VM level and be capable of following VMs as they move from host to host.

Separation of per-tenant traffic in the Compute layer of the infrastructure leverages the following technologies:

- **vNICs—**In the highly virtualized DC, separation of traffic is accomplished via use of multiple vNICs, rather than physical NICs. For example, in VMDC 2.X, multiple vNICs are used to logically separate production (data) traffic from back-end management traffic. This is accomplished with the Cisco UCS Virtual Interface Card (i.e., M81KR VIC in this case), which allows for the creation of virtual adapters and their mapping to unique VMs and VMkernal interfaces within the hypervisor.
- **VLANs—**VLANs provide logical isolation across the L2 domain, including the Nexus 1000V virtual access switching domain within the compute tier of the infrastructure.

- **Port Profiles—**When combined with Cisco's VN-link technology, port profiles provide a means of applying tenant traffic isolation and security policy at the VLAN and VM (vNIC) level of granularity. Implemented at the virtual access switching domain, these map to vCenter port groups, and thus, provide policy mobility through VMotion events.

## Storage Separation

In the VMDC reference architecture, separation of VM data stores within the storage domain of the shared infrastructure is accomplished in the following ways:

- **Cluster File System Management—**The vSphere hypervisor's cluster file system management creates a unique Virtual Machine Disk (VMDK) per VM, ensuring that multiple VMs cannot access the same VMDK sub-directory within the Virtual Machine File System (VMFS) volume, and thus, isolating one tenant's VMDK from another.
- **VSANs and FC Zoning—**Segmentation of the shared SAN fabric into smaller logical domains via VSANs and FC zoning provides isolation at the physical host level of granularity.
- **LUN Masking—**Logical Unit Number (LUN) masking creates an authorization process that restricts storage LUN access to specific hosts on the shared SAN. This, combined with VSANs implemented on the Cisco MDS SAN switching systems plus FC zoning, effectively extends tenant data store separation from the SAN switch ports to the physical disks and virtual media within the storage array.
- **vFilers/Server Virtual Machines**—In clustered Data ONTAP, a Storage Virtual Machine (SVM) contains data volumes and one or more LIFs (logical interfaces which have IPs) through which it serves data to the clients. An SVM securely isolates the shared virtualized data storage and network, and appears as a single dedicated server to its clients. Each SVM has a separate administrator authentication domain and can be managed independently by a SVM administrator. Secure multi-tenancy is provided by network administration and control that is scoped to a particular SVM. Multiple SVMs can coexist in a single cluster without being bound to any node in a cluster. Additional methods for implementing secure customer separation within a FlexPod unit can be found at: https://tech.netapp.com/internal/03/technet_services_solutions_smt.html

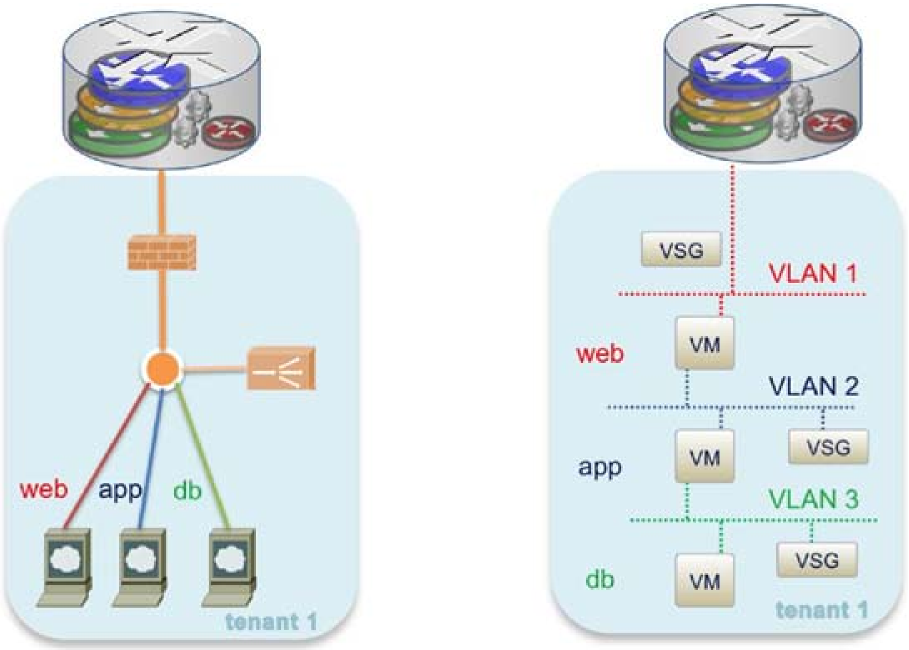
## Application Tier Separation

Many applications follow a three-tiered functional model, consisting of web, application, and database tiers. Servers in the web tier provide the public facing, "front-end" presentation services for the application, while servers in the application and database tiers function as the middleware and back-end processing components. Due to this functional split, servers in the web tier are typically considered to be likely targets of malicious attacks, with the level of vulnerability increasing in proportion to the scope of the user community. Applications meant to be accessible over the public Internet rather than simply remain in the Enterprise private cloud or the Enterprise's VPDC in the public cloud would represent the broadest scope, and thus, a major security concern.

Several methods exist for separation of application tiers:

1. **Network-centric method.** This method involves the use of VLANs within the L2 domain to logically separate each tier of servers (at left in Figure 2-5).
2. **Server-centric method.** This method relies on the use of separate VM vNICs to daisy chain server tiers together (at right in Figure 2-5).

*Figure 2-5 VLAN and vNIC Application Tier Separation*

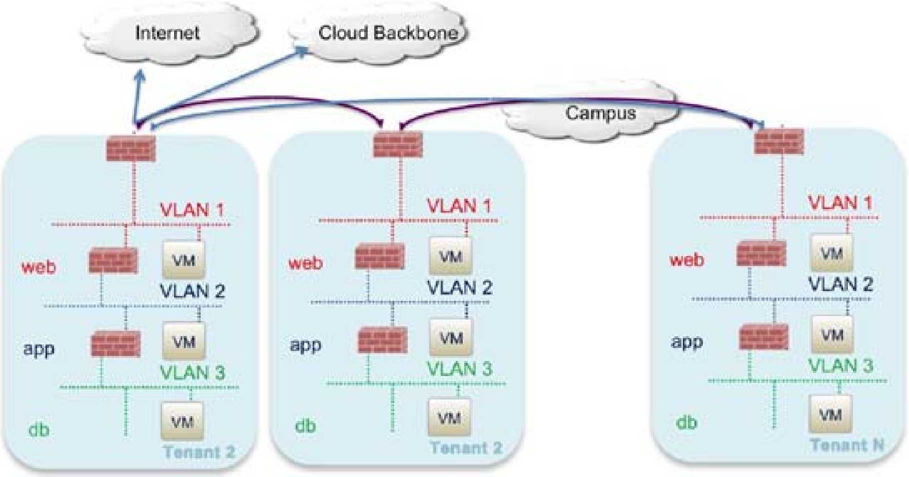


Each method has its pros and cons, and the most desirable method will depend on specific deployment characteristics and operational concerns. From an architectural perspective, network service application will be a major factor. The server-centric method naturally lends itself to vApp-based virtualized service insertion, in Cisco's case, leveraging the Nexus 1000V vPath strengths to classify and more optimally redirect traffic flows at the virtual access switching level of the infrastructure. The network-centric method lends itself to designs in which some or all services are applied from outside the compute tier of the infrastructure, in a Services core layer of the hierarchy, with routing of interVLAN flows. From an administrative perspective, IT executives must consider expertise across the network and server operations staff together with the available management solutions required to support centralized or highly distributed tenant segmentation or service application models.

The network-centric method is the traditional approach. As not all services that one might wish to apply today are available in vApp form, the current trend is a migration from the network-centric model to hybrid-service application scenarios, with some services applied more centrally from the services core and some applied from within the Compute layer of the infrastructure. This is particularly true with respect to security services, where from an operational process and business policy enforcement perspective, it may be necessary to hierarchically deploy policy enforcement points, centralizing and more tightly controlling some while distributing others. This trend is the rationale driving consideration of the hybrid approach to security policy enforcement.

In consideration of application separation, it is common for IT administrators to begin by rigorously separating each tier, assuming that minimal communication between servers on each tier is required. This may sometimes translate to a practice of enforcing separation at each tier with firewalls (see Figure 2-6).

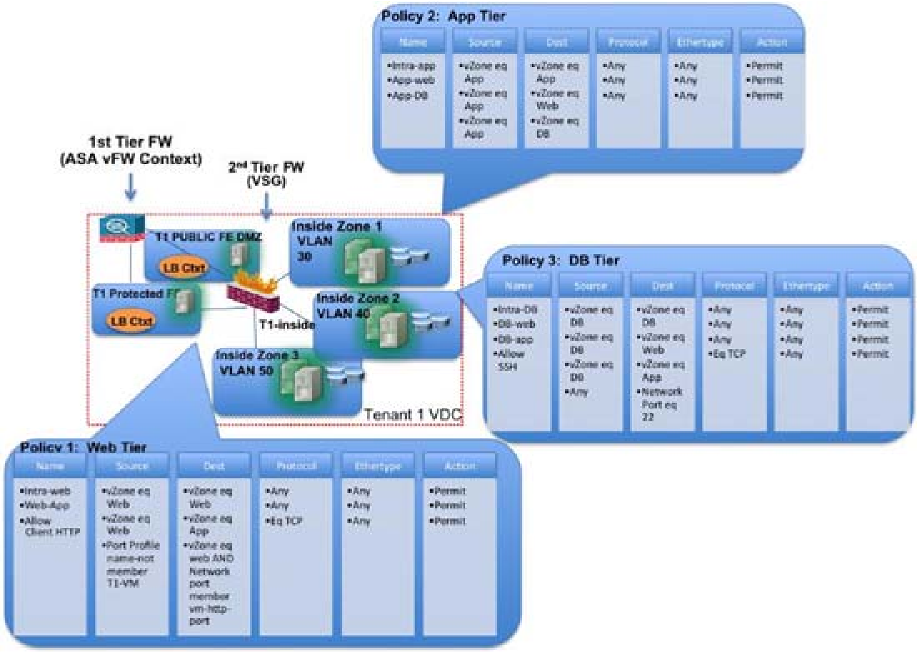
*Figure 2-6 Three-Tier Firewall Example*



While this approach seems reasonable in theory, in practice one soon discovers that it is too simplistic. One problem is that applications are complex and they do not necessarily follow a strict hierarchical traffic flow pattern. Some applications may, for example, be written to function in a database-centric fashion, with communications flows to the middleware (app) and perhaps presentation (web) tiers from a database core, while others may be written to leverage the middleware layer. Another problem, particularly common for Enterprise scenarios, is that some application flows may need to extend outside of the private cloud tenant or workgroup container, across organizational boundaries and perhaps from site-to-site. Finally, application tiers may themselves be distributed, either logically or physically, across the DC, or in the private case, across the Enterprise campus. The result is unnecessary and sub-optimal proliferation of policy enforcement points, in which traffic may needlessly be required to traverse multiples of firewalls on the path end-to-end from source to destination.

With a hybrid two-tiered firewall model (Figure 2-7), the VMDC architecture seeks to provide a simplified framework that mitigates firewall proliferation over the physical and virtualized infrastructure while allowing for defense-in-depth, as per traditional security best practices. As noted earlier, a benefit of this framework is that it enables hierarchical policy definition, with rigorous, fine-grained enforcement at the outer edges of the tenant container and more permissive, coarse-grained enforcement within the tenant container. This framework also provides a graceful transition from physical to virtual policy enforcement, allowing cloud administrators to leverage existing inventory and expertise.

*Figure 2-7 VMDC Two-Tier Hybrid Tenant Firewall Model*

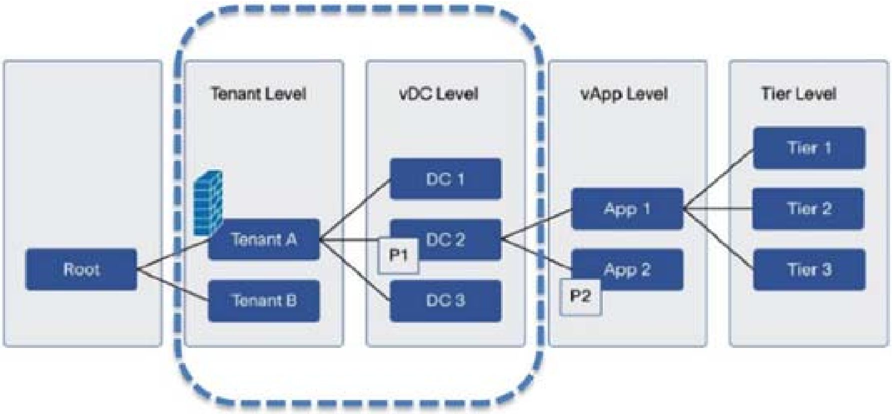

Policy 2: App Tier
Name
Source
Dest
Protocol
Ethertype
Action
•Intra-app •App-web •App-DB
•vZone eq App •vZone eq App •vZone eq App
•vZone eq App •vZone eq Web •vZone eq DB
•Any •Any •Any
•Any •Any •Any
•Permit •Permit •Permit
1st Tier FW (ASA vFW Context)
2nd Tier FW (VSG)
T1 PUBLIC FE DMZ
LB Ctxt
T1 Protected FE
LB Ctxt
Inside Zone 1 VLAN 30
Inside Zone 2 VLAN 40
T1-inside
Inside Zone 3 VLAN 50
Tenant 1 VDC
Policy 3: DB Tier
Name
Source
Dest
Protocol
Ethertype
Action
•Intra-DB •DB-web •DB-app •Allow SSH
•vZone eq DB •vZone eq DB •vZone eq DB •Any
•vZone eq DB •vZone eq Web •vZone eq App •Network Port eq 22
•Any •Any •Any •Eq TCP
•Any •Any •Any •Any
•Permit •Permit •Permit •Permit
Policy 1: Web Tier
Name
Source
Dest
Protocol
Ethertype
Action
•Intra-web •Web-App •Allow Client HTTP
•vZone eq Web •vZone eq Web •Port Profile name-not member T1-VM
•vZone eq Web •vZone eq App •vZone eq web AND Network port member vm-http-port
•Any •Any •Eq TCP
•Any •Any •Any
•Permit •Permit •Permit

## Virtual Security Gateway

The Virtual Security Gateway (VSG) is a new addition to the VMDC reference architecture. In the VMDC architecture, inter-tenant communication (if allowed) is established through routing at the Aggregation layer, however, Figure 2-8 illustrates how the VSG virtual security appliance fulfills the functional role of an intra-tenant second tier firewall to filter communication between and within application tiers and from client to server. Tightly integrated with the Nexus 1000V Distributed Virtual Switch (DVS), the VSG uses the virtual network service path (vPath) technology embedded within the Nexus 1000V Virtual Ethernet Module (VEM). The vPath capability within the Nexus 1000V offloads the switching logic directly to the host, providing high performance, seamless interaction with other virtual appliances, and resiliency in case of appliance failure. There is a significant performance improvement, since most of the packets are offloaded to the hypervisor and processed by the fast path. In addition, the Cisco Nexus 1000V vPath is tenant-aware, which allows for the implementation of security policies within and across multiple tenants.

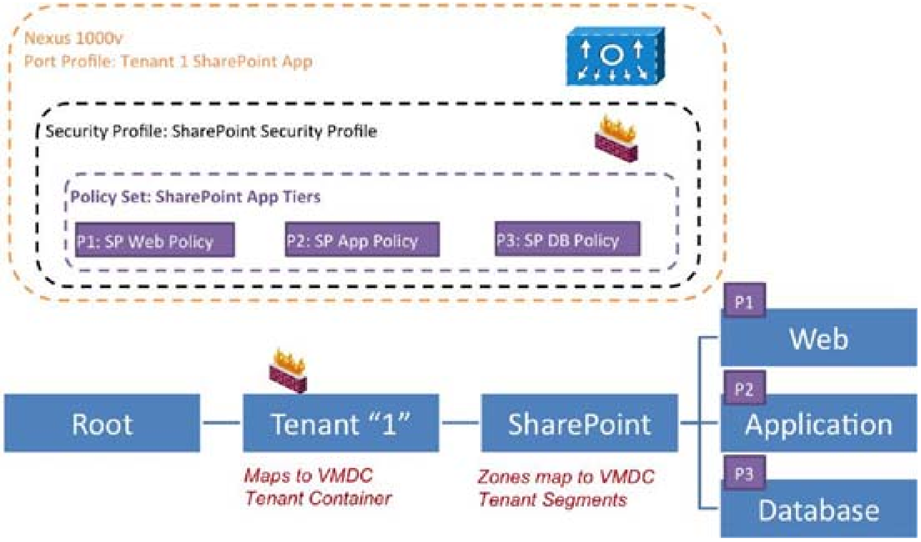
The VSG multitenant support relies on a hierarchical policy model (Figure 2-8). This model allows each tenant to be divided into three different sub-levels, which are commonly referred to as Virtual Data Center (vDC), vApp, and tier levels. Security rules and policy definitions can be set at any point in the hierarchy. These rules apply to all VMs that reside at or below the enforcement point (i.e., tenant level in Figure 2-8). Root-level policies and pools are systemwide and available to all organizations. In a multitenant system such as VMDC, to provide proper tenant separation and policy control, a unique instance of VSG must be deployed for each tenant.

*Figure 2-8 VSG Hierarchical Policy Model*



The VSG hierarchical policy classification is available to be leveraged for more complex policy rule sets, however, it is not mandatory to use all policy levels. For example, in the VMDC system reference model, though the VSG policy model allows for sub-tenancy, we commonly envision a tenant container as a single vDC with a requirement to support multiple categories of applications, each with multiple application tiers. Figure 2-9 illustrates this mapping, using the example of a specific application category (i.e., SharePoint). Implementers should follow a practical, “keep it simple” approach that meets their security policy profile requirements without unnecessary complexity.

*Figure 2-9 VSG Policy Profile Hierarchy Mapped to VMDC Tenancy*


Nexus 1000v
Port Profile: Tenant 1 SharePoint App
Security Profile: SharePoint Security Profile
Policy Set: SharePoint App Tiers
P1: SP Web Policy
P2: SP App Policy
P3: SP DB Policy
Root
Tenant “1”
SharePoint
Maps to VMDC Tenant Container
Zones map to VMDC Tenant Segments
P1
Web
P2
Application
P3
Database

VSG access controls can be applied to network traffic between packet source and destination based on Transmission Control Protocol (TCP)/User Datagram Protocol (UDP) ports, VM, or even custom attributes, making policy definition much more context-aware than simple legacy stateful packet filtering firewalls. In terms of application separation in the dynamic environment of a cloud-based infrastructure, a key benefit of the VSG is that by moving policy enforcement to the Nexus 1000V DVS, policy zones will automatically follow a VM as it moves from one hypervisor to another within the logical DVS boundary.

As of this writing, Nexus 1000V VSG Release 1.5 supports the following policy attributes for source/destination filtering:

- src.net.ip-address
- src.net.port
- dst.net.ip-address
- dst.net.port
- net.ip-address
- net.port net.protocol
- net.ethertype
- src.vm.name
- dst.vm.name
- vm.name
- src.vm.host-name
- dst.vm.host-name
- vm.host-name
- src.vm.os-fullname
- dst.vm.os-fullname
- vm.os-fullname
- src.vm.vapp-name
- dst.vm.vapp-name
- vm.vapp-name
- src.vm.cluster-name
- dst.vm.cluster-name
- vm.cluster.name
- src.vm.inventory-path
- dst.vm.inventory-path
- vm.inventory-path
- src.vm.portprofile-name
- dst.vm.portprofile-name
- vm.portprofile-name
- src.vm.custom.xxx
- dst.vm.custom.xxx
- vm.custom.xxx

## Perimeter Security

In traditional security models, it has long been a best practice to apply policy enforcement at defined boundaries between trusted and untrusted user communities or zones. A security zone comprises a logical construct of compute, network, and storage resources which share common policy attributes. One can leverage the common attributes within this construct to create security policies that apply to all the resources within that zone, however, in a highly virtualized system, it may be difficult to determine where these perimeters lie, particularly for the multitenant use case. In this system release, there are three perimeters essential for maintaining Enterprise-grade tenant security in a public or private cloud infrastructure:

1. **Front-end tenant perimeter—**This is the perimeter between less trusted zones and the interior of the tenant vDC within the cloud.
2. **(Intra-VDC) back-end tenant perimete—**This is the perimeter between a tenant's front-end servers and back-end servers.
3. **Back-end management perimeter—**This is the perimeter between the tenant "production" servers and back-end infrastructure management servers.

Between these perimeters, the following zones are defined:

1. **Public/Shared—**This zone provides a means of entry to the tenant vDC from a broader scope of external clients, sourced from either the public Internet, the Enterprise campus, or remote access VPNs (not shown in Figure 2-10). This is an untrusted or less trusted zone (i.e., versus those within the tenant vDC). Note that this zone would also potentially hold a general/shared infrastructure Demilitarized Zone (DMZ).
2. **Private—**The Private zone provides a means of entry to the tenant vDC via the cloud backbone, i.e., either the private WAN backbone or the public provider IP/NGN. In the latter case, the expectation is that clients will typically be utilizing a private L2 or L3 MPLS VPN across the public IP/NGN for access.
3. **Tenant DMZ—**This zone provides for a per-tenant DMZ (i.e., versus a more generalized DMZ elsewhere in the Enterprise or public provider infrastructure). It is understood that not all tenant vDCs will feature a DMZ Zone.
4. **Tenant front-end (web)—**This provides for a general front-end server zone, suitable for the placement of front-end application presentation servers.
5. **Tenant back-end—**Minimally, this would include two zones for app and database servers, but could be additional as required to accommodate multiple types of applications and additional application or policy-specific objectives.
6. **Back-end Management**—This zone contains the back-end servers that are used to manage the infrastructure.

   These could be virtual or bare-metal servers depending on the requirements of the management stack solution. The Storage Management network, which is the network used for administration of the cluster, nodes, and Storage Virtual Machines, would also be included in this back-end management network.
7. **Intercluster Network**—Optionally, you could also have the intercluster network or replication network, which is the network used for communication and replication between various Data ONTAP clusters. This network can be a dedicated network for replication separate from the data/management networks or this network can be shared. There are a number of configurations and requirements to consider when determining whether to share or dedicate networks/ports for replication. These include LAN type, available WAN bandwidth (compared to LAN bandwidth), replication interval, change rate, and number of ports used by the solution.

Figure 2-10 and Figure 2-11 illustrate how this model logically overlays onto the shared virtual and physical infrastructure.

*Figure 2-10 Tenant Perimeters and Zones*

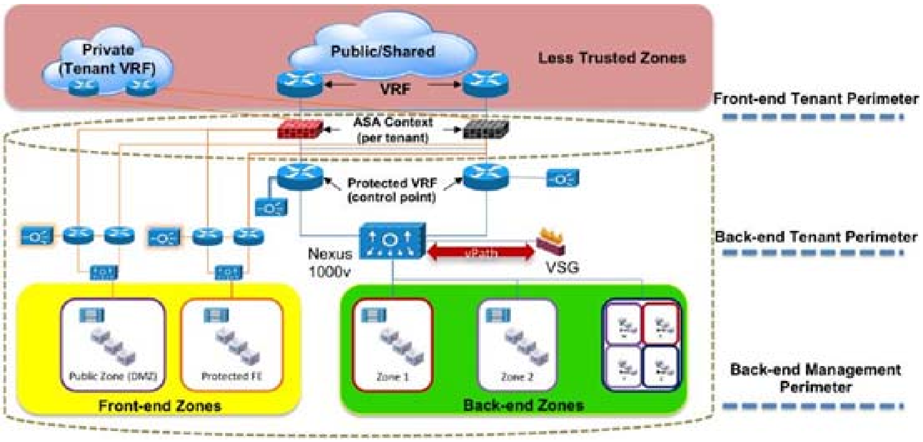
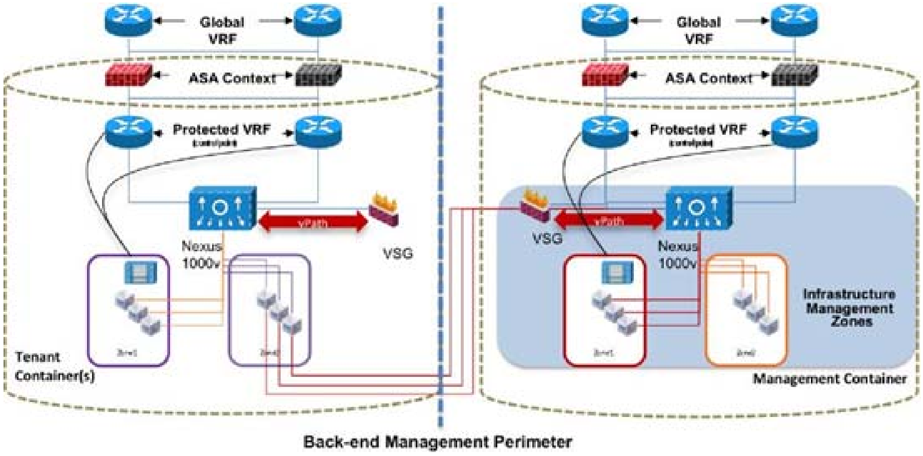


*Figure 2-11 Infrastructure Management Zones*



In Figure 2-11, a separate set of management vNICs allow tenant VMs to be “dual-homed,” with port profiles present on “production” and back-end infrastructure management Nexus 1000V instances. Multiple VSGs may be used in the management container to scale policy enforcement. This framework provides the flexibility to accommodate a variety of options including the following:

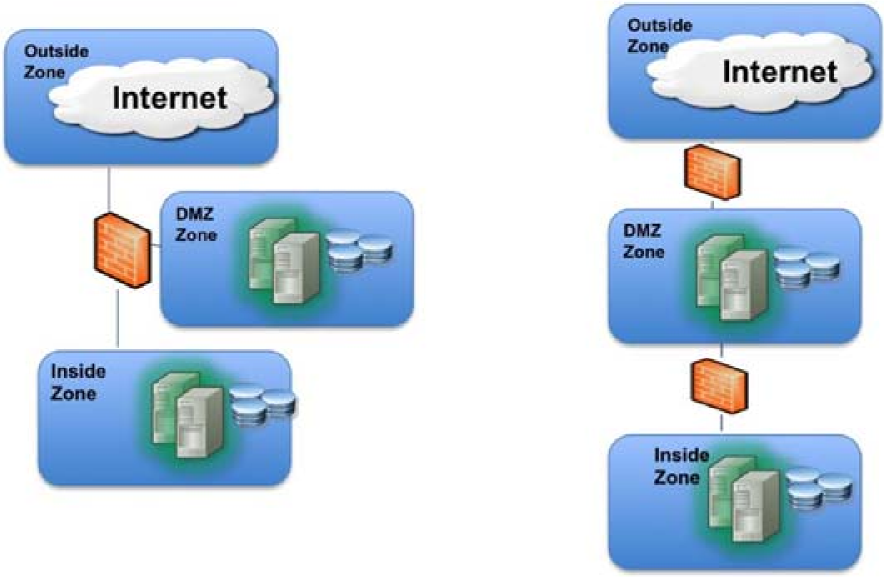
- A **provider (infrastructure) DMZ** (not shown in Figure 2-11).
- **Additional untrusted zones and nested zones.** Instead of a single shared public zone for remote VPN and Internet or campus access, the untrusted zones could be further segmented. Sample use cases applicable to the public provider context would be to provide separate zones for Independent Software Vendor (ISV) access or dedicated per-tenant public access zones.
- **Nested front or back-end zones.** For example, there could be two nested zones with different policy rule sets within a single front-end tenant zone, for DMZ servers and more general application presentation servers. Similarly, nested back-end zones could facilitate separation of “production” from “dev-test” back-end servers.

- **Accommodation of traditional security best practices.** For example, role-based infrastructure or server/VM access control (RBAC) tied to Lightweight Directory Access Protocol (LDAP) or radius directories. RBAC is not the focus of this system or release, however, it is a fundamental security requirement. A prerequisite is definition of role categories, to which differing access policies can be applied, i.e., tenant-user, tenant-administrator, administrator-user, and so on.

# DMZ Zones

A Demilitarized Zone (DMZ) is a small network inserted as a "neutral zone" between a private "inside" network and the outside public network. The DMZ's role is to prevent outside users from getting direct access to a server that has private data. Often, servers placed within the DMZ enhance perimeter firewall security by proxying requests from users within the private network for access to Web sites or other companies accessible on the public network. The proxy server then initiates sessions for these requests on the public network, however, it is not able to initiate a session back into the private network. It can only forward packets that have already been requested. How would a DMZ Zone be inserted into a tenant vDC in the cloud? Two basic models exist for placement of a DMZ Zone. As shown in Figure 2-12, in Model 1, the DMZ Zone is connected to the same network device as the Inside and Outside Zones. In Model 2, the DMZ is in a transit zone between a front-end and back-end firewall. Traditionally, Model 2 is considered to be slightly more secure, the logic being that two firewalls are better than one. This is a defense-in-depth measure, the premise being that if the front-end outside firewall is misconfigured, there is still a measure of security provided by the second firewall. It is this second placement option that the VMDC 2.3 release incorporates into the expanded vDC/VPDC tenancy model.

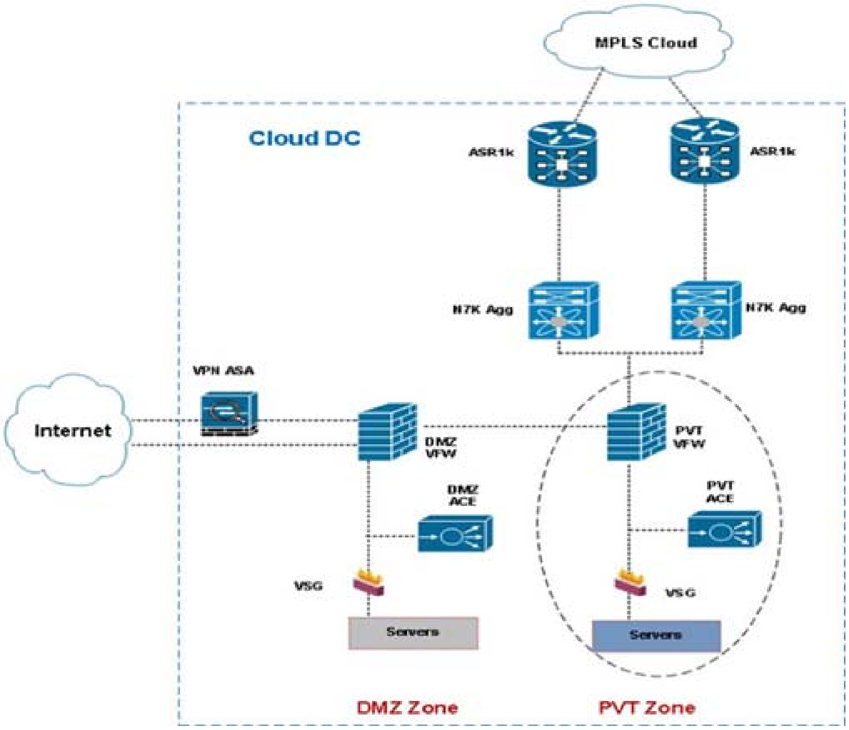
***Figure 2-12 DMZ Placement Options***



Note that though this system focuses on the application of a DMZ within the tenant vDC, typically there would also be a DMZ on the shared portion of the infrastructure.

Figure 2-13 illustrates the tiered security model and defense-in-depth aspect of the VMDC 2.3 design. There are two zones - DMZ and Private for the Expanded Gold container - each with its own front-end perimeter firewall (ASA) and back-end compute firewall (VSG).

*Figure 2-13 VMDC 2.3 Tiered Security with DMZ and Private Zones*



# VMDC 2.3 Containers

The VMDC 2.3 tenancy models (network containers), while aligned with those defined in earlier VMDC 2.2 phase, have been optimized to conserve less resources on the DC platforms and to achieve higher tenancy scale. The following are some of the optimizations done in the tenancy models to obtain higher tenancy scale:

- Define a new Copper container to meet tenants that require basic services and one VLAN and 4-5 VMs. This typically applies to SMB customers looking to place workloads in the public cloud. These Copper container cloud resources can only be accessed through the Internet. Using the global routing table for access from the Internet, and using a shared firewall for protecting these tenants, this container consumes fewer firewall contexts and VRF/ BGP resources on the ASR 1000 PE and Nexus 7004 aggregation nodes.
- VRF separation is a must for each tenant type.
- Eliminate the Core layer and the 6500 DSN.
- Optimize tenancy models to obtain higher scale, 500 tenants per pod, 2000 tenants per DC.
- As there is no Catalyst 6500 DSN used, the Silver and Bronze models have been simplified to consume less resources. There are no longer north and south VRF instances on the Nexus 7004 Aggregation layer for the Silver and Bronze containers. This conserves the VRF and VLAN resources on the Nexus 7004.
- To simplify the design and consume fewer Hot Standby Router Protocol (HSRP) and VLAN resources on the Nexus 7004, the ACE is utilized in one-arm mode, and moved into the ServerVLAN layer. Source NAT is utilized on the ACE, so that the VMs send return traffic back to the ACE.

- With this optimized ACE one-arm model in Silver container, the system can still do load balancing and non-load balancing flows for Silver and Gold containers. The system can still do load balancing for north-south or east-west traffic (across VLANs). VMs still use the Nexus 7000 as the default gateway.
- The Gold container has been tweaked to align with the new Silver container, so that a deployed Silver tenant in VMDC 2.3 can more easily be converted into a Gold tenant.

The following are some of the key aspects of the Services layer in the VMDC 2.3 system architecture:

- ASA and ACE appliances physically connecting to Nexus 7004 aggregation
- ASA logically sitting between the top and bottom VRF instances (for Gold)
- ACE sitting beneath the bottom VRF instance
- ASA 5585-X for firewall, as it supports 250 contexts
- ASA 5555 for remote access IPsec/SSL VPN, as the throughput requirement is low
- ACE 4710 for SLB
- ACE in one-arm mode, moved onto the same VLAN as the Server-VLANs

The figures below illustrate the different tenancy models defined for the VMDC 2.3 system.

*Figure 2-14 VMDC 2.3 Expanded Gold Container*

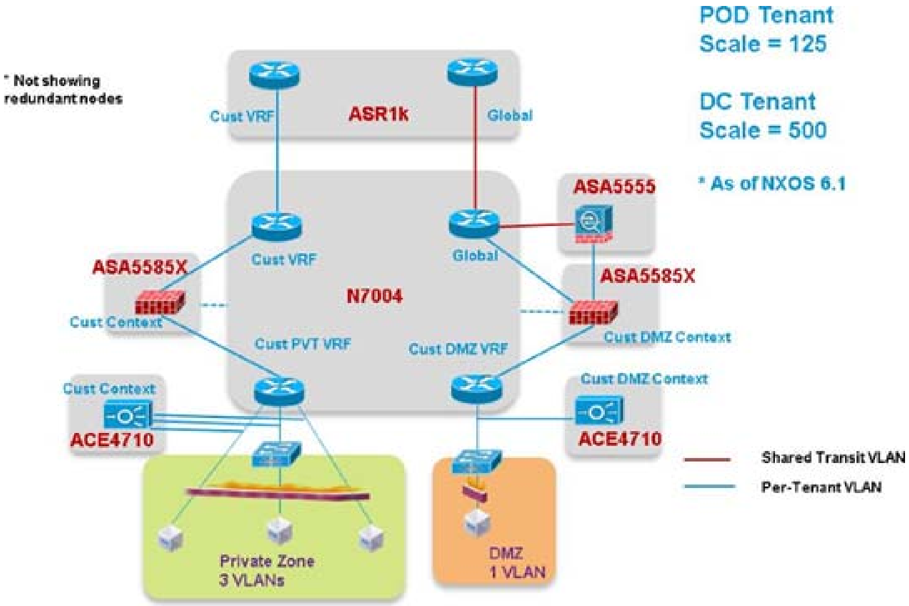


*Figure 2-15 VMDC 2.3 Gold Container*

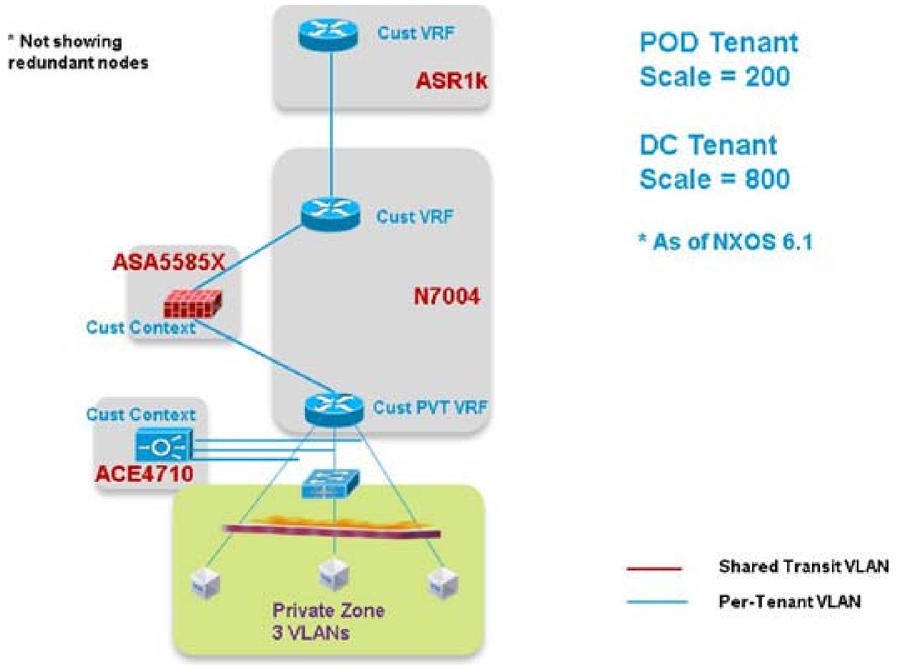


*Figure 2-16 VMDC 2.3 Silver Container*

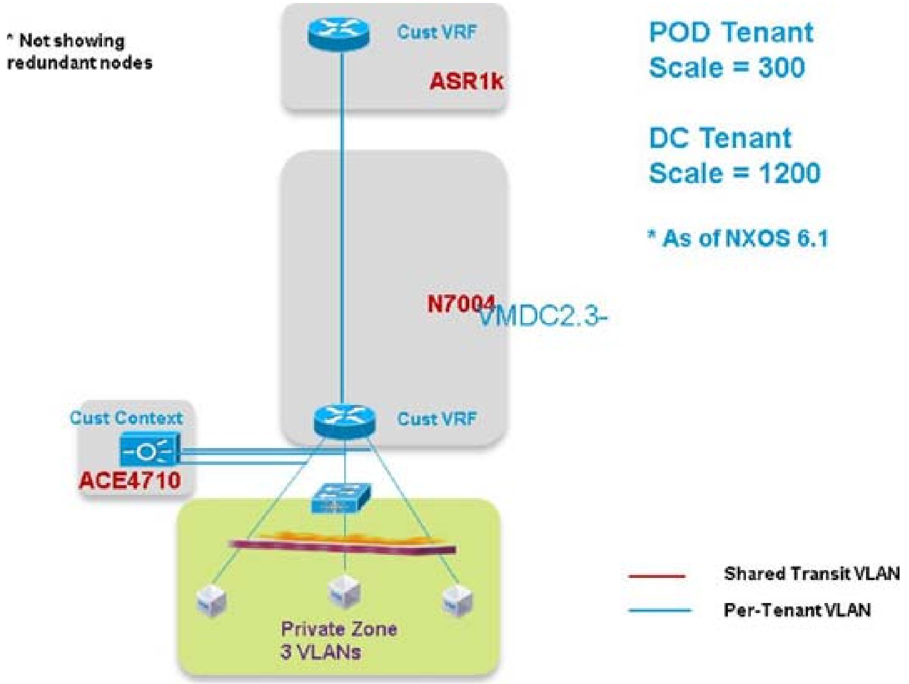


*Figure 2-17 VMDC 2.3 Bronze Container*

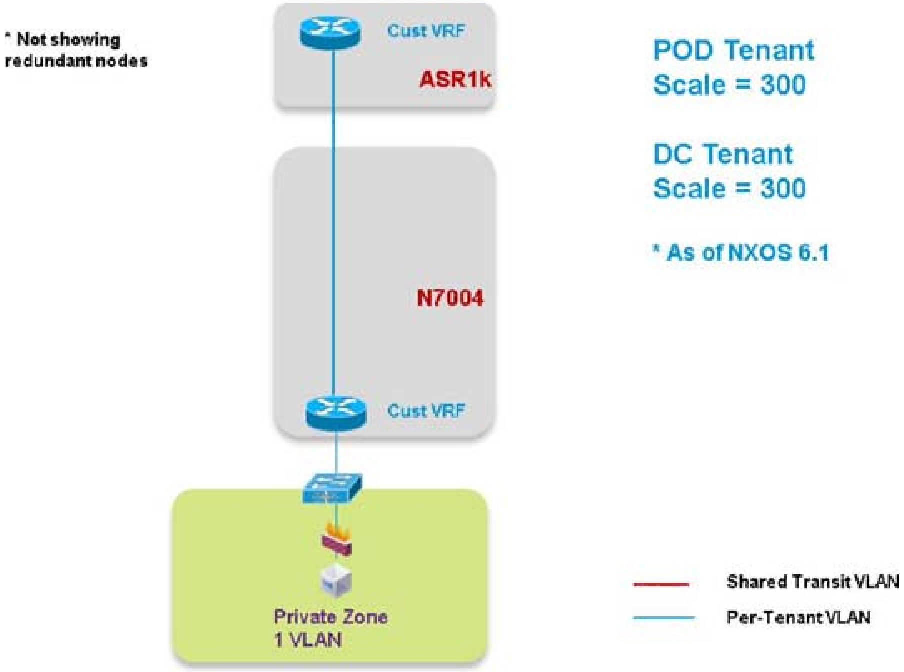
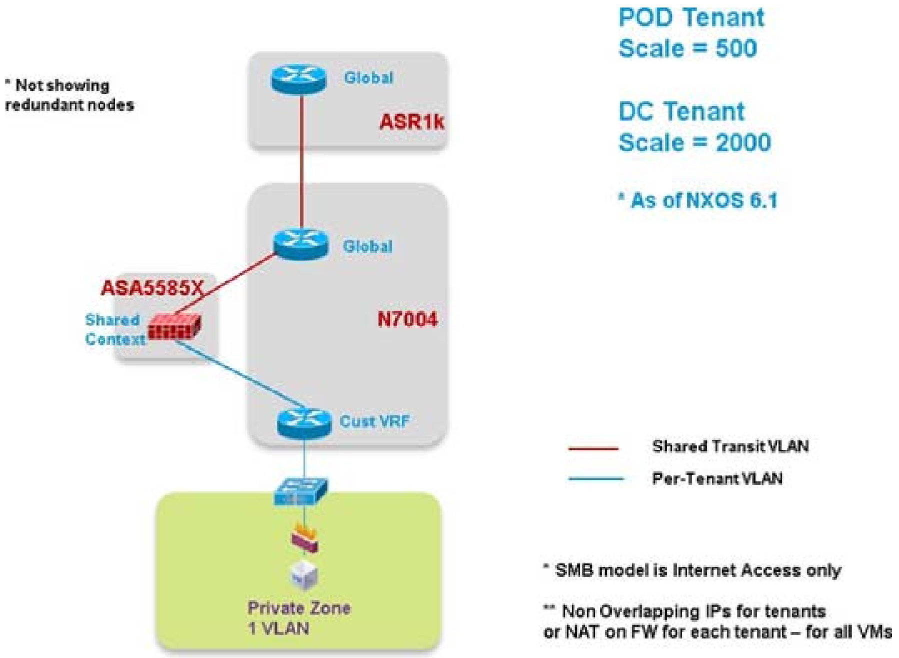


*Figure 2-18 VMDC 2.3 Copper Container*



# High Availability

A highly available infrastructure is the foundation for successful cloud-based services deployment and in particular, for service assurance or SLA guarantees. The VMDC reference architecture is thus modeled for the highest possibility infrastructure availability, to ensure no single point of failure, however, resiliency comes at incremental cost and complexity. The ongoing goal of this effort is to model and validate resiliency mechanisms in a multi-dimensional fashion, so that architects and implementers may make informed decisions about which solutions provide the optimal approach for their particular set of business service objectives and technical criteria.

This section presents the following topics:

- Redundant Network Design
- L2 Redundancy
- L3 Redundancy
- Compute Redundancy
- Storage Redundancy
- Services Redundancy

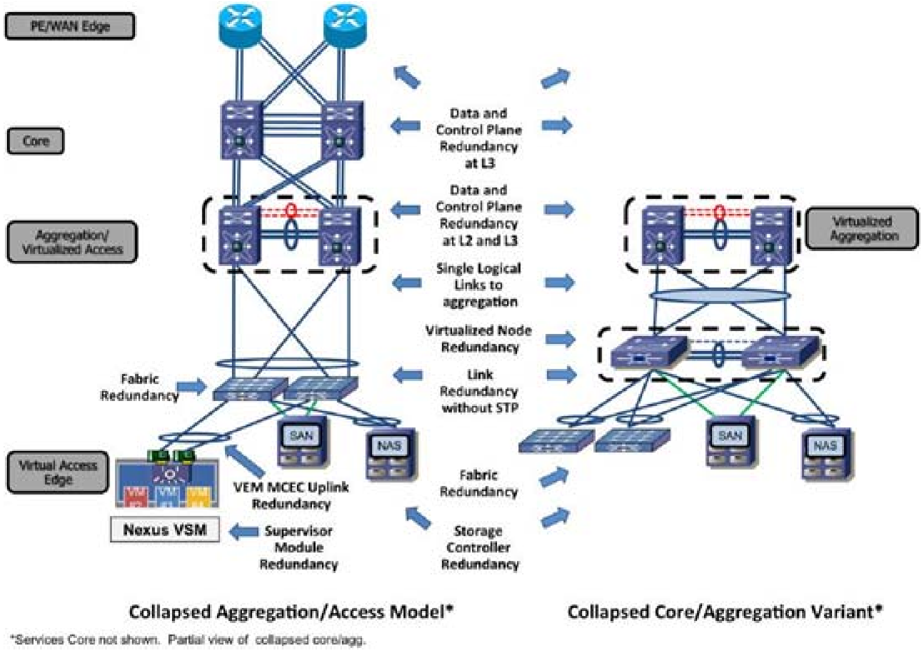
## Redundant Network Design

As discussed in VMDC 2.0 and VMDC 2.2, the reference architecture employs a multilayered approach to infrastructure HA design. Figure 2-19 illustrates how resilience mechanisms are utilized at every level of the infrastructure. These include the following:

- **Redundant Links, Nodes and Paths, End-to-End**
- **Core Layer—**Redundant L3 paths, links and nodes, and redundant supervisors.
- **Services Core (not shown)—**Redundant nodes, redundant data and control plane, redundant supervisors, and links and paths.
- **Aggregation Layer—**Redundant default gateway (Nexus 7000 aggregation nodes), redundant supervisors, and redundant links and L3 paths.
- **Access Layer—**Redundant nodes, supervisors, and links.
- **Compute Layer—**UCS - redundant fabric and control plane and intra-cluster HA.
- **Virtual Access—**Redundant forwarding path (CNA).
- **Storage—** Redundant SAN and NAS switching systems (not shown), redundant controllers, and RAID. This is an always on architecture with each component installed in pairs for redundancy in components as well as connections.

  In a FlexPod environment, the Cisco UCS fabric interconnects and NetApp FAS controllers benefit from the Cisco Nexus vPC abstraction, gaining link and device resiliency as well as full utilization (aggregation of bandwidth) of a non-blocking Ethernet fabric.
- **Management Servers (not shown)—**Intra-cluster HA, clustering or mirroring between management servers, vCenter Server heartbeats, and snapshots and cloning.

*Figure 2-19 Tiered HA Models*


PE/WAN Edge
Core
Aggregation/ Virtualized Access
Virtual Access Edge
Nexus VSM
Fabric Redundancy
Data and Control Plane Redundancy at L3
Data and Control Plane Redundancy at L2 and L3
Single Logical Links to aggregation
Virtualized Node Redundancy
Link Redundancy without STP
Fabric Redundancy
Storage Controller Redundancy
VEM MCEC Uplink Redundancy
Supervisor Module Redundancy
SAN
NAS
VM
Virtualized Aggregation
SAN
NAS
Collapsed Aggregation/Access Model*
Collapsed Core/Aggregation Variant*
*Services Core not shown. Partial view of collapsed core/agg.

# L2 Redundancy

The VMDC reference architecture utilizes several key L2 redundancy mechanisms at various points in the infrastructure to provide optimal multipathing. These are Virtual Port-channels (vPCs), Multi-Chassis EtherChannel (MEC), and MAC-pinning.

## Virtual Port-channels

A Cisco innovation based on port-channel technology (IEEE 802.3ad), vPCs allow multiple links to be used between a portchannel-attached device and a pair of participating switches. The two switches act as vPC peer endpoints and look like a single logical entity to the device. Traffic is forwarded and load balanced across all the links, but because they are bundled as one logical path, there is no loop created, and there is no requirement for Spanning Tree loop avoidance. With multiple active links comprising the path, vPCs typically provide faster link-failure recovery versus Spanning Tree Protocol (STP) processes, which involve relearning the L2 topology. Combining the benefits of load balancing with hardware node redundancy and port-channel loop management, vPCs offer optimal link bandwidth utilization. For these reasons, vPCs are recommended and leveraged whenever possible within the reference architecture. Specifically, in this release as in previous iterations, vPCs are deployed below the L3/L2 boundary, between the Nexus 7000 Aggregation layer and the Nexus 5000 access nodes or UCS 6100 Fabric I/O modules. Once again, as in previous releases, it is recommended that STP be enabled over the L2 portion of the infrastructure (i.e., below the Aggregation layer) for loop avoidance in the event of misconfiguration.

### Multi-Chassis EtherChannel

Another Cisco innovation based on port-channel technology, Multi-Chassis EtherChannel (MEC) is a port-channel that spans the two chassis of a switch. In this case, the DSN in the services core of the infrastructure. The portchannel-attached device views the MEC as a standard port-channel. Similar to vPCs, the MEC allows for optimal link bandwidth utilization across multiple links and redundant hardware nodes. MEC provides resilient routed paths between the Nexus 7000 nodes in the Aggregation layer of the infrastructure and the DSN in the service Core layer.

### MAC-Pinning

VMNICs may be pinned statically or dynamically to uplink paths within the UCS. In the reference architecture, MAC-pinning is used in conjunction with the Nexus 1000V to provide more granular load balancing and redundancy across the system. MAC-pinning does this through the use of notification packets, which in the event of a link failure, inform upstream switches of the new path required to reach destination VMs. These notifications are sent to the UCS 6100 Series Fabric Interconnect, which updates its MAC address tables and sends gratuitous Address Resolution Protocol (ARP) messages on the uplink ports so that the DC Access layer network can learn the new path.

## L3 Redundancy

### HSRP

Hot Standby Router Protocol (HSRP) is a first hop redundancy protocol, enabling the creation of redundant default gateways. HSRP allows two or more routers to act as a single “virtual” router, sharing an IP address and a MAC (L2) address. The members of the virtual router group continually exchange status messages, allowing one router to assume the routing responsibility of another, should it go out of commission for either planned or unplanned reasons. Failover to a standby router in the virtual router group will be transparent to hosts, as they will continue to forward IP packets to the same IP and MAC address. HSRP has been enhanced to gracefully interoperate with vPCs in a quasi “active/ active” state, such that a packet forwarded to the virtual router MAC address is accepted as local by the active and standby HSRP peers, however, responses will only be sent from the active HSRP peer. In order to provide default gateway redundancy, HSRP is deployed on the Nexus 7000 nodes within the Aggregation layer of the infrastructure, i.e., for all VLANs having their L3 termination on the SVI interfaces of the Nexus 7000 aggregation switches.

### BGP

An L3 IP routing protocol is required in the Aggregation and Edge layers of the VMDC model. Through various releases, the VMDC solution has been validated with both Open Shortest Path First (OSPF) and BGP protocols. In this release, BGP is used end-to-end within the DC. BGP is used to establish and maintain IP connectivity within the L3 portions of the infrastructure. In this scenario, External Border Gateway Protocol (eBGP) advertises routes between each defined autonomous system (WAN and Aggregation layers), rerouting over redundant L3 paths in the event of a node or link path failure. The use of loopback interface addressing is common in Interior Gateway Protocol (IGP), including Internal Border Gateway Protocol (iBGP), and for OSPF, ensuring that TCP sessions for routed paths are maintained in the event of link failures, while traffic is restored across active links. Loopback interfaces do not apply for eBGP scenarios, where peer interfaces are directly connected, however, in the event that peering over interfaces that are not directly connected is required, they can be utilized with additional configuration. More common for this scenario is the use of eBGP multi-hop, which must be used in any case in conjunction with an IGP or static route when the external peering interfaces are not directly connected.

By default, BGP selects one best path if there are several external equal-cost paths available from an AS. In the VMDC 2.3 solution, this would result in utilization of only half of the available infrastructure bandwidth during normal conditions. In order to get the most out of the available bandwidth, traffic is load balanced along the redundant paths. For parallel paths between two eBGP peers, loopback interfaces may be used in conjunction with eBGP multi-hop (and an IGP or static routes to communicate eBGP peer reachability) to load balance traffic. In the case of the VMDC solution, community strings are used to identify and load balance traffic across redundant eBGP paths between the Edge and Aggregation DC routers.

Additional optimizations for L3 resiliency leveraged in the system includes Cisco Nonstop Forwarding (NSF), Nonstop Routing (NSR), LDP sync, and MPLS graceful restart. More generally, tuning for fast L3 convergence may include the use of BGP graceful restart, BFD, tuning of hello and hold timers, and route summarization.

## Compute Redundancy

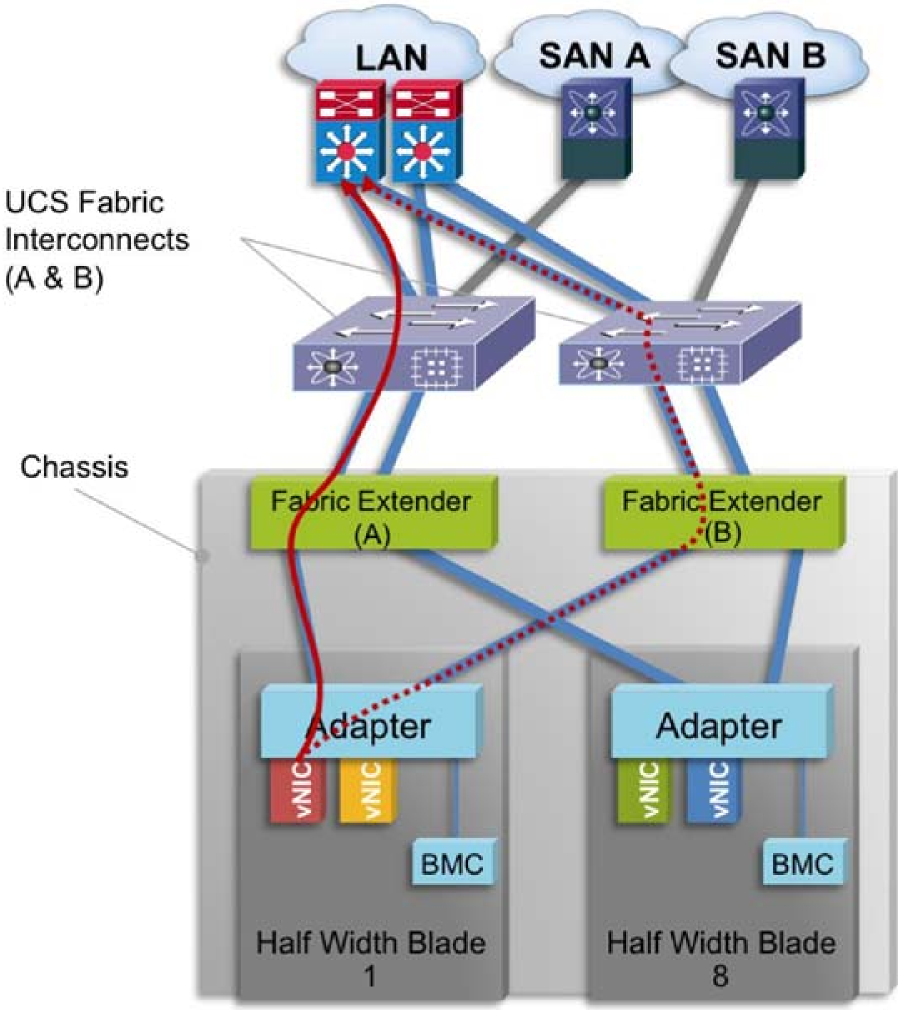
To enable redundancy within the Compute layer of the infrastructure, the following features are leveraged and recommended:

- UCS End-host (EH) mode
- Nexus 1000V and MAC-pinning (i.e., as previously discussed)
- Redundant VSMs and VSGs in active/standby mode
- VMware HA intra-cluster

### UCS End-host Mode

The UCS features a highly redundant architecture with redundant power, fabrics (i.e., data plane), control plane and I/O (see Figure 2-20).

*Figure 2-20 UCS*



At this Compute layer of the infrastructure, VNNICs are pinned to UCS fabric uplinks dynamically or statically. These uplinks connect to the Access layer switching systems, providing redundancy towards the network. In the VMDC solution, UCS Fabric Interconnect uplinks operate in EH mode. In this mode, the uplinks appear as server ports to the rest of the fabric. When this feature is enabled, STP is disabled, and switching between uplinks is not permitted. This mode is the default and recommended configuration if the upstream device is L2 switching. Key benefits with EH mode are as follows:

- All uplinks are used.
- Uplinks can be connected to multiple upstream switches.
- Spanning Tree is not required.
- There is higher scalability due to the control plane not being occupied.
- There is no MAC learning on the uplinks.

## Nexus 1000V and MAC-pinning

The UCS load balances traffic for a given host interface on one of the two redundant internal fabrics. By default, if a fabric fails, traffic automatically fails over to the available fabric, however, the UCS only supports port-ID and source MAC address-based load-balancing mechanisms. As previously discussed, the Nexus 1000V uses the MAC-pinning feature to provide more granular load-balancing methods and redundancy. VMNICs can be pinned to an uplink path using port profiles definitions. Using port profiles, the administrator can define the preferred uplink path to use. If these uplinks fail, then another uplink is dynamically chosen.

## Active/Standby Redundancy

For high availability, the Nexus 1000V Series VSM must be deployed in pairs, where one VSM is defined as the primary module and the other as the secondary. The two VSMs run as an active/standby pair, similar to supervisors in a physical chassis to provide high availability switch management. The Nexus 1000V Series VSM is not in the data path, so even if both VSMs are powered down, the Virtual Ethernet Module (VEM) is not affected and continues to forward traffic.

VSG redundancy is configured similarly to VSM redundancy, that is, like redundant VSMs, redundant VSGs must be installed on two separate physical hosts. One will be defined as the primary VSG and one as a secondary VSG, operating in active/standby HA mode. As in the VSM case, DRS, VMware HA, and VMware FT should be disabled for the redundant VSG VMs. The anti-affinity feature of VMware ESXi can be used to help keep the VSMs on different servers.

## Intra-Cluster High Availability

The VMDC architecture prescribes the use of VMware HA for intra-cluster resiliency. In contrast to VMware FT, which provides a 1:1 failover between a primary and secondary VM within a cluster, VMware HA provides 1:N failover for VMs within a single cluster. In this model, an agent runs on each server and maintains a heartbeat exchange with designated primary servers within the cluster to indicate health. These primary hosts maintain state and initiate failovers. Upon server failure, the heartbeat is lost, and all VMs for that server are automatically restarted on other available servers in the cluster pool. A prerequisite for VMware HA is that all servers in the HA pool must share storage, and virtual files must be available to all hosts in the pool. All adapters in the pool must be in the same zone in the case of FC SANs.

VNMC redundancy is addressed through VMware's HA mechanism, assuming creation of an ESXi cluster in which the redundant VNMC VMs reside. More generally, this technology is applicable for VMs running back-end management applications.

## Additional Considerations

Though not the focus of this release, additional resilience best practices would include the use of application-level clustering, and periodic VM and host backup mechanisms, such as snapshots or cloning and periodic database backups. These are all particularly applicable in terms of ensuring HA for back-end management hosts and VMs.

To facilitate maintenance operations or business continuance inter-site, the creation of automated disaster recovery plans for groups of VMs using scripted tools or utilities such as VMware's Site Recovery Manager may be necessary. This topic is discussed in VMDC 2.0 and Data Center Interconnect systems documentation.

# Storage Redundancy

In the Storage layer, the HA design is consistent with the HA model implemented at other layers in the infrastructure, comprising physical redundancy and path redundancy. In a FlexPod environment, the FlexPod architecture does not have a single point of failure at any level, from the server through the network to the storage. The fabric is fully redundant and scalable, providing seamless traffic failover should any individual component fail at the physical or virtual layer and there exists no single point of failure from a device or traffic path perspective.

## Hardware and Node Redundancy

The VMDC architecture leverages best practice methodologies for storage HA, prescribing full hardware redundancy at each device in the I/O path from host to storage-whether SAN or NAS orunified storage. In terms of hardware redundancy, this begins at the server, with dual-port adapters per host. Redundant paths from the hosts feed into dual UCS Fabric Interconnects and dual Nexus5000 Ethernet/FC/FCoE switches, and then into redundant storage arrays with tiered, RAID protection.

In today's environment, businesses require 24/7 data availability; providing continuous data availability begins with architecting storage systems that facilitate non-disruptive operations (NDO).

In the context of a FlexPod environment, the core foundation of NDO is the HA pair controller configuration, which provides high-availability solutions during planned and unplanned downtime events. Non-disruptive operations have three main objectives:

- **Infrastructure or hardware resiliency (unplanned events)**—This is the base building block for the storage subsystem and helps prevent an unplanned outage when a hardware or software failure occurs. Infrastructure resiliency includes redundant FRU components, multipath HA controller configurations, RAID, and WAFL proprietary software enhancements that help with failures from a software perspective. For node hardware failures or software failures, HA failover allows the node in the HA pair to failover.
- **Hardware and software maintenance operations (planned events)—**This refers to the next level of NDO, where components of the storage subsystem can be maintained and/or upgraded without incurring any outage of data. For example, the replacement of any hardware component from a disk drive or shelf fan to a complete controller head, shelf, or system. Although data is immortal and potentially lives forever, hardware does not. Therefore, maintenance and replacement of hardware will happen one or more times over the lifetime of a dataset.
- **Hardware and software Lifecycle operations (planned events)**—The third level of NDO is around the operations that a customer would perform to optimize the storage environment to meet business SLAs, from both capacity and performance perspectives, in addition to maintaining the most cost-optimized solution.

## Link Redundancy

**SAN**—Multiple individual FC links from the Nexus 5000s are connected to each SAN fabric, and VSAN membership of each link is explicitly configured in the UCS. In the event of an FC (NP) port-link failure, affected hosts will relogin in a round-robin manner using available ports. FC port-channel support, when available, will provide active-active failover support in the event of a link failure. Multipathing software from VMware or the SAN storage vendor can optionally be used to optimize use of the available link bandwidth and enhance load balancing across multiple active host adapter ports and links. In a FlexPod environment, hosts that access data served by clustered Data ONTAP using a block protocol are expected to make use of the Asymmetrical Logical Unit Access (ALUA), which is a standard and formalized way of defining path prioritization, port status, and access characteristics for SCSI devices. This standard is designed to define the protocol on how multipath IO should be managed between hosts and storage devices.

**NAS**—In a FlexPod environment, all system and network links feature redundancy, providing end-to-end HA. Network connectivity failures are addressed through the redundant port, interface groups, and logical interface abstractions offered by thecClustered Data ONTAP system. The NetApp FAS controllers use redundant 10Gb converged adapters configured in a minimum twoport interface group (ifgrp). Each port of the "ifgrp" is connected to one of the upstream switches, allowing multiple active paths by utilizing the Nexus vPC feature, which facilitates network availability and bandwidth.

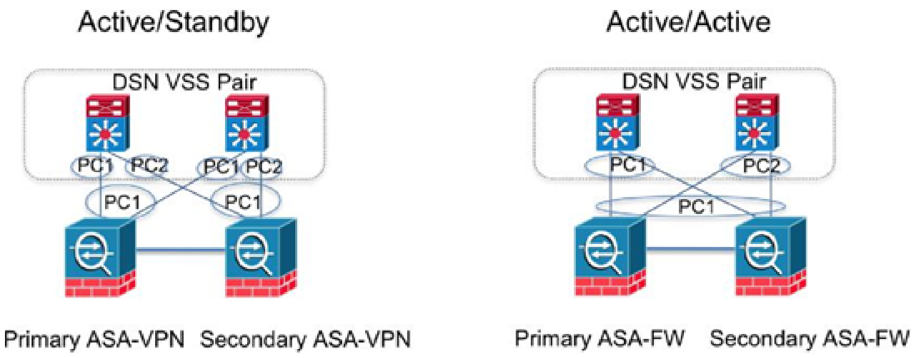
## Services Redundancy

As previously noted, in the Services layer of the infrastructure, redundancy is employed comprehensively to ensure no single point of failure. This includes physical (hardware, links) and logical (i.e., paths, control plane) redundancy.

### ASA

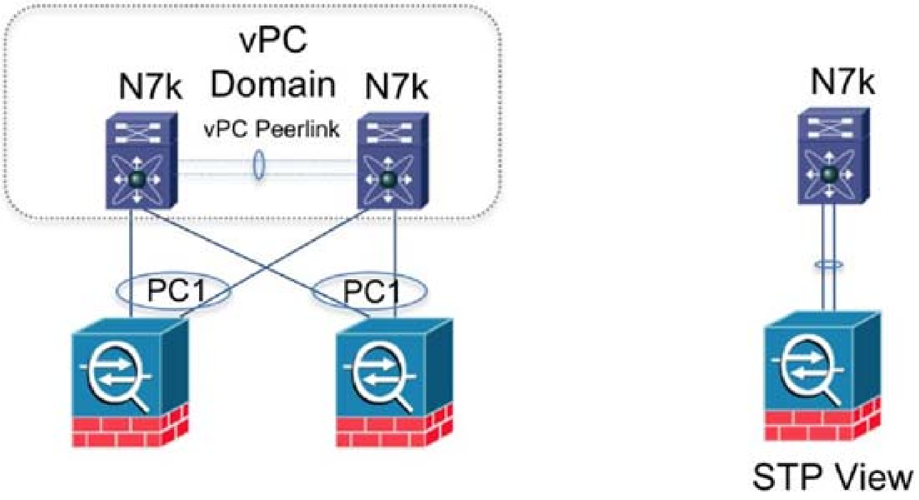
In this system release, two pairs of redundant ASA appliances are utilized for secure VPN remote access and for per-tenant perimeter firewalling. Release 8.4.1 for the ASA introduced support for several key HA features: 802.3ad EtherChannels and stateful failover with dynamic routing protocols, dramatically improving availability for the ASA in vPC or VSS enabled infrastructures. With this release, the ASA systems support configuration of up to 48 EtherChannels. Each channel group may consist of up to eight active interfaces. Two failover modes are supported, active/standby and active/ active. If redundant ASAs are configured in active/standby failover mode, two separate EtherChannels must be configured on each upstream switch in the VSS (i.e., 1 per ASA, as in Figure 2-22). In contrast, in active/active mode, only one EtherChannel is required per switch in the VSS pair. As of this writing, active/active failover is only supported when ASAs are in multi-context mode. Multi-context mode signifies that virtual contexts are configured on the ASA, dividing it into multiple logical firewalls, each supporting different interfaces and policies. Thus in this release, only the ASAs used for firewalling are configured for active/active failover (i.e., right in Figure 2-21). In this scenario, best practice recommendations include enabling interface monitoring and low poll time in failover configuration to get better resiliency and faster convergence of traffic traversing port-channels in the event of link failure.

*Figure 2-21 ASA Redundancy Modes*



This scenario works in a vPC environment as well, for redundant connectivity directly to Nexus 7000 aggregation nodes. In this scenario, the vPC allows creating an L2 port-channel between redundant Nexus 7000 Series devices and each redundant ASA. The concept is slightly different from VSS in that the two Nexus 7000 nodes are still independent switches, with different control and forwarding planes. This is the mechanism used for ASA redundancy in the VMDC 2.3 system.

*Figure 2-22 ASA Redundancy with Nexus 7000*



## ACE

Similar to the ASA, dual ACE appliances are connected to the Nexus 7004 Aggregation layer in vPC mode to provide redundancy.

# Service Assurance

Service assurance is generally defined as a set of service level management processes ensuring that a product or service meet specified performance objectives tailored to customer or client requirements. These processes involve controlling traffic flows, monitoring and managing key performance indicators to proactively diagnose problems, maintain service quality, and restore service in a timely fashion. The fundamental driver behind service assurance is to maximize customer satisfaction.

Though network service assurance covers a broad spectrum of metrics, including traffic engineering, performance monitoring, and end-to-end system availability, the VMDC 2.3 release focuses specifically on one particular component of service assurance that is key to providing differentiated service level agreements or Quality of Service (QoS).

In VMDC 2.3, the QoS framework is defined with the following objectives in mind:

- **Continued support for Network Control, Network Service, and Network Management traffic classes—**Including VMware vMotion, Service Console, and other infrastructure management flows, these are characterized as mission critical categories, essential to maintaining administrative operations during periods of network instability or high CPU utilization.
- **Continued support for three data service tiers (i.e., as in all previous VMDC systems releases)—**In terms of service level agreements, these are characterized by two metrics - differentiated bandwidth (i.e., B1, B2, and B3) and availability.

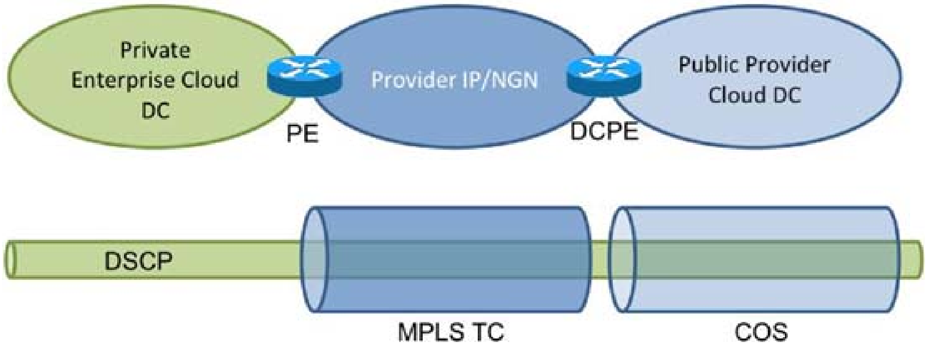
  - In private or public hosted cloud environments, these can be thought of as three utility compute service tiers (i.e., Gold, Silver, and Bronze/Copper).
  - In public hybrid inter-cloud environments, these can be part of a more elaborate set of end-to-end service tiers, with Gold and Silver classes correlating to business critical (in-contract, out-of-contract) service level agreements.
- **Support for multimedia, hosted collaboration traffic flows—**In terms of service level agreements, the low latency traffic classes in this new multimedia service tier (i.e., VoIP bearer and video conference) are characterized by three metrics, bandwidth, delay, and availability. The requisite traffic flows comprise:
  - New data bandwidth class for Cisco WebEx interactive collaboration
  - VoIP bearer traffic
  - VoIP call control
  - Video conferencing
  - Video streaming (future)
- **Support for admission control (future)—**QoS is a prerequisite for admission control, which may be applicable to future cloud bursting scenarios.
- **Support QoS across hybrid public/private domains**.
- **For the purpose of QoS, Copper tier tenants' traffic is classified and treated the same as Bronze tier tenant traffic**.

In the past, various VMDC system releases have followed either the traditional Cisco Enterprise/ Campus QoS model or the Cisco Service Provider IP/NGN QoS model, depending upon the use case scenarios and targeted audience. These differ slightly in terms of traffic classifications and markings, with the Service Provider model featuring slightly more complexity based on the need to support service level agreements end-to-end from public to private QoS domains (see Figure 2-23). In consideration of the objectives above, the QoS framework described in this release aligns with the IP/ NGN QoS model.

The hybrid prerequisite imposes an additional requirement that has traditionally been unique to the public provider case, but in the future as cloud service level agreements evolve, may apply to inter-cloud networking scenarios in a private-to-private cloud context. This is the need for QoS transparency. Described in RFC3270, QoS transparency allows a public provider to use their own marking scheme, prioritizing the Enterprise's priority traffic without remarking the Differentiated Services Code Point (DSCP) field of the IP packet. With this, the QoS marking delivered to the destination network corresponds to the marking received when the traffic entered the IP/NGN domain.

Any service level agreements that are applied would be committed across each domain, thus, public provider end-to-end service level agreements would be a concatenation of domain service level agreements IP/NGN + public provider DC. Within the public provider DC QoS domain, service level agreements must be committed from EC edge to edge: at the PE southbound (into the DC), in practice there would be an SLA per tenant, per class, aligning with the IP/NGN SLA, and at the Nexus 1000V northbound, there would be an SLA per VNIC, per VM (or optionally per class, per VNIC per VM). As this model requires per-tenant configuration at the DC edges only (i.e., PE and Nexus 1000V), ideally there is no *per-tenant* QoS requirement at the Core/Aggregation/Access layers of the infrastructure.

*Figure 2-23 Hybrid End-to-end QoS Domains*



The QoS framework defined in VMDC 2.3 follows the "hose" model for point-to-cloud services. This defines a Point-to-Multipoint (P2MP) resource provisioning model for VPN QoS, and is specified in terms of ingress committed rate and egress committed rate with edge conditioning. In this model, the focus is on the total amount of traffic that a node receives from the network (i.e., tenant aggregate) and the total amount of traffic it injects into the network. In terms of the VMDC architecture, the hose model is directly applicable to the edge QoS implementation at the public provider PE (i.e., the ASR 1000 DC PE in this release). Use case scenarios include P2MP VPLS-based transport services (i.e., hybrid DCI use cases), as well as more general VPDC services (i.e., where MPLS L2 or L3 VPNs provide inter-cloud transport).

In order to provide differentiated services, this release leverages the following QoS functionality:

- Traffic classification and marking
- Congestion management and avoidance (queuing, scheduling, and dropping)
- Traffic conditioning (shaping and policing)

## Traffic Classification and Marking

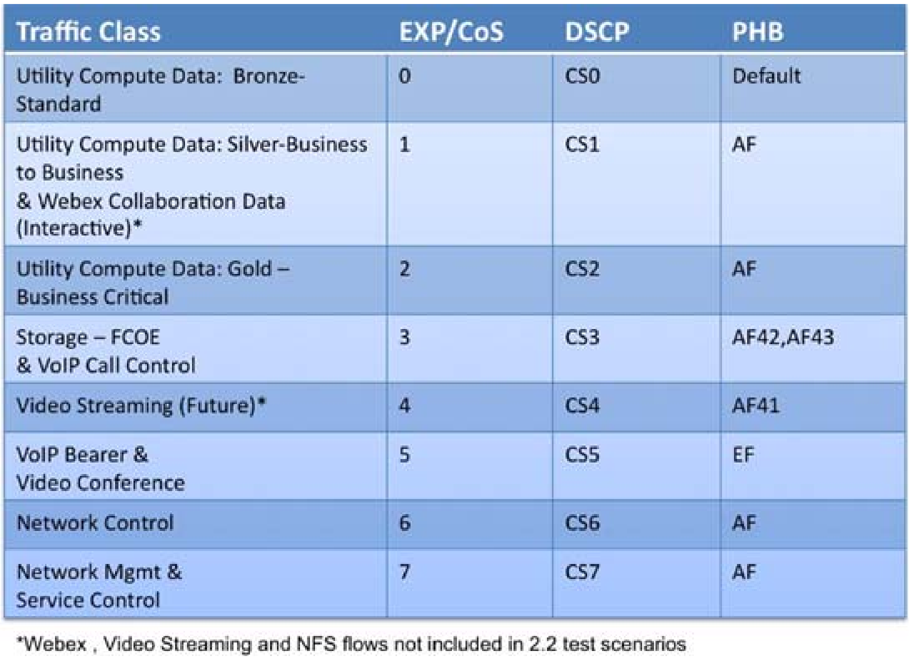
Classification and marking allow QoS-enabled networks to identify traffic types based on information in source packet headers (i.e., L2 802.1p CoS and DSCP information) and assign specific markings to those traffic types for appropriate treatment as the packets traverse nodes in the network. Marking Chapter 2 Design Details (coloring) is the process of setting the value of the DSCP, MPLS EXP, or Ethernet L2 Class of Service (CoS) fields so that traffic can easily be identified later, i.e., using simple classification techniques. Conditional marking is used to designate in-contract (i.e., "conform") or out-of-contract (i.e., "exceed") traffic.

As in previous releases, the traffic service objectives considered in VMDC 2.3 translate to support for three broad categories of traffic:

1. Infrastructure
2. Tenant service classes (three data; two multimedia priority)
3. Storage

Figure 2-24 illustrates a more granular breakdown of the requisite traffic classes characterized by their DSCP markings and Per-Hop Behavior (PHB) designations. This represents a normalized view across the VMDC and hosted collaboration validated reference architectures in the context of an eight-class IP/NGN aligned model.

*Figure 2-24 VMDC 2.3 Traffic Classes (Eight-Class Reference)*



| Traffic Class | EXP/CoS | DSCP | PHB |
|---|---|---|---|
| Utility Compute Data: Bronze-Standard | 0 | CS0 | Default |
| Utility Compute Data: Silver-Business to Business<br>& Webex Collaboration Data (Interactive)* | 1 | CS1 | AF |
| Utility Compute Data: Gold – Business Critical | 2 | CS2 | AF |
| Storage – FCOE<br>& VoIP Call Control | 3 | CS3 | AF42,AF43 |
| Video Streaming (Future)* | 4 | CS4 | AF41 |
| VoIP Bearer &<br>Video Conference | 5 | CS5 | EF |
| Network Control | 6 | CS6 | AF |
| Network Mgmt &<br>Service Control | 7 | CS7 | AF |

*Webex , Video Streaming and NFS flows not included in 2.2 test scenarios

It is a general best practice to mark traffic at the source-end system or as close to the traffic source as possible in order to simplify the network design, however, if the end system is not capable of marking or cannot be trusted, one may mark on ingress to the network. In the QoS framework defined in this release, the Provider DC represents a single QoS domain, with the Nexus 1000V forming the “southern” access edge, and the ASR 1000 forming the “northern” DC PE/WAN edge. These QoS domain edge devices will mark traffic, and these markings will be trusted at the nodes within the DC infrastructure. In other words, they will use simple classification based on the markings received from the edge devices.

## Queuing, Scheduling, and Dropping

In a router or switch, the packet scheduler applies policy to decide which packet to dequeue and send next, and when to do it. Schedulers service queues in different orders. The following are the most frequently used:

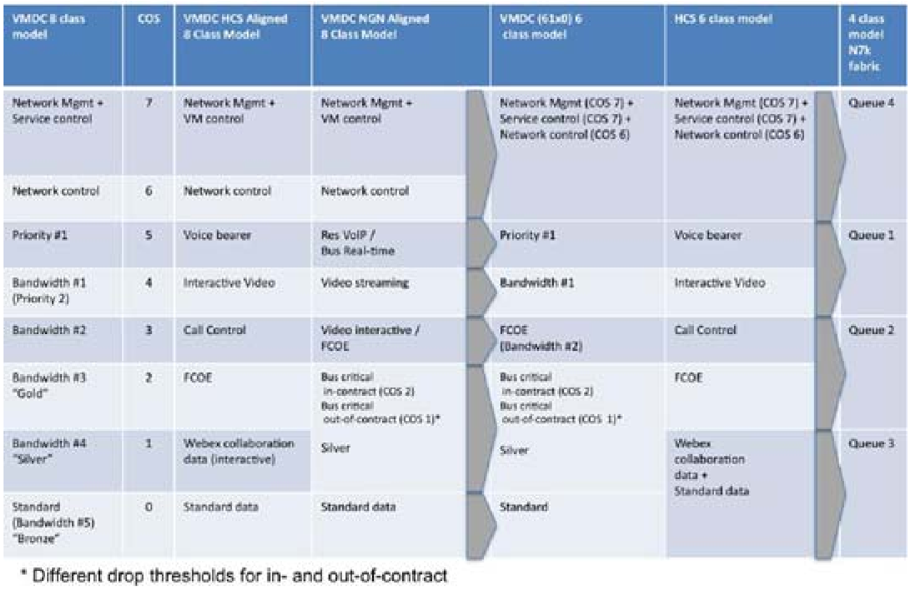
- First in, First Out (FIFO)
- Priority scheduling (aka priority queuing)
- Weighted bandwidth

In this release, a variant of weighted bandwidth queuing called Class-based Weighted Fair Queuing/ Low Latency Queuing (CBWFQ/LLQ) is used on the Nexus 1000V at the southern edge of the DC QoS domain. At the ASR 1000 northern DC WAN edge, Priority Queuing(PQ)/CBWFQ is used to bound delay and jitter for priority traffic while allowing for weighted bandwidth allocation to the remaining types of data traffic classes.

Queuing mechanisms manage the front of a queue, while congestion avoidance mechanisms manage the tail end of a queue. Since queue depths are of limited length, dropping algorithms are used to avoid congestion by dropping packets as queue depths build. Two algorithms are commonly used, weighted tail drop (often for VoIP or video traffic) or Weighted Random Early Detection (WRED), typically for data traffic classes. In this release, WRED is used to drop out-of-contract data traffic (i.e., CoS value 1) before in-contract data traffic (i.e., Gold, CoS value 2), and for Bronze/Copper/Standard traffic (CoS value 0) in the event of congestion.

One of the challenges in defining an end-to-end QoS architecture is that not all nodes within a QoS domain have consistent implementations. Within the cloud DC QoS domain, we run the gamut from systems that support 16 queues per VEM (i.e., Nexus 1000V) to four internal fabric queues (i.e., Nexus 7000). This means that traffic classes must be merged together on systems that support less than eight queues. In the context of alignment with either the HCS reference model or the more standard NGN reference, Figure 2-25 illustrates the class to queue mapping that applies to the cloud DC QoS domain in the VMDC 2.2 reference architecture.

*Figure 2-25 VMDC Class to Queue Mapping*



| VMDC 8 class model | COS | VMDC HCS Aligned 8 Class Model | VMDC NGN Aligned 8 Class Model | VMDC (61x0) 6 class model | HCS 6 class model | 4 class model N7k fabric |
|---|---|---|---|---|---|---|
| Network Mgmt + Service control | 7 | Network Mgmt + VM control | Network Mgmt + VM control | Network Mgmt (COS 7) + Service control (COS 7) + Network control (COS 6) | Network Mgmt (COS 7) + Service control (COS 7) + Network control (COS 6) | Queue 4 |
| Network control | 6 | Network control | Network control | | | |
| Priority #1 | 5 | Voice bearer | Res VoIP / Bus Real-time | Priority #1 | Voice bearer | Queue 1 |
| Bandwidth #1 (Priority 2) | 4 | Interactive Video | Video streaming | Bandwidth #1 | Interactive Video | |
| Bandwidth #2 | 3 | Call Control | Video interactive / FCOE | FCOE (Bandwidth #2) | Call Control | Queue 2 |
| Bandwidth #3 "Gold" | 2 | FCOE | Bus critical in-contract (COS 2)<br>Bus critical out-of-contract (COS 1)* | Bus critical in-contract (COS 2)<br>Bus critical out-of-contract (COS 1)* | FCOE | |
| Bandwidth #4 "Silver" | 1 | Webex collaboration data (interactive) | Silver | Silver | Webex collaboration data + Standard data | Queue 3 |
| Standard (Bandwidth #5) "Bronze" | 0 | Standard data | Standard data | Standard | | |

* Different drop thresholds for in- and out-of-contract

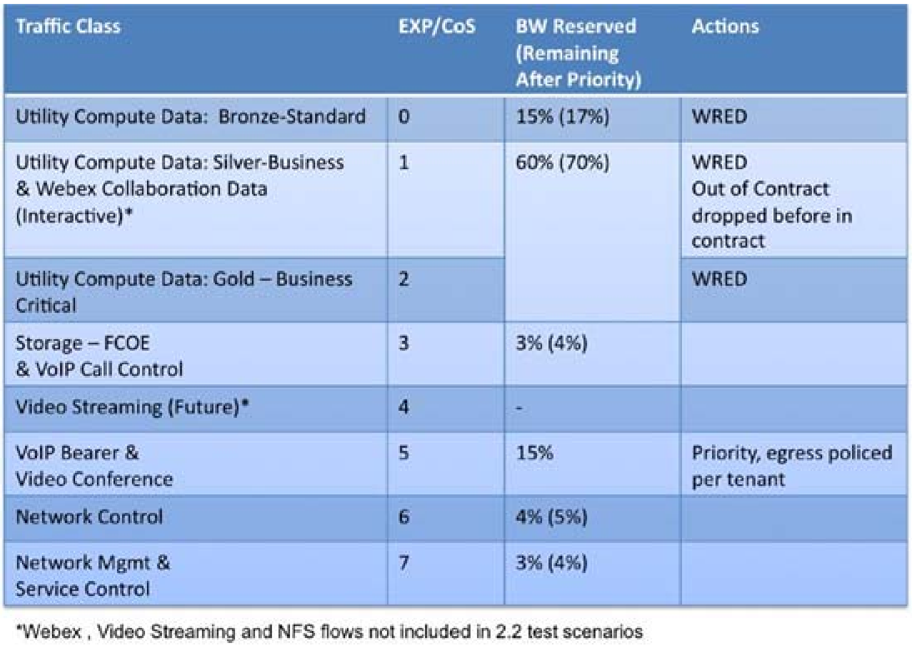
## Policing and Shaping

Policing and shaping are techniques used to enforce a maximum bandwidth rate on a traffic stream; while policing effectively does this by dropping out-of-contract traffic, shaping does this by delaying out-of-contract traffic.

In this release, policing is utilized within and at the edges of the cloud DC QoS domain to rate limit data and priority traffic classes. At the ASR 1000 DC PE, Hierarchical QoS (HQoS) is implemented on egress to the cloud DC. This uses a combination of shaping and policing in which L2 traffic is shaped at the aggregate (port) level per class, while policing is utilized to enforce per-tenant aggregates.

Sample bandwidth port reservation percentages used in validation to analyze QoS policy effects are shown in Figure 2-26.

*Figure 2-26 Sample Bandwidth Reservations (% of Port)*

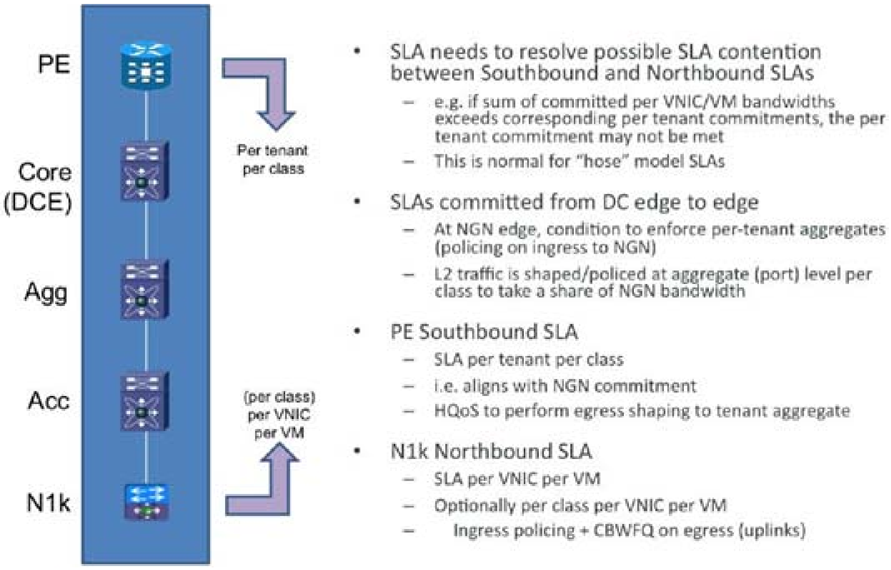


| Traffic Class | EXP/CoS | BW Reserved (Remaining After Priority) | Actions |
|---|---|---|---|
| Utility Compute Data: Bronze-Standard | 0 | 15% (17%) | WRED |
| Utility Compute Data: Silver-Business & Webex Collaboration Data (Interactive)* | 1 | 60% (70%) | WRED<br>Out of Contract dropped before in contract |
| Utility Compute Data: Gold – Business Critical | 2 | | WRED |
| Storage – FCOE & VoIP Call Control | 3 | 3% (4%) | |
| Video Streaming (Future)* | 4 | - | |
| VoIP Bearer & Video Conference | 5 | 15% | Priority, egress policed per tenant |
| Network Control | 6 | 4% (5%) | |
| Network Mgmt & Service Control | 7 | 3% (4%) | |

*Webex , Video Streaming and NFS flows not included in 2.2 test scenarios

Figure 2-27 provides a high-level synopsis of this end-to-end SLA framework.

*Figure 2-27 End-to-end SLA Framework*



- SLA needs to resolve possible SLA contention between Southbound and Northbound SLAs
  - e.g. if sum of committed per VNIC/VM bandwidths exceeds corresponding per tenant commitments, the per tenant commitment may not be met
  - This is normal for "hose" model SLAs
- SLAs committed from DC edge to edge
  - At NGN edge, condition to enforce per-tenant aggregates (policing on ingress to NGN)
  - L2 traffic is shaped/policed at aggregate (port) level per class to take a share of NGN bandwidth
- PE Southbound SLA
  - SLA per tenant per class
  - i.e. aligns with NGN commitment
  - HQoS to perform egress shaping to tenant aggregate
- N1k Northbound SLA
  - SLA per VNIC per VM
  - Optionally per class per VNIC per VM
  - Ingress policing + CBWFQ on egress (uplinks)

# Scalability Considerations

The ability to grow and scale the cloud infrastructure is a function of many factors, ranging from environmental, to physical and logical capacity. Considerations extend beyond the technical scope into the administrative domain.

## L2 Scale

Within the L2 domain, the following factors affect scale.

- **VM Density—**The number of VMs enabled on each server blade depends on the workload type and the CPU and memory requirements. Workload types demand different amounts of compute power and memory, e.g., desktop virtualization with applications such as web browser and office suite would require much less compute and memory resources compared to a server running a database instance or VoIP or video service. Similarly, Communications as a Service (CaaS), which provides raw compute and memory resources on-demand, agnostic to the applications running, is often characterized simply in terms of VMs per CPU core, with packaged bundles of memory options. The number of VMs per CPU core is a significant factor in another way, in that it in turn drives the number of network interfaces (virtual) required to provide access to VMs.
- **VMNICs per VM—**Each VM instance requires at minimum two vNICs. In most cases, several are utilized for connections to various types of Ethernet segments, and the ESX host itself will require network interfaces, i.e., for management control interfaces.
- **MAC Address Capacity—**The number of VMs and vNICs per VM will drive MAC table size requirements on switches within the L2 domain. Generally, these tables are implemented in hardware rather than software. So, unless a hardware upgrade is feasible, they will provide an upper bound to the scope of a single L2 domain. In the VMDC system reference architecture, the aggregate number of MAC addresses required within a pod is calculated based on the following formula: (# of server blades per pod) x (# of cores/blade) x (# of VMs/core = 1, 2, 4) x (# of MACs/VM = 4)
- **Cluster Scale—**Cluster sizes are constrained in a number of dimensions, i.e., in terms of number of servers, VMs, and logical storage I/O.
- ARP table size.
- **VLANS—**VLANs provide logical segmentation within the L2 domain, scaling VM connectivity, providing application tier separation and multitenant isolation. Every platform within the L2 and L3 portions of the infrastructure will have VLAN budgets, which must be considered when designing tenant containers.
- **Port Capacity—**At the Network layer, hardware port density is another physical budgetary constraint. Similarly, this consideration also applies to the Compute layer, in terms of logical Ethernet capacity on virtual access edge switches.
- **Logical Failure Domain—**An L2 domain is also a single, logical failure domain. From an administrative perspective, operational considerations come into play, in terms of how long it may take to recover from various types of failures if the affected set of resources is quite large.
- **L2 Control Plane—**When building L2 Access/Aggregation layers, the L2 control plane also must be designed to address the scale challenge. Placement of the spanning-tree root is key in determining the optimum path to link services, as well as providing a redundant path to address network failure conditions.

## L3 Scale

Scaling the L3 domain depends on the following factors:

- **BGP Peering—**Peering is implemented between the Edge, Core, and Aggregation layers. The Edge layer terminates the IP/MPLS VPNs and the Internet traffic in a VRF and applies SSL/ IPsec termination at this layer. The traffic is then fed to the Core layer via VRF-Lite. Depending on the number of data centers feeding the Edge layer, the BGP peering is accordingly distributed. Similarly, depending on the number of pods feeding a Core layer, the scale of BGP peering decreases as the layers are descended.
- **HRSP Interfaces—**Used to virtualize and provide a redundant L3 path between the Services, Core, Edge, and Aggregation layers.
- **VRF Instances—**VRF instances can be used to define a tenant network container. The scaling of VRF instances depends on the sizing of these network containers.
- **Routing Tables and Convergence—**Though individual tenant routing tables are expected to be small, scale of the VRF (tenants) introduces challenges to the convergence of the routing tables upon failure conditions within the DC.
- **Services—**Services consume IP address pools for NAT and load balancing of the servers. Services use contexts to provide tenant isolation.

## Resource Oversubscription

Increasing the efficiency of resource utilization is the key driver to oversubscription of hardware resources. This drives CAPEX savings up while still maintaining service level agreements.

### Network Oversubscription

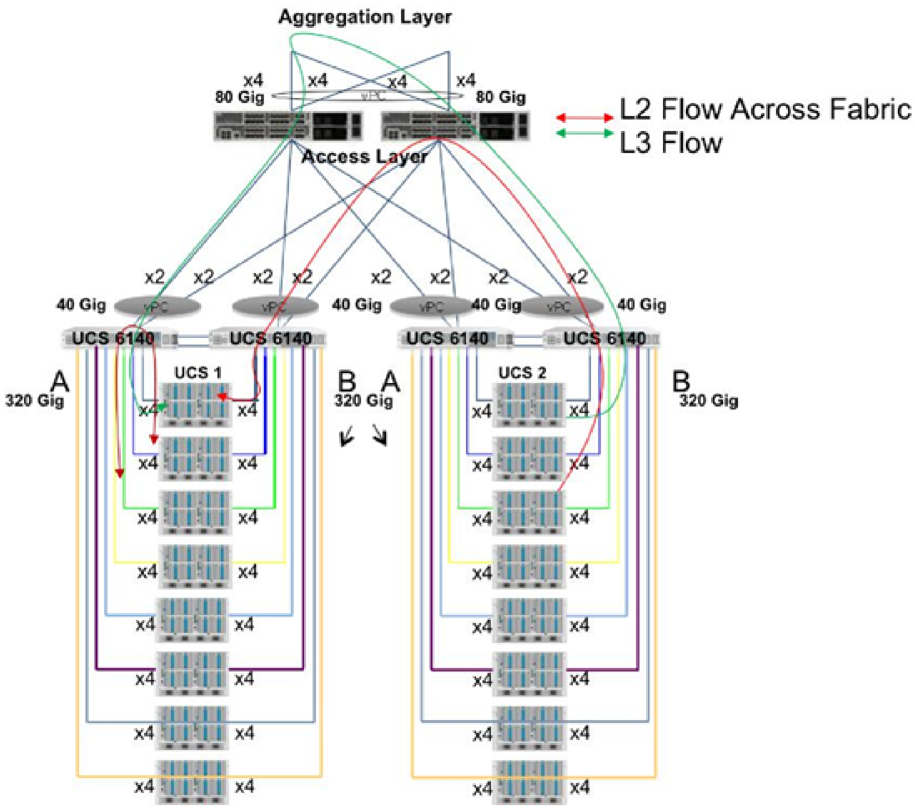
In considering what network oversubscription ratios will meet their performance requirements, network architects must consider likely traffic flows within the logical and physical topology. Multi-tier application flows create a portion of traffic that does not pass from the server farm to the Aggregation layer. Instead, it passes directly between servers. Application-specific considerations can affect the utilization of uplinks between switching layers. For example, if servers that belong to multiple tiers of an application are located on the same VLAN in the same UCS fabric, their traffic flows are local to the pair of UCS 6100/6200s (in the VMDC 2.3 design, UCS 6248UP Fabric Interconnect was utilized) and do not consume uplink bandwidth to the Aggregation layer.

Some traffic flow types and considerations are as follows:

- **Server-to-server L2 communications in the same UCS fabric—**Because the source and destinations reside within the UCS 6248 pair belonging to the same UCS fabric, traffic remains within the fabric. For such flows, 10 Gb of bandwidth is provisioned.
- **Server-to-server L2 communications between different UCS fabrics—**As depicted in Figure 2-28, the EH Ethernet mode should be used between the UCS 6248s (Fabric Interconnects) and Aggregation layer switches. This configuration ensures that the existence of multiple servers is transparent to the Aggregation layer. When the UCS 6248s are configured in EH mode, they maintain the forwarding information for all the virtual servers belonging to their fabric and perform local switching for flows occurring within their fabric, however, if the flows are destined to another pair of UCS 6248s, traffic is sent to the Access layer switches and eventually forwarded to the servers by the correct UCS 6248.

- **Server-to-server L3 communications**—Keeping multiple tiers of an application within the same UCS fabric is recommended if feasible, as it will provide predictable traffic patterns, however, if the two tiers are on the same UCS fabric but on different VLANs, routing is required between the application tiers. This routing results in traffic flows to and from the Aggregation layer to move between subnets.

*Figure 2-28 Traffic Flows Across the UCS System*



In practice, network oversubscription ratios commonly used a range from 4:1 to 8:1, depending on use case and level of infrastructure hierarchy. In this VMDC 2.X reference design, an 8:1 network oversubscription for inter-server traffic is considered for general compute deployment. This concept is illustrated in Figure 2-28, where the UCS chassis are connected to each UCS 6248 with 40 Gb (4x10 Gb) of bandwidth. When all eight chassis are connected, 320 Gb of bandwidth is aggregated at each UCS 6248. The four 10-Gb uplinks from each UCS 6248 form a port-channel where both vPC trunks are forwarding to the Access layer over 40 Gb of bandwidth. This configuration defines a ratio of 320 Gb /40 Gb, an oversubscription ratio of 8:1 at the Access layer when all links are active. Note, with the UCS 6200, you can use the FEX 2204XP (with 4 10G ports) or FEX 2208XP (with 8 10G ports), but for the VMDC 2.3 design, using 4 ports from each UCS FEX to each UCS Fabric Interconnect is sufficient for the bandwidth requirements.

In VMDC 2.3, an oversubscription ratio of 1:1 is provisioned at the Aggregation layer when all links are active, and using one ICS (Nexus 5548 Access switch pair) connecting to the Nexus 7004 Aggregation pair. When considering 3 ICS (3 Nexus 5548 Access pairs), the oversubscription ratio becomes 3:1. Oversubscription at the Aggregation layer depends on the amount of traffic expected to exit the pod. There will be flows where external clients access the servers. This traffic must traverse the Access layer switch to reach the UCS 6248.

The amount of traffic that passes between the client and server is constrained by WAN link bandwidth. In metro environments, Enterprises may provision between 10 and 20 Gb for WAN connectivity bandwidth, however, the longer the distance, the higher the cost of high bandwidth connectivity. Therefore, WAN link bandwidth is the limiting factor for end-to-end throughput.

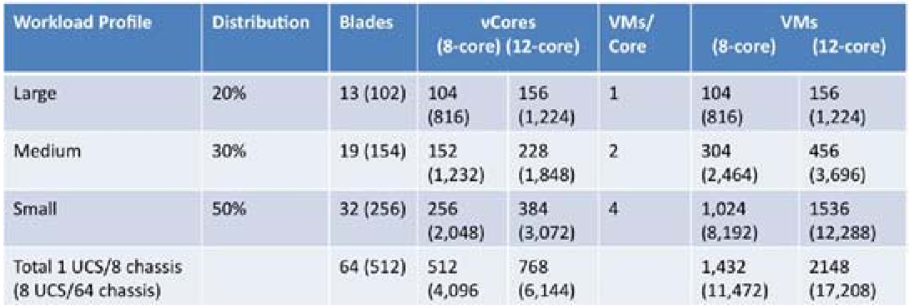
## Compute Oversubscription

Server virtualization involves allocating a portion of the processor and memory capacity per VM. Processor capacity is allocated as Virtual CPUs (vCPUs) by assigning a portion of the processor frequency. In general parlance, a vCPU is often equated to a blade core. In a very simple sense, compute oversubscription may be thought of as the ratio of vCores per VM per server or blade, and in terms of VMs per Gb of memory per blade. Of course, application workloads in real environments have distinct logical footprints of processing, memory, and storage requirements. For this reason, analysis of ICS stacks, which includes consideration of IOPS performance, is in fact conducted with specific applications generating traffic streams, however, for *infrastructure* modeling purposes, if IOPS performance is not a test criteria, it is useful to create profiles representing averages of varying workload sizes. In modeling the VMDC infrastructure, three workload profiles are leveraged with the following characteristics:

- Large (20%) - 1 vCore/VM (1:1)
- Medium (30%) - .5 vCore/VM (2:1)
- Small (50%) - .25 vCore/VM (4:1)

Older Cisco UCS B Series blade servers have two sockets, each supporting four to eight cores. B Series blade servers equipped with the Xeon 5570 processors support four cores per socket or eight total cores. The current generation of B series blade servers supports 12 cores (or more) per blade. In an eight-chassis system, this will equate to 64 blades x 12 cores or 768 cores per system. With workload distributions as above, this equates to 2,148 VMs per eight-chassis system, or 17,208 VMs per eight ICS with eight UCS chassis each (VMDC 2.2). In the VMDC 2.3 design, 3 such ICS system of eight UCS chassis are utilized (total of 192 UCS half-width blades), so this results in 6444 VMs. Figure 2-29 illustrates a sample workload distribution using 3 ICS of 8 UCS chassis each (8 blades per UCS chassis, 64 blades per ICS, 192 blades per VMDC 2.3 Pod)

***Figure 2-29 Sample Workload Profile Distributions***



| Workload Profile | Distribution | Blades | vCores<br>(8-core) | (12-core) | VMs/<br>Core | VMs<br>(8-core) | (12-core) |
|---|---|---|---|---|---|---|---|
| Large | 20% | 13 (102) | 104<br>(816) | 156<br>(1,224) | 1 | 104<br>(816) | 156<br>(1,224) |
| Medium | 30% | 19 (154) | 152<br>(1,232) | 228<br>(1,848) | 2 | 304<br>(2,464) | 456<br>(3,696) |
| Small | 50% | 32 (256) | 256<br>(2,048) | 384<br>(3,072) | 4 | 1,024<br>(8,192) | 1536<br>(12,288) |
| Total 1 UCS/8 chassis<br>(8 UCS/64 chassis) |  | 64 (512) | 512<br>(4,096 | 768<br>(6,144) |  | 1,432<br>(11,472) | 2148<br>(17,208) |

## Bandwidth per VM

As illustrated in Figure 2-28 and Figure 2-29, a 1:1, 1:2, and 1:4 Core:VM ratio for Large/Medium/Small workload types with a 20/30/50 distribution leads to an average of 22 VMs per blade (eight-core blades), 1,432 VMs per UCS, and 4,296 per VMDC 2.3 pod. In the case of twelve-core blades, this is 34 VMs per blade, 2,148 VMs per UCS and 6,444 VMs per VMDC 2.3 pod. The network bandwidth per VM can be derived as follows:

The UCS 6248 Fabric Interconnect in VMDC 2.3 design uses eight uplinks, so each UCS FI domain can support 80G/2148 = 37M per VM (twelve-core scenario); assuming all links are utilized and there is uniform load-balancing across links. Oversubscription prunes per VM bandwidth at each layer - Access, Aggregation and Edge. The Aggregation layer provides 3:1 oversubscription (assuming 3 ICS in a VMDC 2.3 Pod), hence 12.4M per VM at the Aggregation layer, assuming all North-South traffic.

## Storage Oversubscription

In a shared storage environment, thin provisioning is a method for optimizing utilization of available storage through oversubscription. It relies on on-demand allocation of blocks of data versus the traditional method of allocating all the blocks up front. This methodology eliminates almost all white space, which helps avoid poor utilization rates that may occur in the traditional storage allocation method where large pools of storage capacity are allocated to individual servers but remain unused (not written to). In this model, thinly provisioned pools of storage may be allocated to groups of vApps with homogenous workload profiles. Utilization will be monitored and managed on a pool-by-pool basis.

Storage bandwidth calculations for this system can be derived as follows:

There are 4x4G links from each UCS 6200 Fabric Interconnect to MDS SAN switch (aligning with a VCE Vblock 700). Assuming equal round-robin load-balancing from each ESX blade to each fabric, there is 32G of SAN bandwidth. Inside each UCS system, there is (160G/2) 80G FCoE mapped to 32G on the MDS fabrics. On the VMAX, eight FA ports are used for a total (both fabrics) of 32G bandwidth. EMC's numbers for IOPS are around 11,000 per FA port. Using eight ports, there are a total of 88,000 IOPS. Considering a UCS system, 88,000/1432 equates to 61 IOPS per VM. Extrapolating to a maximum 512 server pod, 88,000/11,472 provides just under 8 IOPS per VM (eightcore scenario) or approximately 5 IOPS per VM (twelve-core scenario). Of course, additional FC and Ethernet ports can be added to increase the per VM Ethernet and FC bandwidth.

In the context of a FlexPod environment, thin provisioning, data deduplication, and FlexClone thincloning technology are the critical components of the NetApp solution, offering multiple levels of storage efficiency across the virtual desktop OS data, installed applications, and user data. This helps customers save 50% to 90% of the cost associated with shared storage (based on existing customer deployments and NetApp solutions lab validation). Thin provisioning is a way of logically presenting more storage to hosts than is physically available. With thin provisioning, the storage administrator can access a pool of physical disks (known as an aggregate) to create logical volumes for different applications to use, while not pre-allocating space to those volumes. The space is allocated only when the host needs it. The unused aggregate space is available for the existing thin-provisioned volumes to expand or for use in the creation of new volumes. NetApp deduplication saves space on primary storage by removing redundant copies of blocks in a volume that is hosting hundreds of virtual desktops. This process is transparent to the application and user and can be enabled and disabled on the fly. Using NetApp deduplication and file FlexClone technology can reduce the overall storage footprint of virtual machines.

Some reference storage sizing considerations in a FlexPod environment (with NetApp FAS unified arrays) for this system are described below:

Selecting the proper system is more complicated than selecting a system that meets capacity requirements. Performance is a regular requirement and is often more complicated to plan for than capacity. Sizing is the process of obtaining or validating one or more system configurations that can provide the capacity and performance resources necessary to meet customer requirements.

Each FlexPod can easily scale to a VMDC "pod" and handle the storage requirements based on appropriate sizing guidelines. It is easily scalable when requirements and demands change. This includes vertical scaling (adding additional resources within the FlexPod), as well as horizontal scaling (adding additional FlexPod units). A given FlexPod unit can be scaled up, down or out based on performance and capacity requirements, physical limitations, best practices, data center power and cooling availability and so on.

### Vertical Scaling

Scaling a FlexPod unit vertically involves modifying or increasing components within the base FlexPod unit dependent upon specific customer requirements. Some examples of reasons to change the base FlexPod configuration include a need for:

- Modifications to the bandwidth per Cisco UCS chassis
- Increased bandwidth per Cisco UCS fabric interconnect
- Addition or modification of compute resources
- Specific storage performance requirements
- Addition or modification of network interfaces
- Addition or modification of storage I/O capabilities or the modification of storage capacity

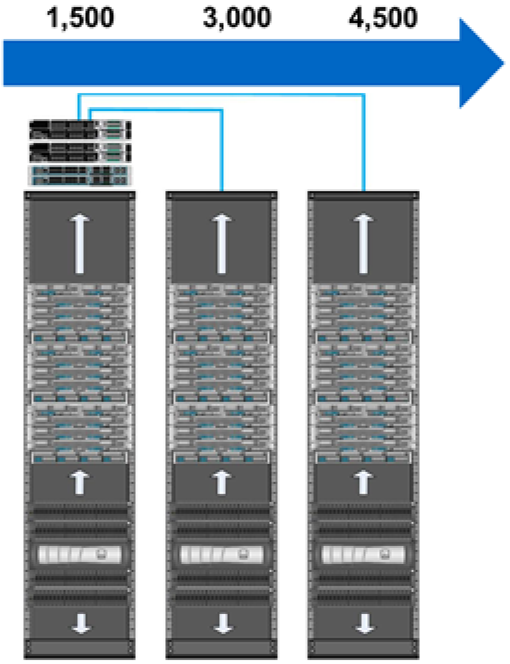
The benefit of the FlexPod architecture is that each of these elements can be modified independently, providing best practices are followed, and supportability of the architecture remains.

### Horizontal Scaling

Scaling horizontally within the FlexPod construct involves the addition of FlexPod units based on specific customer requirements. Some reasons to increase the number of FlexPod units include:

- Physical data center space limits
- Power limits
- Specific storage performance requirements
- Storage I/O considerations

*Figure 2-30 Scale Out with FlexPod*



Scale out with standard and proven configurations

- Predictable and highly efficient
  - Capacity and performance
  - Floor space, power, cooling
- Or scale up within a FlexPod

Benefits

- Reduce effort for architecture, design, deployment, and test
- Reduce infrastructure deployment cycle time by up to 50%
- Manage resource pools not individual systems
  - Storage, compute, network

Much as vertical scaling benefits the customer, the ability to deploy additional pre-configured base FlexPod units eases many of the decisions that must be made when constructing a shared infrastructure. Huge operational efficiencies result when choosing a standard deployment chunk for infrastructure and using that in a repeated manner to scale out to meet the needs of the business.

Further, the time to acquire and deploy resources can be drastically reduced when dealing with a standardized IT asset like FlexPod. This means a reduction in the effort to design, deploy, and test your expanding environment, and reduces the number of unique components to be managed.

### Characterized and Uncharacterized Workloads

Definitive sizing recommendations can be determined when the workloads are very well defined, well studied, and well understood. These are characterized workloads like Microsoft Exchange Server, Microsoft SQL Server, or Oracle Database, and so on. In general, because the workload is well known and well understood, precise sizing tools already exist for characterized workloads. Some of the parameters that have been well understood in characterized workloads might include the number of users, the level of concurrency, the working set size, the backup/DR requirements, and so on. These values are given to a sizing tool to translate into the specific hardware recommended for the workload. Additionally, you have to identify the sizing limitations of the compute environment. These would be the amount of memory per blade, the number of cores per blade, and the number of VMs allowed on each blade. Note that these numbers also depend on the application demands and the workloads running inside the VM, meaning not all VMs are created equal. The resource consumption of a VDI desktop is not the same as the resource consumption of a departmental SharePoint server. As the name implies, uncharacterized workloads vary widely and are neither well defined, nor well understood. Both public and private cloud general-purpose virtualized client and server workloads tend to fall into the uncharacterized category.

Application workloads in real environments have distinct logical footprints of processing, memory, and storage requirements. For this reason, analysis of ICS stacks, which includes consideration of IOPS performance, is in fact conducted with specific applications generating traffic streams.

However, for infrastructure modeling purposes, if IOPS performance is not a test criterion, it is useful to create profiles representing averages of varying workload sizes. Refer to FlexPod Solutions for design and sizing guidelines: http://www.netapp.com/us/system/pdf-reader.aspx?pdfuri=tcm:10-61208-16&m=tr-3884.pdf

To illustrate an example of storage sizing for a VMDC infrastructure, the workload profile distribution shown in Figure 2-29 is expanded to size for a clustered ONTAP deployment. The workloads are divided into small, medium, and large to define VM classes as described in Figure 2-29.

Assuming a small VM requires 10 IOPS of disk performance and has 4GB of RAM and 20GB of disk capacity. A medium VM requires 30 IOPS of disk performance and has 8GB of RAM and 50GB of disk capacity. A large VM requires 100 IOPS of disk performance and has 16GB of RAM and 100GB of disk capacity. This is shown in Table 2-2 below.

*Table 2-2 VM Sizing Characteristics*

| VM Size | Distribution | Compute (GHz) / (VM:vCore) | Memory (GB) | Storage (GB) | Storage Throughput (IOPS) |
|---|---|---|---|---|---|
| **Large** | 20% | 1(1:1) | 16 | 100 | 100 |
| **Medium** | 30% | 0.5 (2:1) | 8 | 50 | 30 |
| **Small** | 50% | 0.25 (4:1) | 4 | 20 | 10 |

For a 2000 VM configuration with the above VM specifications and distribution, the storage capacity required is shown in Table 2-3 below.

*Table 2-3 VM Sizing Characteristics*

| VM Size | Distribution | VMCount | Storage (GB) | (IOPS) |
|---|---|---|---|---|
| **Large** | 20% | 400 | 40,000 | 40,000 |
| **Medium** | 30% | 600 | 30,000 | 18,000 |
| **Small** | 50% | 1000 | 20,000 | 10,000 |
| **Total** | 100% | 2000 | 90,000 | 68,000 |

Sizing this workload using the NetApp sizing tool available at https://spm.netapp.com we see that a workload of 68k random IOPS can be hosted on a NetApp FAS62xx series running clustered Data ONTAP. The FlexPod architecture can be scaled out in additional chunks to accommodate additional user workloads as shown in Figure 2-30.

# DC Scalability

The DC scalability based on the large pod is determined by the following factors:

- **MAC address support on the Aggregation layer.** The Nexus 7000 platform supports up to 128,000 MAC addresses. For example, considering the modeled distribution mix of Small, Medium, and Large workloads, 11,472 workloads would theoretically be enabled in each large pod, which translates to 11,472 VMs (i.e., on eight-core blades) or 17,208 workloads and VMs on twelve-core B200 series blades. Different vNICs with unique MAC addresses are required for each VM data and management network, as well as NICs on the ESX host itself. The VMDC solution assumes four

MAC addresses per VM and this translates to 45,888 (or 68,832) MAC addresses per large pod. In order to optimize intra-pod scale, sharing VLANs between pods is generally discouraged unless it is required for specific purposes, such as application mobility. Filtering VLANs on trunk ports stops MAC address flood.

- **10 Gig port densities.** The total number of 10-Gig ports supported by the Access/Aggregation layer platform dictates how many additional pods can be added while still providing network oversubscription ratios that are acceptable for the deployed applications. For example, from a physical port density standpoint (based on the M1 series line cards), the Nexus 7018 could theoretically support up to six large pods, each equating to 512 blades.
- **Control plane scalability.** Control plane scalability will vary depending upon the type of encapsulation(s) used to identify tenants, L2 protocols in use (i.e., HSRP and STP), and upon route protocol selection. In the case where VRF-Lite is used, each tenant VRF deployed on the Aggregation layer device must maintain a routing adjacency for its neighboring routers. These routing adjacencies must maintain and exchange routing control traffic, such as hello packets and routing updates, which consume CPU cycles. As a result, control plane scalability is a key factor in determining the number of VRF instances (or tenants) that can be supported. This design has been characterized for 150 tenants. A DC based on a large pod design can provide a minimum of 256 tenants and a range of workloads from 8,192 and up, depending on workload type. It can be expanded further by adding additional large pods to the existing Core layer. In the future, application of LSP and Inter-AS at the core of the infrastructure will serve to further scale this model.

# VMDC 2.3 Scale

Some of the key scale factors for the VMDC 2.3 platforms and design are listed below.

- The ASR 1006 can have up to 12 10G interfaces.
- There are two ports on the ASR 1000 for upstream and two ports per downstream pod.
- VMDC 2.3 can build four pods when using the ASR 1006.
- Nexus 7000 with NX-OS 6.1 has control plane scale limits of 1000 VRF instances, 1000 HSRP, 1000 BGP peers, and 4000 VLANs (NX-OS 6.2 will increase these numbers by a large factor).
- Based on these limits, within one pod, VMDC 2.3 can do 125 Expanded Gold containers, 200 Gold containers, 300 Silver containers, 300 Bronze containers, or 500 Copper containers.
- Using a mixed-tenancy model, VMDC 2.3 can support up to 500 tenants in a pod - 10 Expanded Gold, 20 Silver, 220 Bronze, and 250 Copper containers.
- Using four pods, VMDC 2.3 can scale to 2000 mixed tenants in a VMDC 2.3 DC.
- The Nexus 7004 with F2 line cards can support up to 16,000 MAC addresses (this is a limitation on the F2 line card). Keeping aside 2000 MACs for switch, service appliances, etc., we can use up to 12,000 MACs for VMs. Assuming two vNIC per VM, VMDC 2.3 can support up to 6000 VMs per pod.
- This translates to 2000 VM per FlexPod or ICS (three ICS stacks per pod).

**Note** The above numbers are mostly derived from network-centric limits - number of MACs on Nexus 7000 F2 linecards, number of vEths on Nexus 1000V, number of vNIC per VM etc. For deployment sizing, the application workload and storage requirements need to be taken into consideration. The number of VMs deployed per FlexPod is determined by the application or workload requirements. The FlexPod architecture can be scaled out in chunks as necessary to accommodate additional user workloads. The number of VMs supported per FlexPod unit is determined by the application or workload requirements.

A FlexPod unit can be scaled up or scaled out to host all the VMs for a particular Pod depending on the workload (for example, 4000 or 6000 additional VMs depending on the workload), so carefully consider the workload when sizing the FlexPod solution. Use the following sizing tools for more details: FlexPod sizing tool, and NetApp Storage Performance Modeler sizing tool.

Based on the above factors, the scale-out model for the VMDC 2.3 system - from a pod scale, and from a DC scale perspective - is illustrated in Figure 2-31, Figure 2-32, and Figure 2-33.

***Figure 2-31 VMDC 2.3 Scaled Pod with ICS***

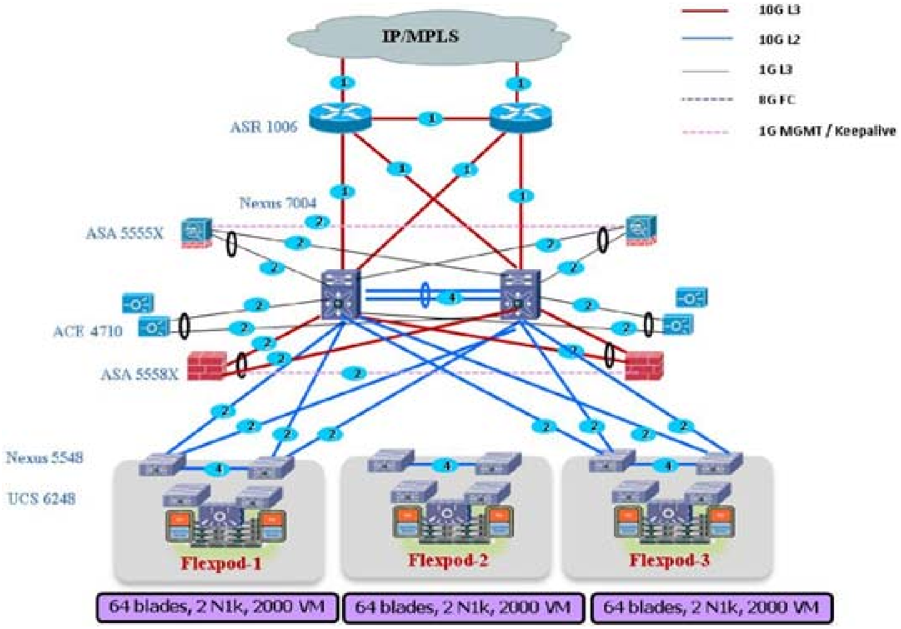


*Figure 2-32 VMDC 2.3 Scaled Pod with FlexPod*

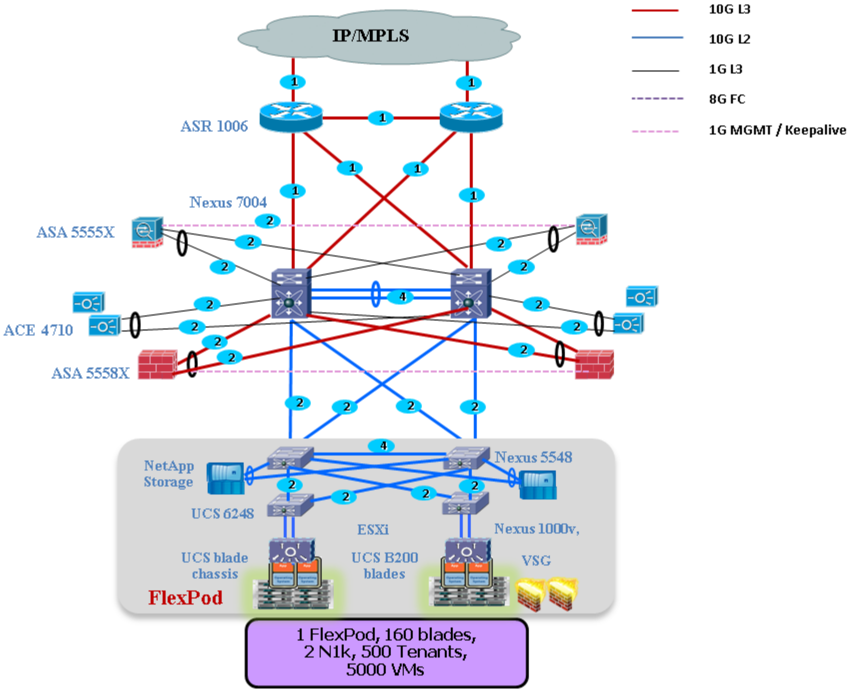
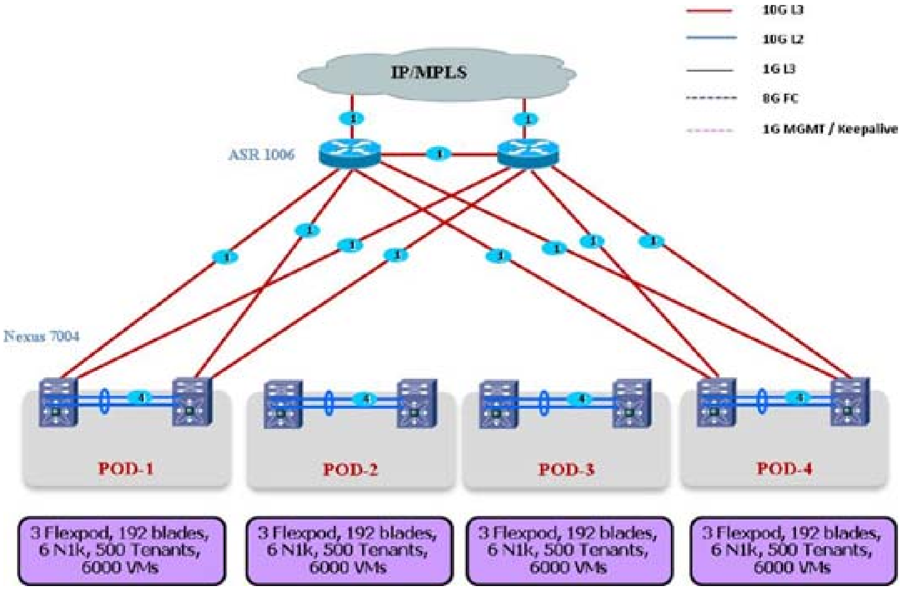


*Figure 2-33 VMDC 2.3 Scaled DC*


IP/MPLS
ASR 1006
Nexus 7004
POD-1
POD-2
POD-3
POD-4
3 Flexpod, 192 blades, 6 N1k, 500 Tenants, 6000 VMs
3 Flexpod, 192 blades, 6 N1k, 500 Tenants, 6000 VMs
3 Flexpod, 192 blades, 6 N1k, 500 Tenants, 6000 VMs
3 Flexpod, 192 blades, 6 N1k, 500 Tenants, 6000 VMs
10G L3
10G L2
1G L3
8G FC
1G MGMT / Keepalive

The VMDC 2.3 system can thus be scaled out horizontally - ICS stacks within a pod, and pods within the DC. This scale-out model is built on mixed tenancy of 10 Expanded Gold, 20 Silver, 220 Bronze, and 250 SMB containers. This requires two ASA 5585-X40s, four ACE 4710s, and two ASA 5555-Xs per pod. This includes three FlexPods (24 UCS chassis, 192 UCS blades) per pod, and four pods per DC. This design is for 500 mixed tenants and 6000 mixed VMs per pod, and 2000 mixed tenants and 24,000 mixed VMs per DC.

Figure 2-31 outlines the ICS (FlexPod) topology and port density/connectivity for the VMDC 2.3 system.

*Figure 2-34 VMDC 2.3 Integrated Compute and Storage Stack*

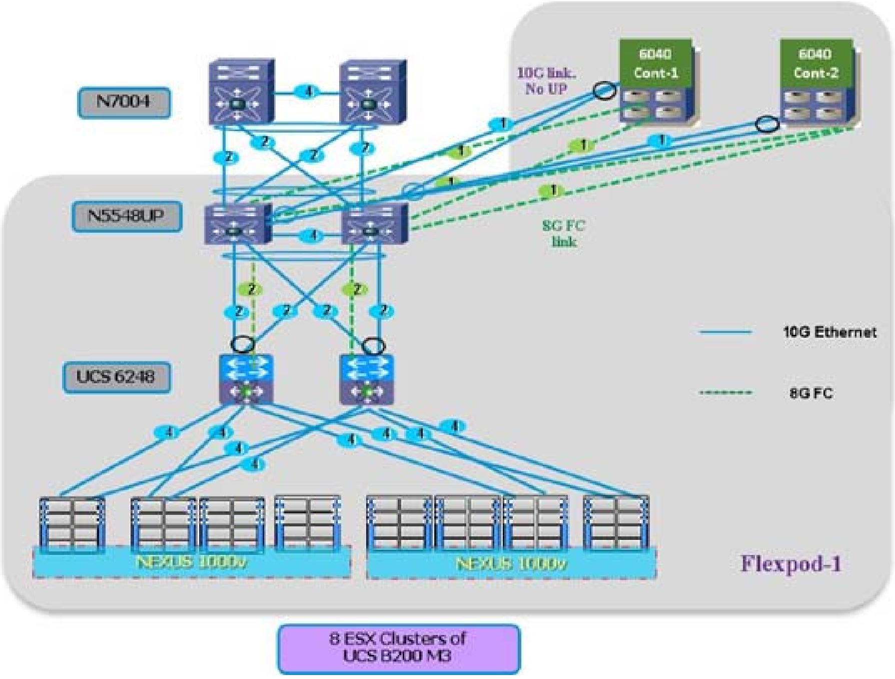


Table 2-4 lists the scale points for different tenant container types in the VMDC 2.3 solution.

*Table 2-4 VMDC 2.3 Tenancy Scale*

| Tenancy Model | Scale per Pod | Scale in DC (4 Pod per DC) |
|---|---|---|
| All Expanded Gold containers | 125 | 500 |
| All Gold containers | 200 | 800 |
| All Silver containers | 300[1] | 1200 |
| All Bronze containers | 300 | 1200 |
| All Copper containers | 500 | 2000 |
| Mixed containers[2] | 500 | 2000 |

[1] Needs multiple pairs of ASA and/or ACE appliances per pod.

[2] Mixed = 10 Expanded Gold, 20 Silver, 220 Bronze, and 250 Copper containers.

Table 2-5 lists the Compute scale points in the VMDC 2.3 solution.

*Table 2-5 VMDC 2.3 Compute Scale in Pod and DC*

| | ICS | Pod | DC | Per VM[1] |
|---|---|---|---|---|
| VM[2] | 2000 | 6000 | 24,000 | |
| CPU | 128 | 384 | 1536 | |

*Table 2-5 VMDC 2.3 Compute Scale in Pod and DC (continued)*

| | ICS | Pod | DC | Per VM[1] |
|---|---|---|---|---|
| Cores | 1024 | 3072 | 1288 | 0.512 |
| GHz | 2969.6 | 8908.8 | 35635.2 | 1.48 |
| GB | 12288 | 36864 | 147456 | 6.14 |

[1] Actual VM sizing and distribution ratios will be defined in SPCSS FlexPod and VM sizing definitions

[2] Assuming 2000 VM in each scaled ICS.

**Note** Scaled ICS = 64 UCS Blades, Scaled pod = 192 blades, Scaled DC = 768 blades.
Assuming UCS B200 M2 blades, each with 2 2.90 GHz E5-2690 CPU, 24 8GB DDR3-1600- MHz RDIMM, 1 VIC 1240 MLOM.

Table 2-6 lists the resources consumed in a scaled-out DC based on the VMDC 2.3 solution, when considering a mixed-tenancy model of 10 ExpGold, 20 Silver, 220 Bronze, and 250 Copper containers,

*Table 2-6 VMDC 2.3 Resources Consumed in a Scaled DC*

| Resource | Per Pod | Per DC (Four Pods) |
|---|---|---|
| Nexus 7004 VRF | 520 | 2080 |
| Nexus 7004 VLAN / HSRP | 860 | 3440 |
| Nexus 7004 BGP | 750 | 3000 |
| Nexus 7004 MAC | 14,000 | 56,000 |
| ASR 9000 VRF | 250 | 1000 |
| ASR 9000 Subinterface | 250 | 1000 |
| ASR 9000 BGP | 500 | 200 |

# GLOSSARY

## A

| | |
|---|---|
| **ACE** | Application Control Engine (Cisco) |
| **ACL** | Access Control List |
| **ARP** | Address Resolution Protocol |
| **ASA** | Adaptive Security Appliance (Cisco) |
| **ASR** | Aggregation Services Router (Cisco) |

## B

| | |
|---|---|
| **BFD** | Bidirectional Forwarding Detection |
| **BGP** | Border Gateway Protocol |

## C

| | |
|---|---|
| **CaaS** | Communications as a Service |
| **CAPEX** | Capital Expense |
| **CBWFQ** | Class-Based Weighted Fair Queuing (QoS) |
| **CNA** | Converged Network Adapter |
| **CoS** | Class of Service |
| **CPU** | Central Processing Unit |
| **CRI** | Cloud Ready Infrastructure |

## D

| | |
|---|---|
| **DC** | Data Center |
| **DC-PE** | Data Center Provider Edge |

| | |
|---|---|
| **DMZ** | Demilitarized Zone |
| **DRS** | Distributed Resource Scheduling |
| **DSCP** | Differentiated Services Code Point |
| **DSN** | Data Center Service Node |
| **DVS** | Distributed Virtual Switch |

## E

| | |
|---|---|
| **eBGP** | External Border Gateway Protocol |
| **EH** | End-host (mode) |

## F

| | |
|---|---|
| **FC** | Fibre Channel |
| **FCoE** | Fibre Channel over Ethernet |
| **FEX** | Fabric Extender |
| **FIFO** | First In, First Out |
| **FT** | Fault Tolerance |
| **FWSM** | Firewall Services Module (Cisco) |

## G

| | |
|---|---|
| **GEM** | Gigabit Ethernet Module |

## H

| | |
|---|---|
| **HA** | High Availability |
| **HQoS** | Hierarchical QoS |
| **HSRP** | Hot Standby Router Protocol |

# I

| | |
|---|---|
| **IaaS** | Infrastructure as a Service |
| **iBGP** | Internal Border Gateway Protocol |
| **ICS** | Integrated Compute and Storage |
| **IGP** | Interior Gateway Protocol |
| **IOPS** | Input/Output Operations Per Second |
| **IPsec** | IP security |
| **ISV** | Independent Software Vendor |

# L

| | |
|---|---|
| **L2** | Layer 2 |
| **L3** | Layer 3 |
| **L4** | Layer 4 |
| **L7** | Layer 7 |
| **LACP** | Link Aggregation Control Protocol |
| **LAN** | Local Area Network |
| **LDAP** | Lightweight Directory Access Protocol |
| **LDP** | Label Distribution Protocol |
| **LLQ** | Low Latency Queuing |
| **LUN** | Logical Unit Number |

# M

| | |
|---|---|
| **MAC** | Media Access Control |
| **MDS** | Multilayer Director Switch |
| **MEC** | Multi-Chassis EtherChannel |
| **MPLS** | Multiprotocol Label Switching |

## N

| | |
|---|---|
| **NAS** | Network Attached Storage |
| **NAT** | Network Address Translation |
| **NFS** | Network File System |
| **NGN** | Next Generation Network |
| **NPIV** | N-Port Identifier Virtualization |
| **NPV** | N-Port Virtualization |
| **NSF** | Nonstop Forwarding (Cisco) |
| **NSR** | Nonstop Routing (Cisco) |

## O

| | |
|---|---|
| **OPEX** | Operating Expense |
| **OSPF** | Open Shortest Path First |

## P

| | |
|---|---|
| **P2MP** | Point-to-Multipoint |
| **PE** | Provider Edge |
| **PHB** | Per-Hop Behavior |
| **Pod** | Point of Delivery. A basic infrastructure module that is a physical, repeatable construct with predictable infrastructure characteristics and deterministic functions. A pod identifies a modular unit of data center components and enables customers to add network, compute, and storage resources incrementally. |
| **PQ** | Priority Queuing |

## Q

| | |
|---|---|
| **QoS** | Quality of Service |

## R

| | |
|---|---|
| **RAID** | Redundant Array of Independent Disks |
| **RBAC** | Role-based Access Control |

## S

| | |
|---|---|
| **SAN** | Storage Area Network |
| **SLA** | Service Level Agreement |
| **SLB** | Server Load Balancing |
| **SMB** | Small/Medium Business |
| **SSL** | Secure Sockets Layer |
| **STP** | Spanning Tree Protocol |
| **SVI** | Switched Virtual Interface |

## T

| | |
|---|---|
| **TCP** | Transmission Control Protocol |
| **ToR** | Top-of-Rack |
| **TTM** | Time to Market |

## U

| | |
|---|---|
| **UCS** | Unified Computing System (Cisco) |
| **UDP** | User Datagram Protocol |
| **UIM** | Unified Infrastructure Manager (EMC) |

## V

| | |
|---|---|
| **vCPU** | Virtual CPU |
| **vDC** | Virtual Data Center |
| **VEM** | Virtual Ethernet Module |
| **vHBA** | Virtual Host Bus Adapter |
| **VIC** | Virtual Interface Card |

| | |
|---|---|
| **VLAN** | Virtual LAN |
| **VM** | Virtual Machine |
| **VMDC** | Virtualized Multiservice Data Center (Cisco) |
| **VMDK** | Virtual Machine Disk |
| **VMFS** | Virtual Machine File System |
| **VMNIC** | VM Network Interface Card |
| **vNIC** | Virtual Network Interface Card |
| **vMotion** | Virtual Motion |
| **VNMC** | Virtual Network Management Center (Cisco) |
| **VoIP** | Voice over IP |
| **vPC** | Virtual Port-channel |
| **VPDC** | Virtual Private Data Center |
| **VRF** | Virtual Routing and Forwarding |
| **VSAN** | Virtual SAN |
| **VSG** | Virtual Security Gateway (Cisco) |
| **vSLB** | Virtual Server Load Balancer |
| **VSM** | Virtual Supervisor Module (Cisco) |
| **VSS** | Virtual Switch System (Cisco) |

## W

| | |
|---|---|
| **WAN** | Wide Area Network |
| **WRED** | Weighted Random Early Detection |