

FlexPod Data Center with Oracle RAC on Oracle Linux

Deployment Guide for Oracle Database 12c RAC with Oracle Linux 6.5 on NetApp FAS 8000 Series Running Clustered ONTAP 8.3

Last Updated: October 23, 2015

About Cisco Validated Designs

The CVD 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/designzone>.

ALL DESIGNS, SPECIFICATIONS, STATEMENTS, INFORMATION, AND RECOMMENDATIONS (COLLECTIVELY, "DESIGNS") IN THIS MANUAL ARE PRESENTED "AS IS," WITH ALL FAULTS. CISCO AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE WARRANTY 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 THE DESIGNS, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

THE DESIGNS ARE SUBJECT TO CHANGE WITHOUT NOTICE. USERS ARE SOLELY RESPONSIBLE FOR THEIR APPLICATION OF THE DESIGNS. THE DESIGNS DO NOT CONSTITUTE THE TECHNICAL OR OTHER PROFESSIONAL ADVICE OF CISCO, ITS SUPPLIERS OR PARTNERS. USERS SHOULD CONSULT THEIR OWN TECHNICAL ADVISORS BEFORE IMPLEMENTING THE DESIGNS. RESULTS MAY VARY DEPENDING ON FACTORS NOT TESTED BY CISCO.

CCDE, CCENT, Cisco Eos, Cisco Lumin, Cisco Nexus, Cisco StadiumVision, Cisco TelePresence, Cisco WebEx, the Cisco logo, DCE, and Welcome to the Human Network are trademarks; Changing the Way We Work, Live, Play, and Learn and Cisco Store are service marks; and Access Registrar, Aironet, AsyncOS, Bringing the Meeting To You, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, CCVP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Collaboration Without Limitation, EtherFast, EtherSwitch, Event Center, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, iQuick Study, IronPort, the IronPort logo, LightStream, Linksys, MediaTone, MeetingPlace, MeetingPlace Chime Sound, MGX, Networkers, Networking Academy, Network Registrar, PCNow, PIX, PowerPanels, ProConnect, ScriptShare, SenderBase, SMARTnet, Spectrum Expert, StackWise, The Fastest Way to Increase Your Internet Quotient, TransPath, WebEx, and the WebEx logo are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0809R)

© 2015 Cisco Systems, Inc. All rights reserved.

Table of Contents

About Cisco Validated Designs	2
Executive Summary	5
Solution Overview.....	6
Introduction	6
Audience.....	6
Purpose of this Document.....	6
FlexPod Program Benefits.....	6
FlexPod System Overview	7
Solution Summary.....	9
Oracle Database 12c R1 RAC on FlexPod with Oracle Direct NFS Client.....	9
FlexPod Oracle RAC Solution Design Principles.....	9
Technology Overview	10
Cisco Unified Computing System.....	10
Cisco UCS B-Series Blade Servers.....	11
Cisco UCS C-Series Rack-Mount Server.....	11
Cisco UCS Adapters.....	11
Cisco UCS Manager	11
Cisco Unified Fabric.....	11
Cisco UCS 6200 Series Fabric Interconnects.....	12
Cisco UCS 5108 Blade Server Chassis	13
Cisco UCS 5100 Series Blade Server Chassis	13
Cisco UCS 2208XP Fabric Extenders.....	13
Cisco UCS Blade Servers.....	14
Cisco Virtual Interface Card 1340	15
Cisco Wire-Once Model	16
Cisco UCS Manager	17
Cisco UCS Service Profile.....	18
Cisco Nexus 9396PX Switches.....	21
NetApp FAS8080EX Hybrid Storage	22
NetApp FAS8000	22
NetApp Storage Controllers.....	23
NetApp Clustered Data ONTAP 8.3 Fundamentals.....	24
Scale-Out.....	25

Non-disruptive Operations.....	26
Availability	27
NetApp Advanced Data Management Capabilities.....	27
Unified Storage Architecture and Multiprotocol Support	28
Storage Virtual Machines.....	29
Oracle 12c (12.1.0.1) Database.....	30
Oracle Direct NFS.....	31
Solution Design.....	32
Design Topology.....	32
Inventory and Bill of Materials	32
Physical Topology.....	33
Deployment Hardware and Software	35
Configuring Cisco Unified Computing System.....	35
Configure/Verify Global Chassis Discovery Policy	35
Create Port-Channels for Both Fabric Interconnect A and B.....	37
Configure Cisco Nexus 9396PX Switches.....	52
Configuring NetApp Storage.....	63
OS and Database Deployment	68
OS Configuration	68
Miscellaneous Post-Install Steps	69
Oracle Database 12c GRID Infrastructure Deployment	71
Validation.....	75
Test Plan	75
Infrastructure Tests.....	75
Database Creation with DBCA	76
Direct NFS Configuration	76
Database Workload Configuration.....	79
Database Performance	79
Resiliency Tests.....	85
Summary	97
Appendix.....	98
Cisco Nexus 9396PX Running Configuration.....	98
Sysctl.conf settings.....	104
About Authors	108
Acknowledgements	108

Executive Summary

The Cisco Unified Computing System™ (Cisco UCS®) is a next-generation data center platform that unites computing, network, storage access, and virtualization into a single cohesive system. Cisco UCS is an ideal platform for the architecture of mission critical database workloads. The combination of Cisco UCS platform, NetApp® storage, and Oracle Real Application Cluster (RAC) architecture can accelerate your IT transformation by enabling faster deployments, greater flexibility of choice, efficiency, and lower risk. This Cisco Validated Design (CVD) highlights a flexible, multitenant, highly performant and resilient FlexPod® reference architecture featuring the Oracle 12c RAC Database.

The FlexPod platform, developed by NetApp and Cisco, is a flexible, integrated infrastructure solution that delivers pre-validated storage, networking, and server technologies. It's designed to increase IT responsiveness to business demands while reducing the overall cost of computing. Think maximum uptime, minimal risk. FlexPod components are integrated and standardized to help you achieve timely, repeatable, consistent deployments. You can plan with accuracy the power, floor space, usable capacity, performance, and cost of each FlexPod deployment.

FlexPod embraces the latest technology and efficiently simplifies the data center workloads that redefine the way IT delivers value:

- NetApp FAS Hybrid Arrays with Flash Pool™ flash provide the capability to deploy the precise proportion of flash to spinning media for your specific application or environment.
- Leverage a pre-validated platform to minimize business disruption and improve IT agility and reduce deployment time from months to weeks.
- Slash administration time and total cost of ownership (TCO) by 50 percent.
- Meet or exceed constantly expanding hardware performance demands for data center workloads.

Solution Overview

Introduction

Data powers essentially every operation in a modern enterprise, from keeping the supply chain operating efficiently to managing relationships with customers. Database administrators and their IT departments face many challenges that demand needs for a simplified deployment and operation model providing high performance, availability and lower TCO. The current industry trend in data center design is towards shared infrastructures featuring multitenant workload deployments. By moving away from application silos and toward shared infrastructure that can be quickly deployed, customers increase agility and reduce costs. Cisco® and NetApp® have partnered to deliver FlexPod, which uses best-in-class storage, server, and network components to serve as the foundation for a variety of workloads, enabling efficient architectural designs that can be quickly and confidently deployed. This CVD describes how Cisco UCS can be used in conjunction with NetApp FAS storage systems to implement an Oracle Real Application Clusters (RAC) 12c solution that is an Oracle Certified Configuration.

Audience

The intended audience for this document includes, but is not limited to, sales engineers, field consultants, professional services, IT managers, partner engineering, and customers who want to deploy Oracle RAC12c database on FlexPod architecture with NetApp clustered Data ONTAP® and the Cisco UCS platform. Readers of this document need to have experience installing and configuring the solution components used to deploy the FlexPod Datacenter solution.

Purpose of this Document

This FlexPod CVD demonstrates how enterprises can apply best practices to deploy Oracle RAC 12c Database using Cisco Unified Computing System, Cisco Nexus family switches, and NetApp FAS storage systems. This validation effort exercised typical Online transaction processing (OLTP) and Decision-support systems (DSS) workloads to ensure expected stability, performance and resiliency design as demanded by mission critical data center deployments.

FlexPod Program Benefits

Cisco and NetApp have carefully validated and verified the FlexPod solution architecture and its many use cases while creating a portfolio of detailed documentation, information, and references to assist customers in transforming their data centers to this shared infrastructure model. This portfolio includes, but is not limited to, the following items:

- Best practice architectural design
- Workload sizing and scaling guidance
- Implementation and deployment instructions
- Technical specifications (rules for what is a FlexPod configuration)

- Frequently asked questions (FAQs)
- Cisco Validated Designs (CVDs) and NetApp Validated Architectures (NVAs) covering a variety of use cases

Cisco and NetApp have also built a robust and experienced support team focused on FlexPod solutions, from customer account and technical sales representatives to professional services and technical support engineers. The support alliance between NetApp and Cisco gives customers and channel services partners direct access to technical experts who collaborate with cross vendors and have access to shared lab resources to resolve potential issues.

FlexPod supports tight integration with virtualized and cloud infrastructures, making it the logical choice for long-term investment. FlexPod also provides a uniform approach to IT architecture, offering a well-characterized and documented shared pool of resources for application workloads. FlexPod delivers operational efficiency and consistency with the versatility to meet a variety of SLAs and IT initiatives, including:

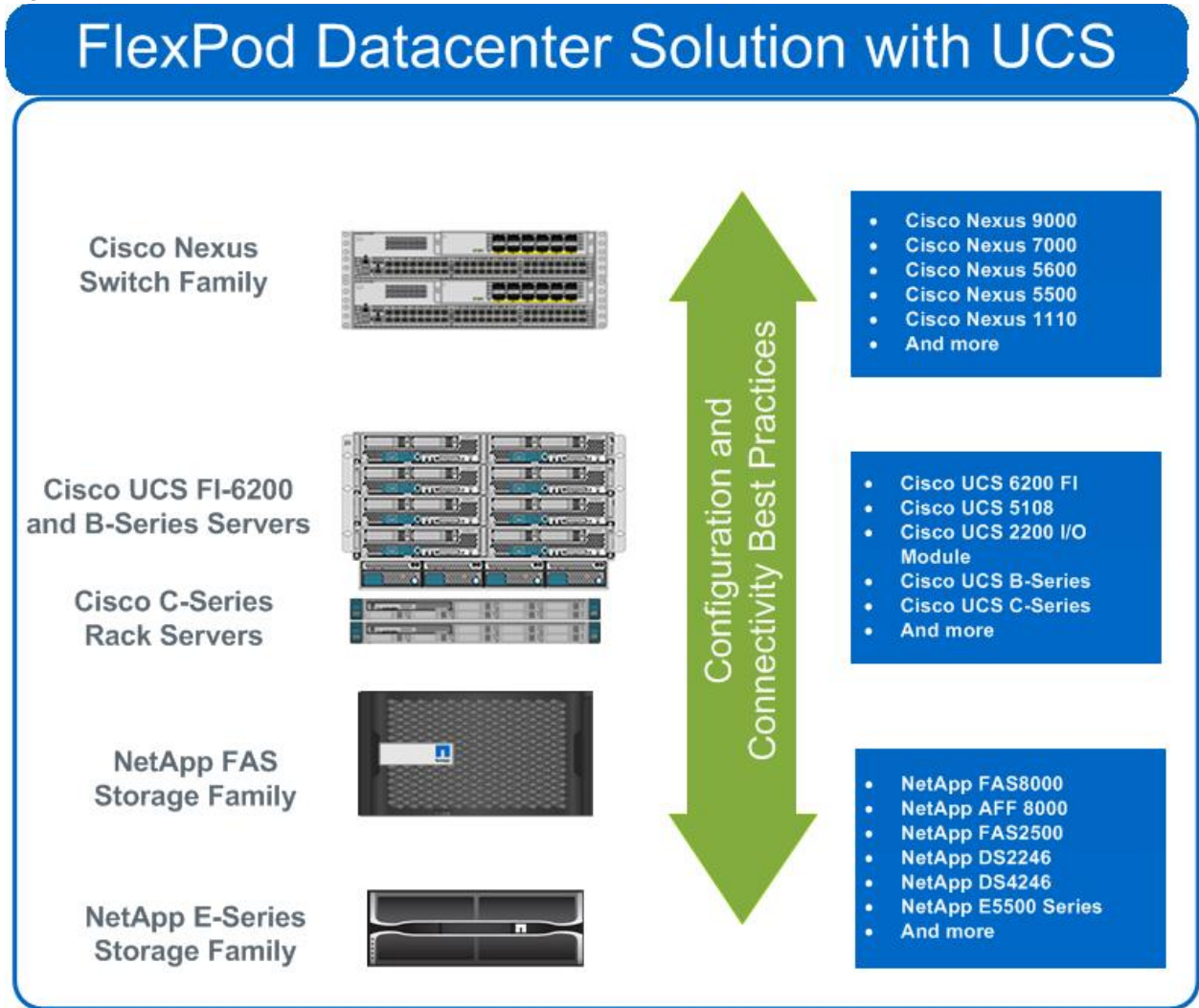
- Application rollouts or application migrations
- Business continuity and disaster recovery
- Desktop virtualization
- Cloud delivery models (public, private, and hybrid) and service models (IaaS, PaaS, and SaaS)
- Asset consolidation and virtualization

FlexPod System Overview

FlexPod is a best practice data center architecture that includes these three components:

- Cisco Unified Computing System (Cisco UCS)
- Cisco Nexus switches
- NetApp fabric-attached storage (FAS) systems

Figure 1 FlexPod Components



As shown in Figure 1, these components are connected and configured according to best practices of both Cisco and NetApp and provide the ideal platform for running a variety of enterprise workloads with confidence. FlexPod can scale up for greater performance and capacity (adding compute, network, or storage resources individually as needed), or it can scale out for environments that require multiple consistent deployments (rolling out additional FlexPod stacks). The reference architecture covered in this document leverages the Cisco Nexus 9000 for the switching element.

One of the key benefits of FlexPod is the ability to maintain consistency at scale. Each of the component families shown in (Cisco UCS, Cisco Nexus, and NetApp FAS) offers platform and resource options to scale the infrastructure up or down, while supporting the same features and functionality that are required under the configuration and connectivity best practices of FlexPod.

Solution Summary

Oracle Database 12c R1 RAC on FlexPod with Oracle Direct NFS Client

The FlexPod Data Center with Oracle RAC on Oracle Linux solution provides an end-to-end architecture with Cisco UCS, Oracle, and NetApp technologies and demonstrates the FlexPod configuration benefits for running Oracle Database 12c RAC with Cisco VICs (Virtual Interface Cards) and Oracle Direct NFS Client.

The following infrastructure and software components are used for this solution:

- Oracle Database 12c R1 RAC
- Cisco UCS
- Cisco Nexus 9000 switches
- NetApp FAS 8080EX storage system and supporting components
- NetApp OnCommand® System Manager
- Swingbench, a benchmark kit for online transaction processing (OLTP) and decision support system (DSS) workloads

FlexPod Oracle RAC Solution Design Principles

The FlexPod for Oracle RAC solution addresses the following primary design principles:

- **Application availability.** Makes sure that application services are accessible, easy to configure, and ready to use once configured.
- **Scalability.** Addresses increasing demands with appropriate resources.
- **Flexibility.** Provides new services or recovers resources without requiring infrastructure modification.
- **Manageability.** Facilitates efficient infrastructure operations through open standards and APIs.
- **Infrastructure resiliency.** Eliminates all single points of failure in the solution. Seamless failover and protection against common hardware/cable failures.

Technology Overview

This section provides a technical overview of products used in this solution.

Cisco Unified Computing System

Figure 2 Cisco UCS Components



Cisco UCS is a next-generation solution for blade and rack server computing. The system integrates a low-latency, lossless 10 Gigabit Ethernet (10GbE) unified network fabric with enterprise-class, x86-architecture servers. The system is an integrated, scalable, multi-chassis platform in which all resources participate in a unified management domain. Cisco UCS accelerates the delivery of new services simply, reliably, and securely through end-to-end provisioning and migration support for both virtualized and non-virtualized systems.

The Cisco UCS consists of the following main components:

- **Compute.** The system is based on an entirely new class of computing system that incorporates rack mount and blade servers based on Intel Xeon 2600 v2 Series Processors.
- **Network.** The system is integrated onto a low-latency, lossless, 10Gbps unified network fabric. This network foundation consolidates LANs, SANs, and high-performance computing networks which are separate networks today. The unified fabric lowers costs by reducing the number of network adapters, switches, and cables, and by decreasing the power and cooling requirements.
- **Virtualization.** The system unleashes the full potential of virtualization by enhancing the scalability, performance, and operational control of virtual environments. Cisco security, policy enforcement, and diagnostic features are now extended into virtualized environments to better support changing business and IT requirements.
- **Storage access.** The system provides consolidated access to both SAN storage and network-attached storage (NAS) over the unified fabric. By unifying the storage access, Cisco UCS can access storage over Ethernet (SMB 3.0 or iSCSI), Fibre Channel (FC), and FCoE. This provides customers with storage choices and investment protection. In addition, the server administrators can preassign storage-access policies to storage resources, for simplified storage connectivity and management leading to increased productivity.

- **Management.** The system uniquely integrates all system components to enable the entire solution to be managed as a single entity by the Cisco UCS Manager. The Cisco UCS Manager has an intuitive graphical user interface (GUI), a command-line interface (CLI), and a powerful scripting library module for Microsoft PowerShell built on a robust application programming interface (API) to manage all system configuration and operations.

Cisco UCS B-Series Blade Servers

Cisco UCS B-Series Blade Servers increase performance, efficiency, versatility and productivity with these Intel-based blade servers. For detailed information, refer to:

<http://www.cisco.com/c/en/us/products/servers-unified-computing/ucs-b-series-blade-servers/index.html>.

Cisco UCS C-Series Rack-Mount Server

Cisco UCS C-Series Rack-Mount Servers delivers unified computing in an industry-standard form factor to reduce total cost of ownership and increase agility. For detailed information, refer to:

<http://www.cisco.com/c/en/us/products/servers-unified-computing/ucs-c-series-rack-servers/index.html>

Cisco UCS Adapters

Cisco UCS Adapters with wire-once architecture, offers a range of options to converge the fabric, optimize virtualization, and simplify management. For detailed information, refer to:

<http://www.cisco.com/c/en/us/products/interfaces-modules/unified-computing-system-adapters/index.html>.

Cisco UCS Manager

Cisco UCS Manager provides unified, embedded management of all software and hardware components in Cisco UCS. Cisco UCS fuses access layer networking and servers. This high-performance, next-generation server system provides a data center with a high degree of workload agility and scalability. For detailed information, refer to: <http://www.cisco.com/c/en/us/products/servers-unified-computing/ucs-manager/index.html>.

Cisco Unified Fabric

The fabric interconnects provide a single point for connectivity and management for the entire system. Typically deployed as an active-active pair, the system's fabric interconnects integrate all components into a single, highly-available management domain controlled by Cisco UCS Manager. The fabric interconnects manage all I/Os efficiently and securely at a single point, resulting in deterministic I/O latency regardless of a server or virtual machine's topological location in the system.

Cisco UCS 6200 Series Fabric Interconnects support the system's 80Gbps unified fabric with low-latency, lossless, cut-through switching that supports IP, storage, and management traffic using a single set of cables. The fabric interconnects feature virtual interfaces that terminate both physical and virtual connections equivalently, establishing a virtualization-aware environment in which blade, rack servers, and virtual machines are interconnected using the same mechanisms. The Cisco UCS 6248UP is a 1-RU Fabric Interconnect that features up to 48 universal ports that can support 80GbE, FCoE, or native FC connectivity.

Cisco UCS 6200 Series Fabric Interconnects

Cisco UCS 6200 Series Fabric Interconnects is a family of line-rate, low-latency, lossless, 10-Gbps Ethernet and Fibre Channel over Ethernet (FCoE) interconnect switches providing the management and communication backbone for Cisco UCS. For detailed information, refer to:

<http://www.cisco.com/c/en/us/products/servers-unified-computing/ucs-6200-series-fabric-interconnects/index.html>.

Figure 3 Cisco UCS 6248 Fabric Interconnect

UCS 6248: Unified Ports

Dynamic Port Allocation: Lossless Ethernet or Fibre Channel

FC

Native Fibre Channel

Eth

Lossless Ethernet:
1/10GbE, FCoE, iSCSI, NAS

Benefits	Use-cases
<ul style="list-style-type: none"> Simplify switch purchase - remove ports ratio guess work Increase design flexibility Remove specific protocol bandwidth bottlenecks 	<ul style="list-style-type: none"> Flexible LAN & storage convergence based on business needs Service can be adjusted based on the demand for specific traffic

Figure 4 Cisco UCS 6000 Series Fabric Interconnects

UCS 6000 Series Fabric Interconnects

Unified Computing Market Leading Platforms

Product Features and Specs	UCS 6120XP	UCS 6140XP	UCS 6248UP	UCS 6296UP
Switch Fabric Throughput	520 Gbps	1.04 Tbps	960 Gbps	1.92 Tbps
Switch Footprint	1RU	2RU	1RU	2RU
1 Gigabit Ethernet Port Density	8	16	48	96
10 Gigabit Ethernet Port Density	26	52	48	96
8G Native FC Port Density	6	12	48	96
Port-to-Port Latency	3.2us	3.2us	2.0us	2.0us
# of VLANs	1024	1024	4096*	4096*
Layer 3 Capability			✓*	✓*
40 Gigabit Ethernet Ready			✓*	✓*
DC Power Supply	No	No	Oct '11	

BACKWARD COMPATIBILITY

AND

FORWARD COMPATIBILITY

*Not at ECS - Radar Features

© 2013 Cisco and/or its affiliates. All rights reserved. Cisco Confidential 12

Cisco UCS 5108 Blade Server Chassis

The Cisco UCS 5100 Series Blade Server Chassis is a crucial building block of the Cisco UCS, delivering a scalable and flexible blade server chassis.

The Cisco UCS 5108 Blade Server Chassis is six rack units (6RUs) high and can mount in an industry-standard 19-inch rack. A single chassis can house up to eight half-width Cisco UCS B-Series Blade Servers and can accommodate both half-width and full-width blade form factors.

Four single-phase, hot-swappable power supplies are accessible from the front of the chassis. These power supplies are 92 percent efficient and can be configured to support non-redundant, N+ 1 redundant and grid-redundant configurations. The rear of the chassis contains eight hot-swappable fans, four power connectors (one per power supply), and two I/O bays for Cisco UCS 2208 XP Fabric Extenders.

A passive mid-plane provides up to 40Gbps of I/O bandwidth per server slot and up to 80Gbps of I/O bandwidth for two slots. The chassis is capable of supporting future 80 Gigabit Ethernet standards.

Figure 5 Cisco UCS Blade Server Front View and Rear View



Cisco UCS 5100 Series Blade Server Chassis

Cisco UCS 5100 Series Blade Server Chassis supports up to eight blade servers and up to two fabric extenders in a six-rack unit (RU) enclosure. For detailed information, refer to:

<http://www.cisco.com/c/en/us/products/servers-unified-computing/ucs-5100-series-blade-server-chassis/index.html>.

Cisco UCS 2208XP Fabric Extenders

The Cisco UCS 2208XP Fabric Extender has eight 10GbE, FCoE-capable, and Enhanced Small Form-Factor Pluggable (SFP+) ports that connect the blade chassis to the fabric interconnect. Each Cisco UCS 2208XP has 32 x 10GbE ports connected through the midplane to each half-width slot in the chassis. Typically configured in pairs for redundancy, two fabric extenders provide up to 160Gbps of I/O to the chassis.

Figure 6 Cisco UCS Fabric Extender



Cisco UCS Blade Servers










Cisco UCS offers a variety of x86-based compute portfolio to address the needs of today’s workloads. Based on Intel® Xeon® processor E7 and E5 product families, Cisco UCS B-Series Blade Servers work with virtualized and non-virtualized applications to increase:

- Performance
- Energy efficiency
- Flexibility
- Administrator productivity

Figure 7 provides a complete summary of fourth generation Cisco UCS compute portfolio featuring Cisco UCS Blade and Rack-Mount Servers.

Figure 7 Cisco UCS Compute Portfolio

UCS Compute Portfolio
Performance Optimized for Bare Metal, Virtualized, and Cloud Applications

 UCS B460 M4 Mission-Critical, 4-Socket Blade for CPU-Intensive Bare Metal and Virtualized Applications Double-Height Blade – Four E7v2 CPUs – 96 DIMM Slots – Up to 320Gb of IO	 UCS C460 M4 Mission-Critical Server for Large, CPU-Intensive Applications 4 RU - Four E7v2 CPUs – 96 DIMM Slots - 10 PCIe Slots
 UCS B260 M4 Mission-Critical, 2-Socket Blade for CPU-Intensive Bare Metal and Virtualized Applications Full-Width Blade – Two E7v2 CPUs – 48 DIMM Slots – Up to 160Gb of IO	 UCS C240 M4 Optimal Platform for Big Data, ERP, and Database Applications 2 RU - Two E5v3 CPUs – 24 DIMM Slots - 6 PCIe Slots – up to 26 HDD
 UCS B420 M4 Dense 4-Socket Blade for -Intensive Bare Metal and Virtualized Applications Full-Width Blade - Four E5v3 CPUs – 48 DIMM Slots – Up to 160Gb of IO	 UCS C220 M4 Versatile, General Purpose Enterprise Infrastructure, and Application Server 1 RU – Two E5v3 CPUs – 24 DIMM Slots - 2 PCIe Slots – up to 8 HDD
 UCS B200 M4 Ideal for Bare Metal Enterprise, VDI, or Dense Virtualization/ Consolidation Workloads Half-Width Blade – Two E5v3 CPUs – 24 DIMM Slots - up to 80Gb of IO	 UCS C3160 Ideal for large unstructured data repositories, media streaming and content distribution 4 RU - Two E5v2 CPUs – 16 DIMM Slots –up to 60 LFF HDD - up to 240TB
 UCS M-Series Designed for Cloud Scale Applications and Deployment 2 RU – 16 E3v3 Servers –4 DIMMs / Server - up to 32 Cores / RU	

For this Oracle RAC solution, we used enterprise-class, Cisco UCS B460 Blade Servers. The Cisco UCS B460 M4 Blade Server provides industry-leading performance and enterprise-critical stability for memory-intensive workloads such as:

- Large-scale databases
- In-memory analytics
- Business intelligence

Figure 8 Cisco UCS Blade Server



The Cisco UCS B460 M4 Blade Server uses the power of the latest Intel® Xeon® processor E7 v3 product family to add new levels of performance and capabilities to the innovative Cisco UCS combines Cisco UCS B-Series Blade Servers and C-Series Rack Servers with networking and storage access resources into a single converged system that greatly simplifies server management and delivers greater cost efficiency and agility. It also offers advances in fabric-centric computing, open APIs, and application-centric management, and uses service profiles to automate all aspects of server deployment and provisioning.

The Cisco UCS B460 M4 harnesses the power of four Intel® Xeon® processor E7 v3 product families and accelerates access to critical data. This blade server supports up to 72 processor cores, 6.0TB of memory, 4.8TB of internal storage, and 320Gbps of overall Ethernet throughput.

In addition, the fabric-centric, architectural advantage of Cisco UCS means that you do not need to purchase, maintain, power, cool, and license excess switches and interface cards in each Cisco UCS blade chassis, enabling Cisco to design uncompromised expandability and versatility in its blade servers. As a result, with their leading CPU core count, frequencies, memory slots, expandability, and drive capacities, the Cisco UCS B-Series Blade Servers offer uncompromised expandability, versatility, and performance.

The Cisco UCS B460 M4 provides:

- Four Intel® Xeon® processor E7 v3 product families
- 96 DDR3 memory DIMM slots
- Four hot-pluggable drive bays for hard-disk drives (HDDs) or solid-state drive (SSDs)
- SAS controller on board with RAID 0 and 1 support
- Two modular LAN on motherboard (mLOM) slots for Cisco UCS Virtual Interface Card (VIC)
- Six PCIe mezzanine slots, with two dedicated for optional Cisco UCS VIC 1340, and four slots for Cisco UCS VIC 1380, VIC port expander, or flash cards

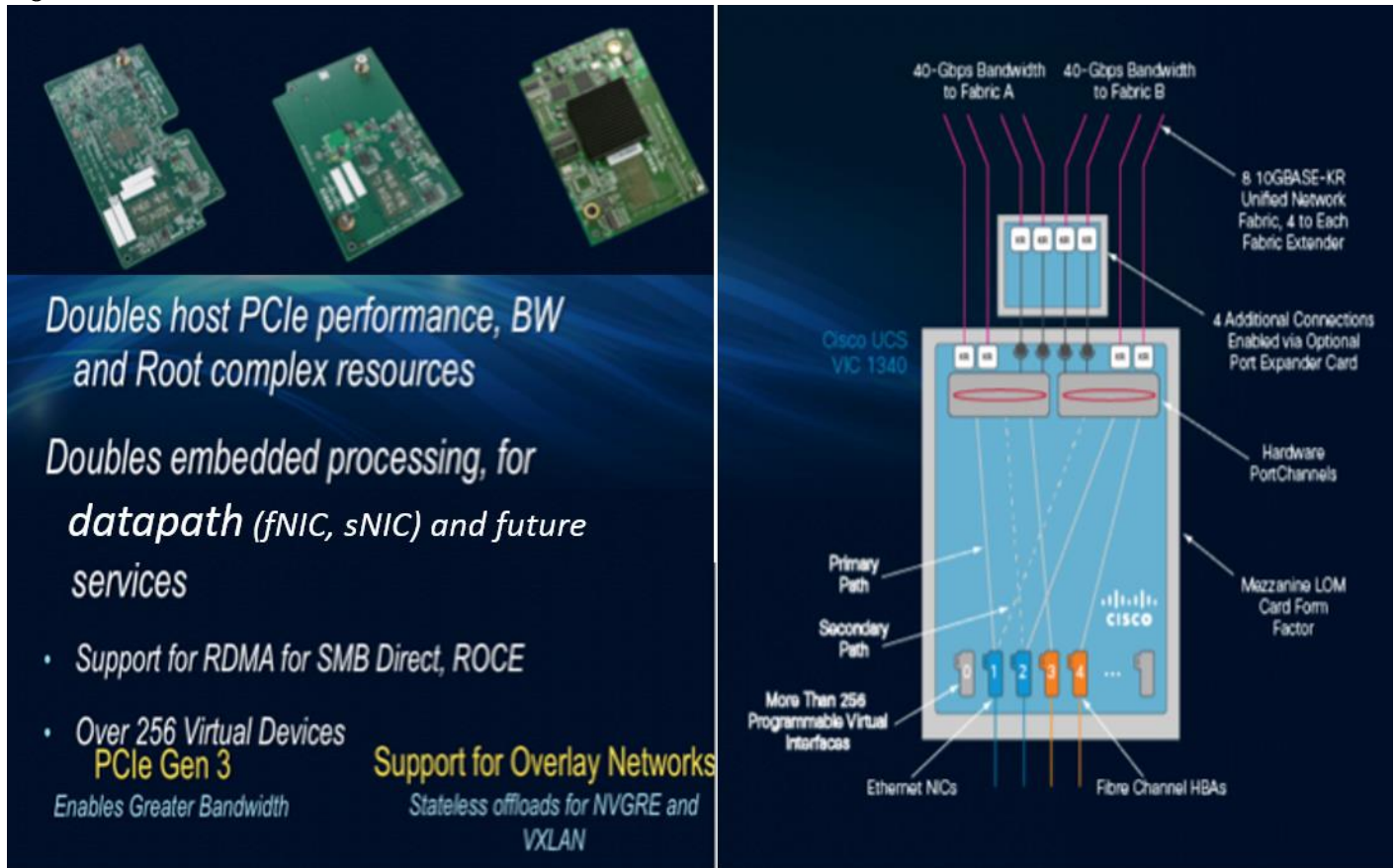
Cisco Virtual Interface Card 1340

The Cisco UCS VIC 1340 is a 2-port 40Gbps Ethernet or dual 4 x 10Gbps Ethernet, FCoE-capable modular LAN on motherboard (mLOM) designed exclusively for the M4 generation of Cisco UCS B-Series Blade

Servers. When used in combination with an optional port expander, the Cisco UCS VIC 1340 capabilities is enabled for two ports of 40Gbps Ethernet.

The Cisco UCS VIC 1340 enables a policy-based, stateless, agile server infrastructure that can present over 256 PCIe standards-compliant interfaces to the host that can be dynamically configured as either network interface cards (NICs) or host bus adapters (HBAs). In addition, the Cisco UCS VIC 1340 supports Cisco® Data Center Virtual Machine Fabric Extender (VM-FEX) technology, which extends the Cisco UCS fabric interconnect ports to virtual machines, simplifying server virtualization deployment and management.

Figure 9 Cisco Virtual Interface Card



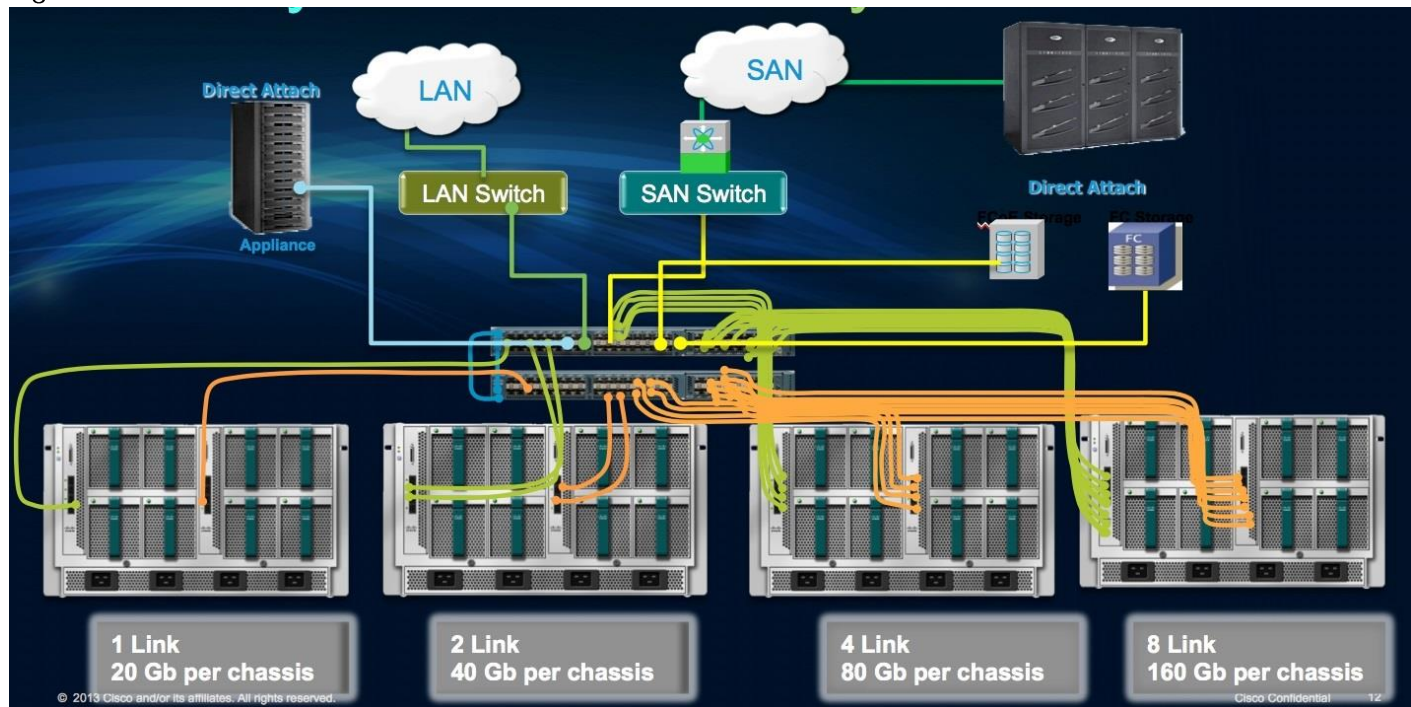
Cisco Wire-Once Model

Cisco UCS is designed with a "wire once, walk away" model in which:

- Cabling and network infrastructure support a unified network fabric in which features such as FCoE can be enabled through Cisco UCS Manager as needed.
- Every element in the hierarchy is programmable and managed by Cisco UCS Manager using a just-in-time resource provisioning model.
- The manager can configure identity information including the universally unique identifier (UUID) of servers, MAC addresses, and WWNs of network adapters.

- It can install consistent sets of firmware throughout the system hierarchy, including each blade's baseboard management controller (BMC), RAID controller, network adapter firmware, and fabric extender firmware.
- It can configure the operational characteristics of every component in the hierarchy, from the hardware RAID level of onboard disk drives to uplink port configurations on the Cisco UCS 6200 Series Fabric Interconnects and everything in between.
- It can configure the types of I/O interfaces on Cisco UCS VIC adapters. The importance of this capability cannot be understated: when a server resource is configured with this mezzanine card, the number, type (HBA or NIC), and identities (WWNs and MAC addresses) of I/O interfaces can be programmed using just-in-time provisioning. This approach allows a server resource to support a traditional OS and application software stack with a pair of Ethernet NICs and FC HBAs at one moment and then be rebooted to run a virtualized environment with a combination of up to 128 NICs and HBAs, with NICs connected directly to virtual machines through hypervisor pass-through technology.

Figure 10 Cisco Wire-Once Model



Cisco UCS Manager

Cisco UCS Manager provides unified, centralized, embedded management of all Cisco UCS software and hardware components across multiple chassis and thousands of virtual machines. Administrators use the software to manage the entire Cisco UCS as a single logical entity through an intuitive GUI, a command-line interface (CLI), or an XML API.

The Cisco UCS Manager resides on a pair of Cisco UCS 6200 Series Fabric Interconnects using a clustered, active-standby configuration for high availability (HA). The software gives administrators a single interface for performing server provisioning, device discovery, inventory, configuration, diagnostics, monitoring, fault

detection, auditing, and statistics collection. Cisco UCS Manager Service profiles and templates support versatile role- and policy-based management, and system configuration information can be exported to configuration management databases (CMDBs) to facilitate processes based on IT Infrastructure Library (ITIL) concepts. Service profiles benefit both virtualized and non-virtualized environments and increase the mobility of non-virtualized servers, such as when moving workloads from server to server or taking a server offline for service or upgrade. Profiles can also be used in conjunction with virtualization clusters to bring new resources online easily, complementing existing virtual machine mobility.

Some of the key elements managed by Cisco UCS Manager include:

- Cisco UCS Integrated Management Controller (IMC) firmware
- RAID controller firmware and settings
- BIOS firmware and settings, including server universal user ID (UUID) and boot order
- Converged network adapter (CNA) firmware and settings, including MAC addresses and worldwide names (WWNs) and SAN boot settings
- Virtual port groups used by virtual machines, using Cisco Data Center VM-FEX technology
- Interconnect configuration, including uplink and downlink definitions, MAC address and WWN pinning, VLANs, VSANs, quality of service (QoS), bandwidth allocations, Cisco Data Center VM-FEX settings, and Ether Channels to upstream LAN switches

For more information on Cisco UCS Manager, refer to:

<http://www.cisco.com/en/US/products/ps10281/index.html>

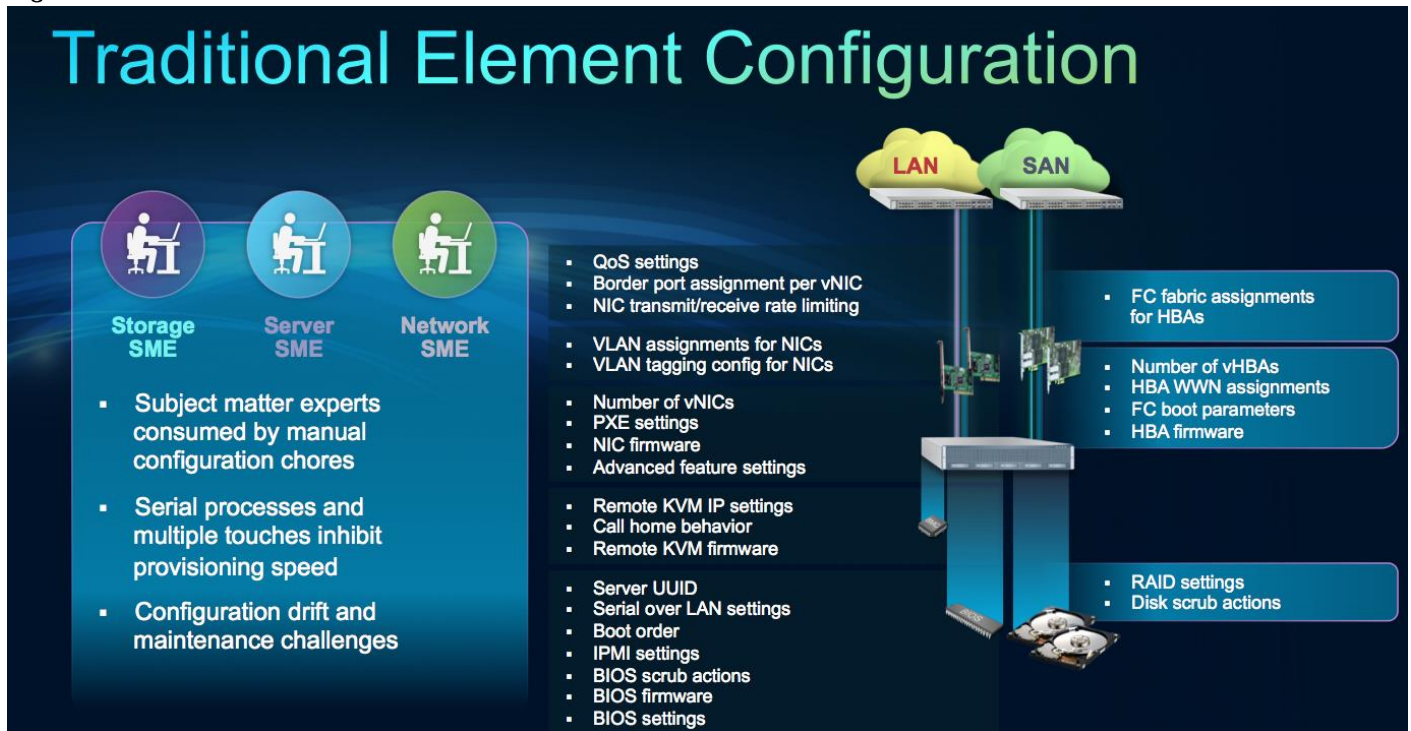
Cisco UCS Service Profile

A server's identity is made up of many properties such as UUID, boot order, IPMI settings, BIOS firmware, BIOS settings, RAID settings, disk scrub settings, number of NICs, NIC speed, NIC firmware, MAC and IP addresses, number of HBAs, HBA WWNs, HBA firmware, FC fabric assignments, QoS settings, VLAN assignments, remote keyboard/video/monitor etc. I think you get the idea. It's a LONG list of "points of configuration" that need to be configured to give this server its identity and make it unique from every other server within your data center. Some of these parameters are kept in the hardware of the server itself (like BIOS firmware version, BIOS settings, boot order, FC boot settings, etc.) while some settings are kept on your network and storage switches (like VLAN assignments, FC fabric assignments, QoS settings, ACLs, etc.). This results in the following server deployment challenges:

- Lengthy deployment cycles
 - Every deployment requires coordination among server, storage, and network teams
 - Need to ensure correct firmware and settings for hardware components
 - Need appropriate LAN and SAN connectivity
- Response time to business needs
 - Tedious deployment process

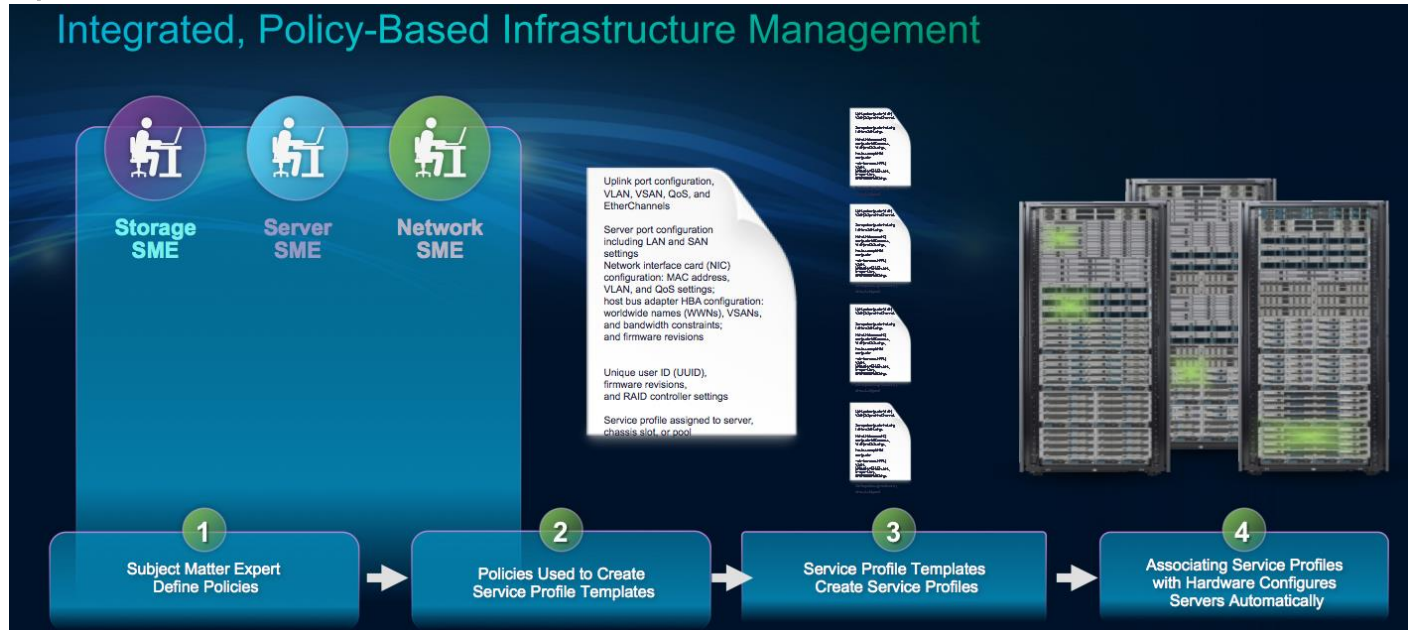
- Manual, error prone processes, that are difficult to automate
- High OPEX costs, outages caused by human errors
- Limited OS and application mobility
 - Storage and network settings tied to physical ports and adapter identities
 - Static infrastructure leads to over-provisioning, higher OPEX costs

Figure 11 Cisco UCS Service Profile



Cisco UCS has uniquely addressed these challenges with the introduction of service profiles that enables integrated, policy based infrastructure management. Cisco UCS Service Profiles hold the DNA for nearly all configurable parameters required to set up a physical server. A set of user defined policies (rules) allow quick, consistent, repeatable, and secure deployments of Cisco UCS servers.

Figure 12 Service Profile Infrastructure



Cisco UCS Service Profiles contain values for a server's property settings, including virtual network interface cards (vNICs), MAC addresses, boot policies, firmware policies, fabric connectivity, external management, and HA information. By abstracting these settings from the physical server into a Cisco Service Profile, the Service Profile can then be deployed to any physical compute hardware within the Cisco UCS domain. Furthermore, Service Profiles can, at any time, be migrated from one physical server to another. This logical abstraction of the server personality separates the dependency of the hardware type or model and is a result of Cisco's unified fabric model (rather than overlaying software tools on top).

This innovation is still unique in the industry despite competitors claiming to offer similar functionality. In most cases, these vendors must rely on several different methods and interfaces to configure these server settings. Furthermore, Cisco is the only hardware provider to offer a truly unified management platform, with Cisco UCS Service Profiles and hardware abstraction capabilities extending to both blade and rack servers.

Some of key features and benefits of Cisco UCS service profiles are detailed below:

- Service profiles and templates.** Service profile templates are stored in the Cisco UCS 6200 Series Fabric Interconnects for reuse by server, network, and storage administrators. Service profile templates consist of server requirements and the associated LAN and SAN connectivity. Service profile templates allow different classes of resources to be defined and applied to a number of resources, each with its own unique identities assigned from predetermined pools.

The Cisco UCS Manager can deploy the service profile on any physical server at any time. When a service profile is deployed to a server, the Cisco UCS Manager automatically configures the server, adapters, fabric extenders, and fabric interconnects to match the configuration specified in the service profile. A service profile template parameterizes the UUIDs that differentiate between server instances.

This automation of device configuration reduces the number of manual steps required to configure servers, Network Interface Cards (NICs), Host Bus Adapters (HBAs), and LAN and SAN switches.

- **Programmatically deploying server resources.** Cisco UCS Manager provides centralized management capabilities, creates a unified management domain, and serves as the central nervous system of the Cisco UCS. Cisco UCS Manager is embedded device management software that manages the system from end-to-end as a single logical entity through an intuitive GUI, CLI, or XML API. Cisco UCS Manager implements role- and policy-based management using service profiles and templates. This construct improves IT productivity and business agility. Now infrastructure can be provisioned in minutes instead of days, shifting IT's focus from maintenance to strategic initiatives.
- **Dynamic provisioning.** Cisco UCS resources are abstract in the sense that their identity, I/O configuration, MAC addresses and WWNs, firmware versions, BIOS boot order, and network attributes (including QoS settings, ACLs, pin groups, and threshold policies) all are programmable using a just-in-time deployment model. A service profile can be applied to any blade server to provision it with the characteristics required to support a specific software stack. A service profile allows server and network definitions to move within the management domain, enabling flexibility in the use of system resources. Service profile templates allow different classes of resources to be defined and applied to a number of resources, each with its own unique identities assigned from predetermined pools.

Cisco Nexus 9396PX Switches

The Cisco Nexus 9396X Switch delivers comprehensive line-rate layer 2 and layer 3 features in a two-rack unit (2RU) form factor. It supports line rate 1/10/40 GE with 960Gbps of switching capacity. It is ideal for top-of-rack and middle-of-row deployments in both traditional and Cisco Application Centric Infrastructure (ACI) – enabled enterprise, service provider, and cloud environments.

Figure 13 Cisco Nexus 9396PX Switch



The Cisco Nexus 9396PX switch features are capabilities are listed below.

ACI Support

- Designed to support the Cisco next-generation data center based on Cisco ACI

Programmability

- Open APIs to manage the switch through remote-procedure calls (JavaScript Object Notation or XML) over HTTP or HTTPS
- Automated configuration through Linux shell scripts
- Support for customer applications through a Linux container environment

High Availability

- Hot-swappable, redundant power supplies and fan trays
- Full patching support

Energy Efficiency

- Power supplies that are rated at 80 Plus Platinum
- Support for front-to-back or back-to-front airflow configuration

Investment Protection

- 40Gb bidirectional transceiver for reuse of existing 10 Gigabit Ethernet multimode cabling plant for 40GbE
- Support for Cisco NX-OS and ACI modes of operation

Specifications At-a-Glance

- Forty-eight 1/10 Gigabit Ethernet Small Form-Factor Pluggable (SFP+) non-blocking ports
- Twelve 40GbE Quad SFP+ (QSFP+) non-blocking ports
- Low latency (approximately 2 microseconds)
- 50 MB of shared buffer
- Line rate VXLAN bridging, routing, and gateway support
- FCoE capability
- Front-to-back or back-to-front airflow

NetApp FAS8080EX Hybrid Storage

NetApp FAS8000

The FlexPod Datacenter solution includes the NetApp fabric-attached storage (FAS) 8080 series unified scale-out storage system for Oracle Database 12c. Powered by NetApp clustered Data ONTAP, the FAS8000 series unifies the SAN and NAS storage infrastructure. Systems architects can choose from a range of models representing a spectrum of cost-versus-performance points. Every model, however, provides the following core benefits:

- **HA and fault tolerance.** Storage access and security are achieved through clustering, HA pairing of controllers, hot-swappable components, NetApp RAID DP® disk protection (allowing two independent disk failures without data loss), network interface redundancy, support for data mirroring with NetApp SnapMirror® software, application backup integration with the NetApp SnapManager® storage management software, and customizable data protection with the NetApp Snap Creator® framework and SnapProtect® products.

- **Storage efficiency.** Users can store more data with less physical media. This is achieved with thin provisioning (unused space is shared among volumes), NetApp Snapshot® copies (zero-storage, read-only clones of data over time), NetApp FlexClone® volumes, and LUNs (read/write copies of data in which only changes are stored), deduplication (dynamic detection and removal of redundant data), and data compression.
- **Unified storage architecture.** Every model runs the same software (clustered Data ONTAP); supports all popular storage protocols (CIFS, NFS, iSCSI, FCP, and FCoE); and uses SATA, SAS, or SSD storage (or a mix) on the back end. This allows freedom of choice in upgrades and expansions, without the need for re-architecting the solution or retraining operations personnel.
- **Advanced clustering.** Storage controllers are grouped into clusters for both availability and performance pooling. Workloads can be moved between controllers, permitting dynamic load balancing and zero-downtime maintenance and upgrades. Physical media and storage controllers can be added as needed to support growing demand without downtime.

NetApp Storage Controllers

A storage system running Data ONTAP (also known as a storage controller) is the hardware device that receives and sends data from the host. Controller nodes are deployed in HA pairs, with these HA pairs participating in a single storage domain or cluster. This unit detects and gathers information about its own hardware configuration, the storage system components, the operational status, hardware failures, and other error conditions. A storage controller is redundantly connected to storage through disk shelves, which are the containers or device carriers that hold disks and associated hardware such as power supplies, connectivity interfaces, and cabling.

The NetApp FAS8000 features a multicore Intel chipset and leverages high-performance memory modules, NVRAM to accelerate and optimize writes, and an I/O-tuned PCIe gen3 architecture that maximizes application throughput. The FAS8000 series come with integrated unified target adapter (UTA2) ports that support 16GB FC, 10GbE, or FCoE. Figure 14 shows a front and rear view of the FAS8080/8060/8040 controllers.

Figure 14 NetApp FAS8080/8060/8040 (6U) – Front and Rear View



If storage requirements change over time, NetApp storage offers the flexibility to change quickly without expensive and disruptive forklift upgrades. This applies to the following different types of changes:

- Physical changes, such as expanding a controller to accept more disk shelves and subsequently more hard disk drives (HDDs) without an outage

- Logical or configuration changes, such as expanding a RAID group to incorporate these new drives without requiring any outage
- Access protocol changes, such as modification of a virtual representation of a hard drive to a host by changing a logical unit number (LUN) from FC access to iSCSI access, with no data movement required, but only a simple dismount of the FC LUN and a mount of the same LUN, using iSCSI

In addition, a single copy of data can be shared between Linux and Windows systems while allowing each environment to access the data through native protocols and applications. In a system that was originally purchased with all SATA disks for backup applications, high-performance solid-state disks can be added to the same storage system to support Tier-1 applications, such as Oracle®, Microsoft Exchange, or Microsoft SQL Server.

NetApp Clustered Data ONTAP 8.3 Fundamentals

NetApp provides enterprise-ready, unified scale out storage with clustered Data ONTAP 8.3, the operating system physically running on the storage controllers in the NetApp FAS storage appliance. Developed from a solid foundation of proven Data ONTAP technology and innovation, clustered Data ONTAP is the basis for large virtualized shared-storage infrastructures that are architected for non-disruptive operations over the system lifetime.



Clustered Data ONTAP 8.3 is the first Data ONTAP release that supports clustered operation only. The previous version of Data ONTAP, 7-Mode, is not available as a mode of operation in version 8.3

Data ONTAP scale-out is one way to respond to growth in a storage environment. All storage controllers have physical limits to their expandability; the number of CPUs, memory slots, and space for disk shelves dictate maximum capacity and controller performance. If more storage or performance capacity is needed, it might be possible to add CPUs and memory or install additional disk shelves, but ultimately the controller becomes completely populated, with no further expansion possible. At this stage, the only option is to acquire another controller. One way to do this is to scale up; that is, to add additional controllers in such a way that each is an independent management entity that does not provide any shared storage resources. If the original controller is completely replaced by a newer, larger controller, data migration is required to transfer the data from the old controller to the new one. This is time consuming and potentially disruptive and most likely requires configuration changes on all of the attached host systems.

If the newer controller can coexist with the original controller, then the two storage controllers must be individually managed, and there are no native tools to balance or reassign workloads across them. The situation becomes worse as the number of controllers increases. If the scale-up approach is used, the operational burden increases consistently as the environment grows, and the end result is a very unbalanced and difficult-to-manage environment. Technology refresh cycles require substantial planning in advance, lengthy outages, and configuration changes, which introduce risk into the system.

In contrast, when using a scale-out approach, additional controllers are added seamlessly to the resource pool residing on a shared storage infrastructure as the storage environment grows. Host and client connections as well as volumes can move seamlessly and non-disruptively anywhere in the resource pool, so that existing workloads can be easily balanced over the available resources, and new workloads can be easily deployed. Technology refreshes (replacing disk shelves, adding or completely replacing storage controllers) are accomplished while the environment remains online and continues serving data.

Although scale-out products have been available for some time, these were typically subject to one or more of the following shortcomings:

- Limited protocol support—NAS only.
- Limited hardware support—supported only a particular type of storage controller or a very limited set.
- Little or no storage efficiency—thin provisioning, deduplication, or compression.
- Little or no data replication capability.

Therefore, while these products are well-positioned for certain specialized workloads, they are less flexible, less capable, and not robust enough for broad deployment throughout the enterprise.

Data ONTAP is the first product to offer a complete scale-out solution, and it offers an adaptable, always-available storage infrastructure for today's highly virtualized environment.

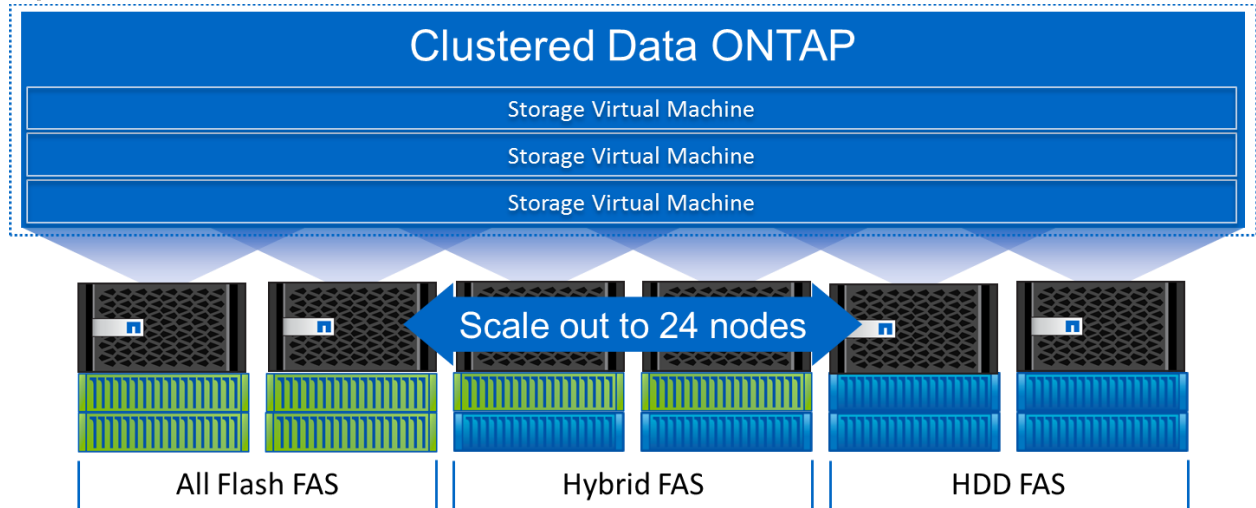
Scale-Out

Data centers require agility. In a data center, each storage controller has CPU, memory, and disk shelf limits. Scale-out means that as the storage environment grows, additional controllers can be added seamlessly to the resource pool residing on a shared storage infrastructure. Host and client connections as well as volumes can be moved seamlessly and non-disruptively anywhere within the resource pool. The benefits of scale-out include:

- Non-disruptive operations
- The ability to add tenants, instances, volumes, networks, and so on without downtime for Oracle databases
- Operational simplicity and flexibility

As Figure 15 illustrates, clustered Data ONTAP offers a way to solve the scalability requirements in a storage environment. A clustered Data ONTAP system can scale up to 24 nodes, depending on platform and protocol, and can contain different disk types and controller models in the same storage cluster with up to 101PB of capacity.

Figure 15 NetApp Clustered Data ONTAP



Non-disruptive Operations

The move to shared infrastructure has made it nearly impossible to schedule downtime for routine maintenance. NetApp clustered Data ONTAP is designed to eliminate the need for planned downtime for maintenance operations and lifecycle operations as well as the unplanned downtime caused by hardware and software failures. NetApp storage solutions provide redundancy and fault tolerance through clustered storage controllers and redundant, hot-swappable components, such as cooling fans, power supplies, disk drives, and shelves. This highly available and flexible architecture enables customers to manage all data under one common infrastructure while meeting mission-critical uptime requirements.

Three standard tools that eliminate the possible downtime:

- NetApp DataMotion™ data migration software for volumes (vol move)—allows you to move data volumes from one aggregate to another on the same or a different cluster node.
- Logical interface (LIF) migration—allows you to virtualize the physical Ethernet interfaces in clustered Data ONTAP. LIF migration allows the administrator to move these virtualized LIFs from one network port to another on the same or a different cluster node.
- Aggregate relocate (ARL)—allows you to transfer complete aggregates from one controller in an HA pair to the other without data movement.

Used individually and in combination, these tools allow you to non-disruptively perform a full range of operations, from moving a volume from a faster to a slower disk all the way up to a complete controller and storage technology refresh.

As storage nodes are added to the system, all physical resources (CPUs, cache memory, network I/O bandwidth, and disk I/O bandwidth) can be easily kept in balance. NetApp Data ONTAP enables users to perform the following tasks:

- Move data between storage controllers and tiers of storage without disrupting users and applications
- Dynamically assign, promote, and retire storage, while providing continuous access to data as administrators upgrade or replace storage

- Increase capacity while balancing workloads and reduce or eliminate storage I/O hot spots without the need to remount shares, modify client settings, or stop running applications

These features allow a truly non-disruptive architecture in which any component of the storage system can be upgraded, resized, or re-architected without disruption to the private cloud infrastructure.

Availability

Shared storage infrastructure can provide services to many different databases and other enterprise applications. In such environments, downtime produces disastrous effects. The NetApp FAS eliminates sources of downtime and protects critical data against disaster through two key features:

- HA. A NetApp HA pair provides seamless failover to its partner in the event of a hardware failure. Each of the two identical storage controllers in the HA pair configuration serves data independently during normal operation. During an individual storage controller failure, the data service process is transferred from the failed storage controller to the surviving partner.
- RAID DP. During any Oracle database deployment, data protection is critical because any RAID failure might disconnect and/or shut off hundreds or potentially thousands of end users from database instances, resulting in lost productivity. RAID DP provides performance comparable to that of RAID 10 and yet it requires fewer disks to achieve equivalent protection. RAID DP provides protection against double-disk failure, in contrast to RAID 5, which can protect against only one disk failure per RAID group, in effect providing RAID 10 performance and protection at a RAID 5 price point.

NetApp Advanced Data Management Capabilities

This section describes the storage efficiencies, advanced storage features, and multiprotocol support capabilities of the NetApp FAS8000 storage controller.

Storage Efficiencies

NetApp FAS includes built-in thin provisioning, data deduplication, compression, and zero-cost cloning with FlexClone technology, offering multilevel storage efficiency across Oracle databases, installed applications, and user data. This comprehensive storage efficiency enables a significant reduction in storage footprint, with a capacity reduction of up to 10:1, or 90% (based on existing customer deployments and NetApp solutions lab validation). Four features make this storage efficiency possible:

- Thin provisioning—allows multiple applications to share a single pool of on-demand storage, eliminating the need to provision more storage for one application if another application still has plenty of allocated but unused storage.
- Deduplication—saves space on primary storage by removing redundant copies of blocks in a volume that hosts hundreds of instances. This process is transparent to the application and the user, and it can be enabled and disabled on the fly or scheduled to run at off-peak hours.
- Compression—compresses data blocks. Compression can be run whether or not deduplication is enabled and can provide additional space savings whether it is run alone or together with deduplication.

- FlexClone technology—offers hardware-assisted rapid creation of space-efficient, writable, point-in-time images of individual databases, LUNs, or flexible volumes. The use of FlexClone technology in Oracle database deployments allows you to instantly create and replicate Oracle databases in minutes using less storage space than other vendors. These virtual copies of production data allow developers to work on essential business operations such as patching, development, testing, reporting, training, and quality assurance without disrupting business continuity. Leveraged with virtualization software, FlexClone technology enables DBAs to spin up customized isolated environments including OS platforms and databases for each development project, eliminating sharing of databases. And, when time is critical, you can leverage NetApp FlexVol® to provision storage resources for additional application development projects or urgent testing of emergency patches.

Advanced Storage Features

NetApp Data ONTAP provides the following additional features:

- NetApp Snapshot technology—a manual or automatically scheduled point-in-time copy that writes only changed blocks, with no performance penalty. A Snapshot copy consumes minimal storage space because only changes to the active file system are written. Individual files and directories can be easily recovered from any Snapshot copy, and the entire volume can be restored back to any Snapshot state in seconds. A Snapshot copy incurs no performance overhead. Users can comfortably store up to 255 Snapshot copies per FlexVol volume, all of which are accessible as read-only and online versions of the data.
- LIF—a logical interface that is associated with a physical port, interface group, or VLAN interface. More than one LIF can be associated with a physical port at the same time. There are three types of LIFs: NFS, iSCSI, and FC. LIFs are logical network entities that have the same characteristics as physical network devices but are not tied to physical objects. LIFs used for Ethernet traffic are assigned specific Ethernet-based details such as IP addresses and iSCSI qualified names and are associated with a specific physical port capable of supporting Ethernet. LIFs used for FC-based traffic are assigned specific FC-based details such as worldwide port names (WWPNs) and are associated with a specific physical port capable of supporting FC or FCoE. NAS LIFs can be non-disruptively migrated to any other physical network port throughout the entire cluster at any time, either manually or automatically (by using policies), whereas SAN LIFs rely on multipath input/output (MPIO) and Asymmetric Logical Unit Access (ALUA) to notify clients of any change in the network topology.
- Storage virtual machines (SVMs; formerly known as Vserver)—an SVM is a secure virtual storage server that contains data volumes and one or more LIFs 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 an SVM administrator.

Unified Storage Architecture and Multiprotocol Support

NetApp also offers the NetApp Unified Storage Architecture. The term unified refers to a family of storage systems that simultaneously support SAN (through FCoE, FC, and iSCSI) and network-attached storage (NAS) (through CIFS and NFS) across many operating environments, including Oracle databases, VMware®,

Windows, Linux, and UNIX. This single architecture provides access to data by using industry-standard protocols, including NFS, CIFS, iSCSI, FCP, SCSI, and NDMP.

Connectivity options include standard Ethernet (10/100/1000Mb or 10GbE) and FC (4, 8, or 16 Gb/sec). In addition, all systems can be configured with high-performance solid-state drives (SSDs) or serial-attached SCSI (SAS) disks for primary storage applications, low-cost SATA disks for secondary applications (backup, archive, and so on), or a mix of the different disk types. By supporting all common NAS and SAN protocols on a single platform, NetApp FAS provides the following benefits:

- Direct access to storage for each client
- Network file sharing across different platforms without the need for protocol-emulation products such as SAMBA, NFS Maestro, or PC-NFS
- Simple and fast data storage and data access for all client systems
- Fewer storage systems
- Greater efficiency from each system deployed

Clustered Data ONTAP can support several protocols concurrently in the same storage system and data replication, and its storage efficiency features are supported across the following protocols:

- NFS v3, v4, and v4.1 (including pNFS)
- iSCSI
- FC
- FCoE
- SMB 1, 2, 2.1, and 3

Storage Virtual Machines

SVM is the secure logical storage partition through which data is accessed in clustered Data ONTAP. A cluster serves data through at least one and possibly multiple SVMs. An SVM is a logical abstraction that represents a set of physical resources of the cluster. Data volumes and logical network LIFs are created and assigned to an SVM and can reside on any node in the cluster to which the SVM has been given access. An SVM can own resources on multiple nodes concurrently, and those resources can be moved non-disruptively from one node to another. For example, a flexible volume can be non-disruptively moved to a new node, and an aggregate, or a data LIF, can be transparently reassigned to a different physical network port. The SVM abstracts the cluster hardware and is not tied to specific physical hardware.

An SVM is capable of supporting multiple data protocols concurrently. Volumes within the SVM can be combined together to form a single NAS namespace, which makes all of an SVM's data available to NFS and CIFS clients through a single share or mount point. For example, a 24-node cluster licensed for UNIX and Windows File Services that has a single SVM configured with thousands of volumes can be accessed from a single network interface on one of the nodes. SVMs also support block-based protocols, and LUNs can be created and exported by using iSCSI, FC, or FCoE. Any or all of these data protocols can be configured for use within a given SVM.

An SVM is a secure entity; therefore, it is aware of only 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 can manage the resources allocated to them through a delegated SVM administration account. An SVM is effectively isolated from other SVMs that share the same physical hardware. Each SVM can connect to unique authentication zones, such as AD, LDAP, or NIS.

From a performance perspective, maximum IOPS and throughput levels can be set per SVM by using QoS policy groups, which allow the cluster administrator to quantify the performance capabilities allocated to each SVM.

Clustered Data ONTAP is highly scalable, and additional storage controllers and disks can easily be added to existing clusters to scale capacity and performance to meet rising demands. Because these are virtual storage servers within the cluster, SVMs are also highly scalable. As new nodes or aggregates are added to the cluster, the SVM can be non-disruptively configured to use them. New disk, cache, and network resources can be made available to the SVM to create new data volumes or to migrate existing workloads to these new resources to balance performance.

This scalability also enables the SVM to be highly resilient. SVMs are no longer tied to the lifecycle of a given storage controller. As new replacement hardware is introduced, SVM resources can be moved non-disruptively from the old controllers to the new controllers, and the old controllers can be retired from service while the SVM is still online and available to serve data.

SVMs have three main components:

- **Logical interfaces.** All SVM networking is done through LIFs created within the SVM. As logical constructs, LIFs are abstracted from the physical networking ports on which they reside.
- **Flexible volumes.** A flexible volume is the basic unit of storage for an SVM. An SVM has a root volume and can have one or more data volumes. Data volumes can be created in any aggregate that has been delegated by the cluster administrator for use by the SVM. Depending on the data protocols used by the SVM, volumes can contain either LUNs for use with block protocols, files for use with NAS protocols, or both concurrently. For access using NAS protocols, the volume must be added to the SVM namespace through the creation of a client-visible directory called a junction.
- **Namespaces.** Each SVM has a distinct namespace through which all of the NAS data shared from that SVM can be accessed. This namespace can be thought of as a map to all of the junctioned volumes for the SVM, regardless of the node or the aggregate on which they physically reside. Volumes can be junctioned at the root of the namespace or beneath other volumes that are part of the namespace hierarchy.

Oracle 12c (12.1.0.1) Database

Oracle Database 12c addresses the key challenges of customers who are consolidating databases in a private cloud model by enabling greatly improved efficiency and lower management costs, while retaining the autonomy of separate databases.

Oracle Multitenant is a new feature of Oracle Database 12c, and allows each database plugged into the new multitenant architecture to look and feel like a standard Oracle Database to applications; so existing applications can run unchanged. By supporting multi-tenancy in the database tier, rather than the application tier, Oracle Multitenant makes all ISV applications that run on the Oracle Database ready for SaaS. Oracle

Database 12c multitenant architecture makes it easy to consolidate many databases quickly and manage them as a cloud service. Oracle Database 12c also includes in-memory data processing capabilities delivering breakthrough analytical performance. Additional database innovations deliver new levels of efficiency, performance, security, and availability.

Oracle Database 12c introduces a rich set of new or enhanced features. It's beyond the scope to document all the features of 12c in this solution but we will showcase some of the features such as multitenancy, consolidation, rapid provisioning, and DirectNFS in this solution.

Oracle Real Application Clusters (RAC) harnesses the processing power of multiple, interconnected servers on a cluster; allows access to a single database from multiple servers on a cluster, insulating both applications and database users from server failures, while providing performance that scales out on-demand at low cost; and is a vital component of grid computing that allows multiple servers to access a single database at one time.

Oracle Direct NFS

Standard NFS client software, provided by the operating system, is not optimized for Oracle Database file I/O access patterns. With Oracle Database 11g, you can configure Oracle Database to access NFS V3 NAS devices directly using Oracle Direct NFS Client, rather than using the operating system kernel NFS client. Oracle Database will access files stored on the NFS server directly through the integrated Direct NFS Client eliminating the overhead imposed by the operating system kernel NFS. These files are also accessible via the operating system kernel NFS client thereby allowing seamless administration.

Direct NFS Client overcomes many of the challenges associated with using NFS with the Oracle Database. Direct NFS Client outperforms traditional NFS clients, is simple to configure, and provides a standard NFS client implementation across all hardware and operating system platforms.

Direct NFS Client includes two fundamental I/O optimizations to increase throughput and overall performance. First, Direct NFS Client is capable of performing concurrent direct I/O, which bypasses any operating system level caches and eliminates any operating system write-ordering locks. This decreases memory consumption by eliminating scenarios where Oracle data is cached both in the SGA and in the operating system cache and eliminates the kernel mode CPU cost of copying data from the operating system cache into the SGA. Second, Direct NFS Client performs asynchronous I/O, which allows processing to continue while the I/O request is submitted and processed. Direct NFS Client, therefore, leverages the tight integration with the Oracle Database software to provide unparalleled performance when compared to the operating system kernel NFS clients. Not only does Direct NFS Client outperform traditional NFS, it does so while consuming fewer system resources. The results of a detailed performance analysis are discussed later in this paper.

Oracle Direct NFS Client currently supports up to 4 parallel network paths to provide scalability and HA. Direct NFS Client delivers optimized performance by automatically load balancing requests across all specified paths. If one network path fails, then Direct NFS Client will reissue commands over any remaining paths – ensuring fault tolerance and HA.

Solution Design

Design Topology

This section describes the physical and logical high-level design considerations for the Oracle Database 12c RAC on FlexPod deployment. The following tables list the inventory of the components used in the FlexPod solution stack.

Inventory and Bill of Materials

Table 1 Server Configuration

Cisco UCS Physical Server Configuration	
Description	Quantity
Cisco UCS 5108 Blade Server Chassis, with 4 power supply units, 8 fans, and 2 Fabric Extenders	2
Cisco UCS B460 Servers	4
8 GB DDR3 DIMM, 1666 MHz (64 per server, 256 GB per Server)	256
Cisco VIC 1340 (2 per server)	8
Hard-disk drives (2 per server)	8
Cisco UCS-6248 48 port Fabric Interconnect	2

Table 2 LAN Configuration

LAN Configuration	
Description	Quantity
Cisco Nexus 9396PX switches	2
VLANs:	4
• Public VLAN	135
• Private VLAN (RAC Interconnect)	10
• NFS Storage VLAN A side	20
• NFS Storage VLAN B side	30

Table 3 Storage Configuration

Storage	
FAS 8080EX Controller	2 Nodes configured as Active-Active pair
2x Dual Port 10GbE PCIe adapters	
2x1TB Flash Controllers:	
• 5 x DS4243 Disk Shelves	5
• 24 x 900 GB SAS Drives per shelf	96
• 15 x 400 GB SAS Flash Drives	15

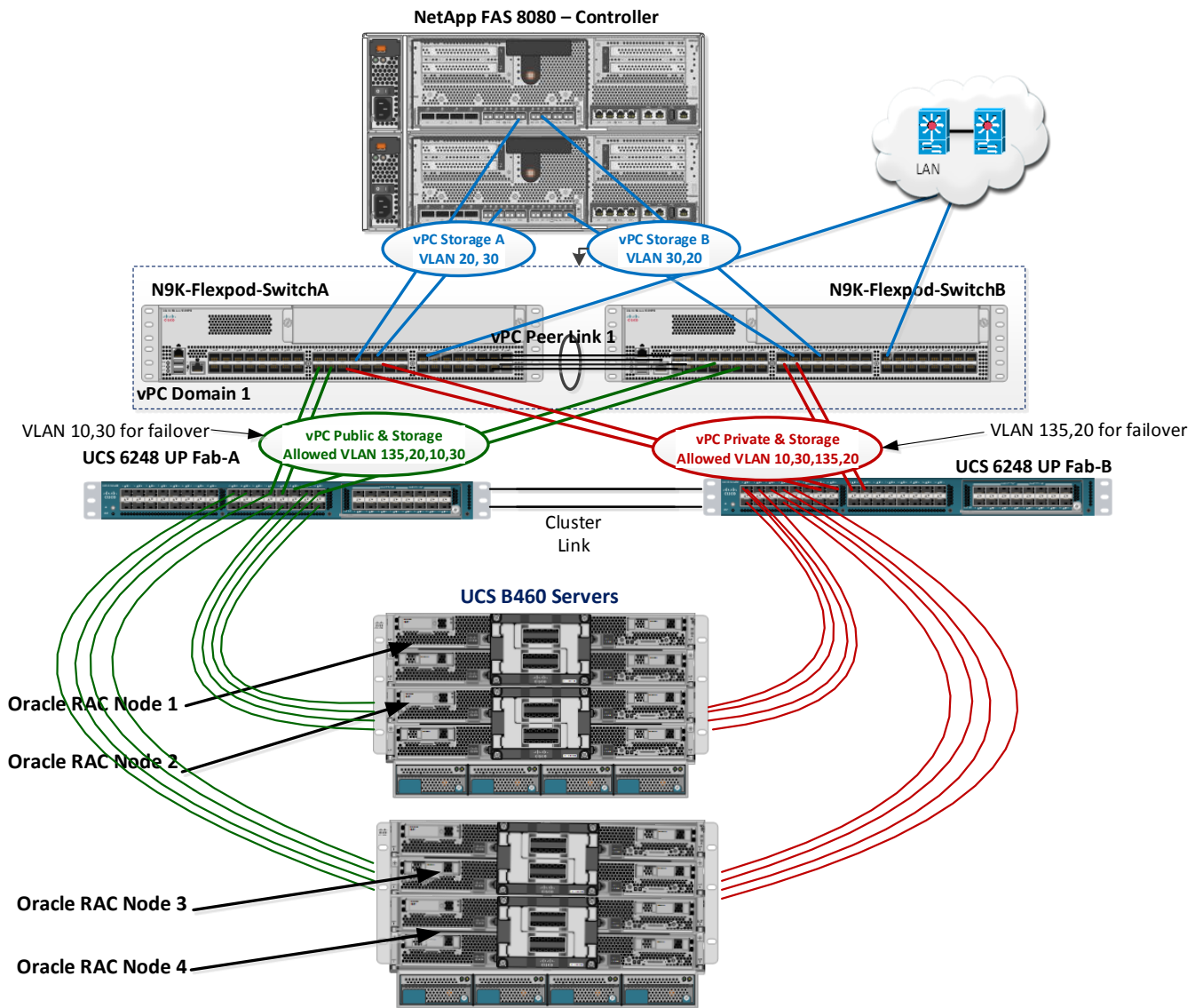
Table 4 OS and Software

Operating System and Software	
Oracle Linux Server 6.5 (64 bit)	Linux orarac1 3.8.13-16.2.1.el6uek.x86_64
Oracle 12c Release 1 GRID	12.1.0.2
Oracle 12c Release 1 Database Enterprise Edition	12.1.0.2
Cisco Nexus 9396PX Switches	7.0(3)I1(2)
Cisco UCS Manager	2.2(3a)
NetApp clustered Data ONTAP	8.3
NetApp OnCommand System Manager	3.1.2

Physical Topology

Figure 16 illustrates the FlexPod solution physical infrastructure.

Figure 16 Physical Topology



In Figure 16, the green lines indicate the public network connecting to fabric interconnect A and the red lines indicate the private interconnects connecting to fabric interconnect B. For Oracle RAC environments, it is a best practice to pin all private interconnect (intra-blade) traffic to one fabric interconnect. The public and private VLANs spanning the fabric interconnects secure connectivity in the event of link failure. Both green and red links also carry NFS storage traffic. Figure 16 is a typical network configuration that can be deployed in a customer's environment. The best practices and setup recommendations are described later in this document.

Deployment Hardware and Software

Configuring Cisco Unified Computing System

It is beyond the scope of this document to cover detailed information about Cisco UCS infrastructure setup and connectivity. The documentation guides and examples are available at http://www.cisco.com/en/US/products/ps10281/products_installation_and_configuration_guides_list.html.



All the tasks to configure Cisco Unified Computing System are detailed in this document, however, only some of the screenshots are included. The following sections detail the high-level steps involved for a Cisco UCS configuration.

Configure/Verify Global Chassis Discovery Policy

The chassis discovery policy determines how the system reacts when you add a new chassis. Cisco UCS Manager uses the settings in the chassis discovery policy to determine the minimum threshold for the number of links between the chassis and the fabric interconnect and whether to group links from the IOM to the fabric interconnect in a fabric port channel. We recommend using the platform max value as shown. Using platform max insures that Cisco UCS Manager uses the maximum number of IOM uplinks available. Please set Link Grouping to Port Channel as shown below.

Figure 17 Discovery Policy

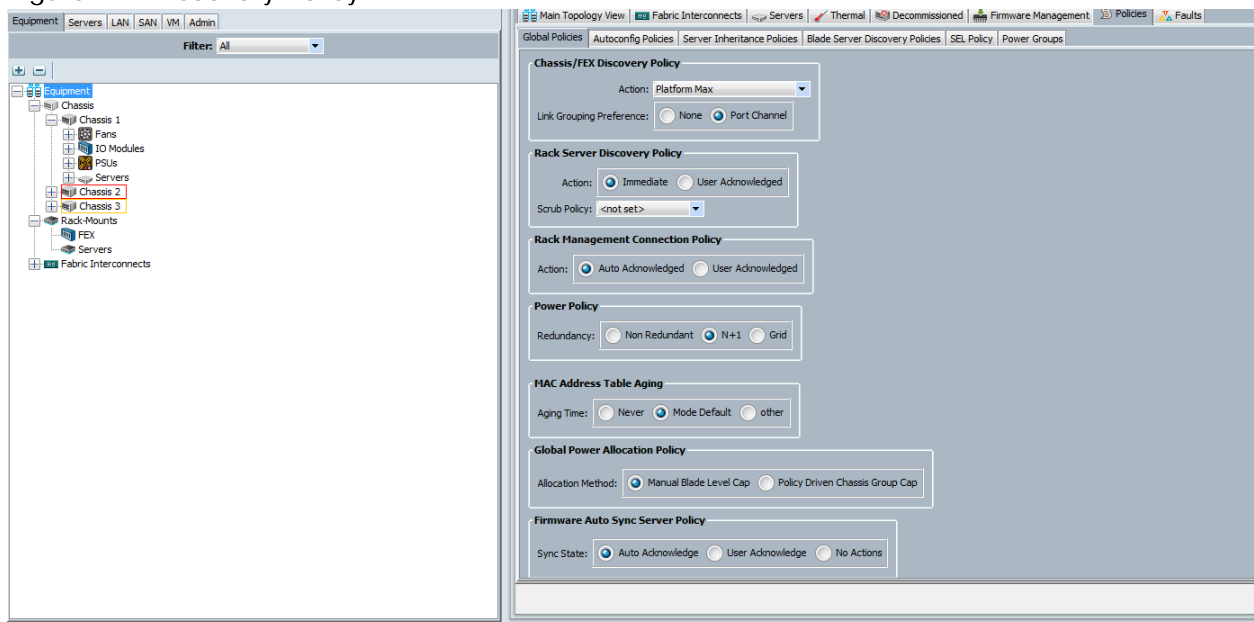
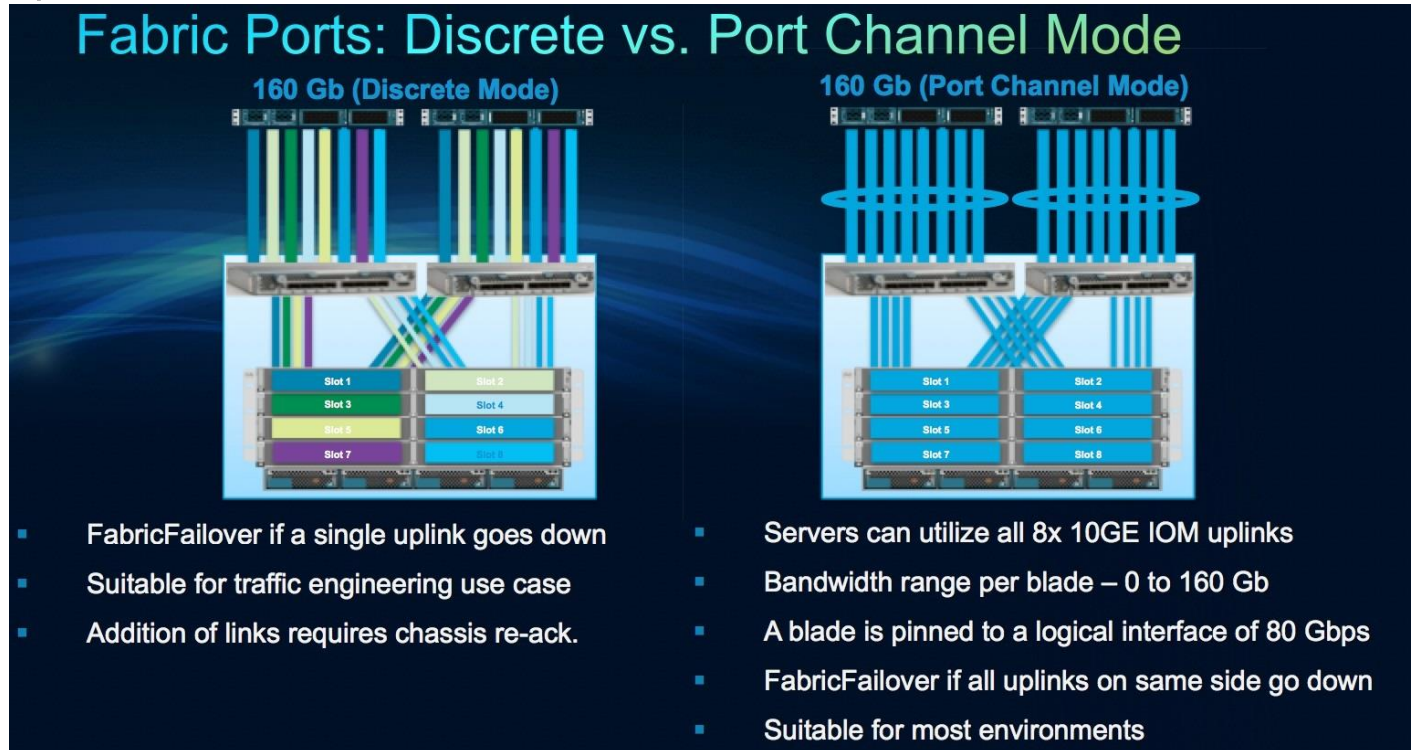


Figure 18 lists some of differences between Discrete vs. Port Channel Link grouping:

Figure 18 Fabric Ports: Discrete vs Port Channel Mode



Configuring Fabric Interconnects for Chassis and Blade Discovery

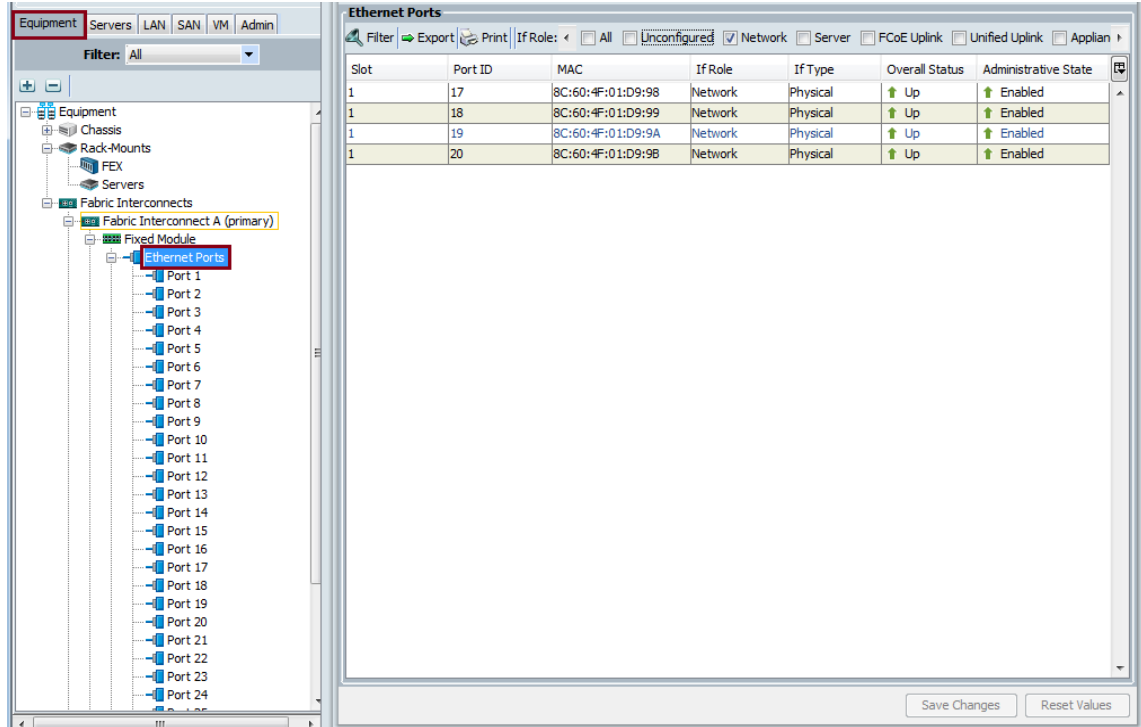
Configure the Server Ports to initiate Chassis and Blade discovery. To configure the server ports, complete the following steps:

1. Go to Equipment->Fabric Interconnects->Fabric Interconnect A->Fixed Module->Ethernet Ports.
2. Select the ports that are connected to the chassis IOM.
3. Right-click and “Configure as Server Port”. Repeat these steps for Fabric Interconnect B.

Configure LAN Specific Tasks

Configure and Enable Ethernet uplink Ports on Fabric Interconnect A and B. We created four uplink ports on each Fabric Interconnect as shown below. As an example we used port 17,18,19 and 20 from both Fabric Interconnect A and Fabric Interconnect B to configure as Ethernet uplink port.

Figure 19 Configuring Ethernet Uplink Port



Create Port-Channels for Both Fabric Interconnect A and B

Create Port-Channel on both Fabric Interconnect A and Fabric Interconnect B using ethernet uplink port 17,18,19 and 20.

Figure 20 Port Channels for Fabric A

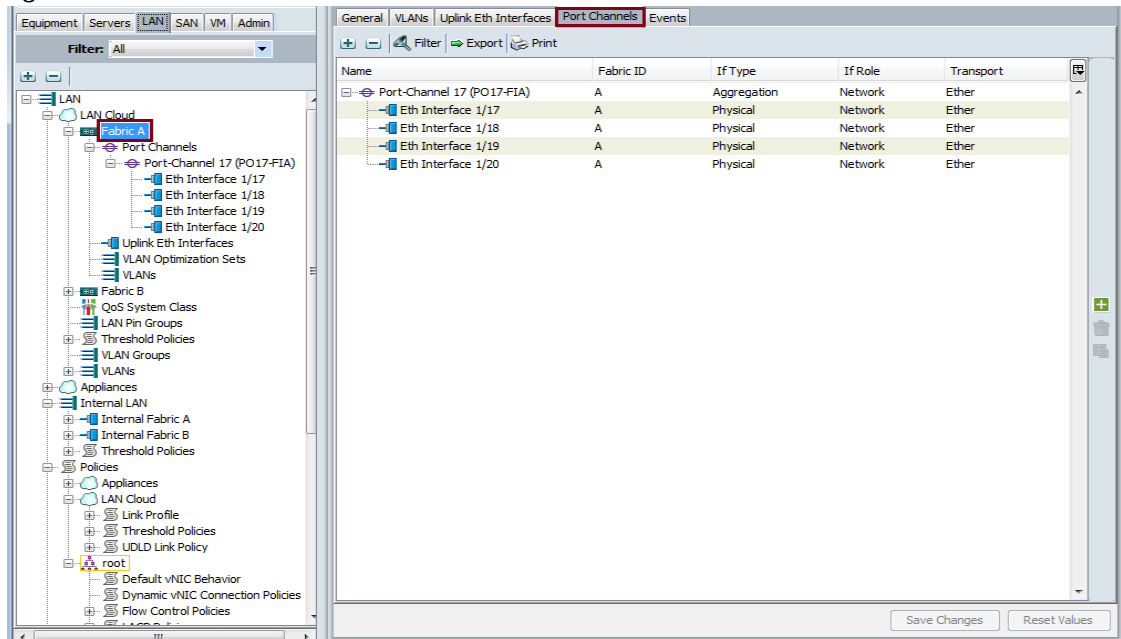
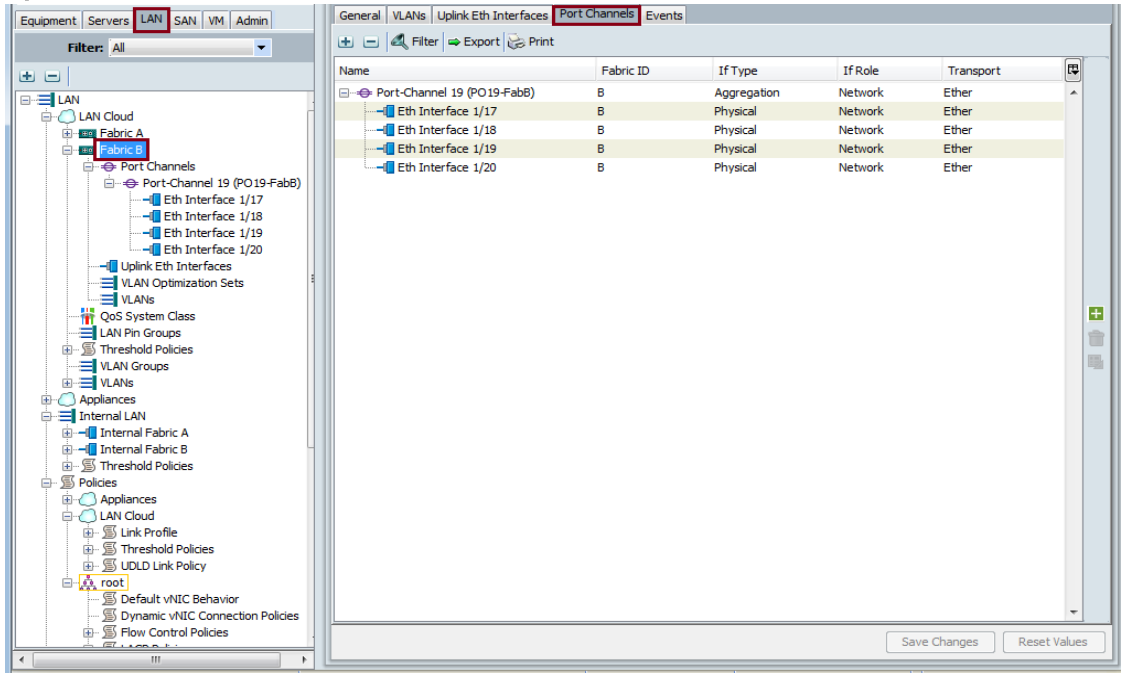


Figure 21 Port Channels for Fabric B



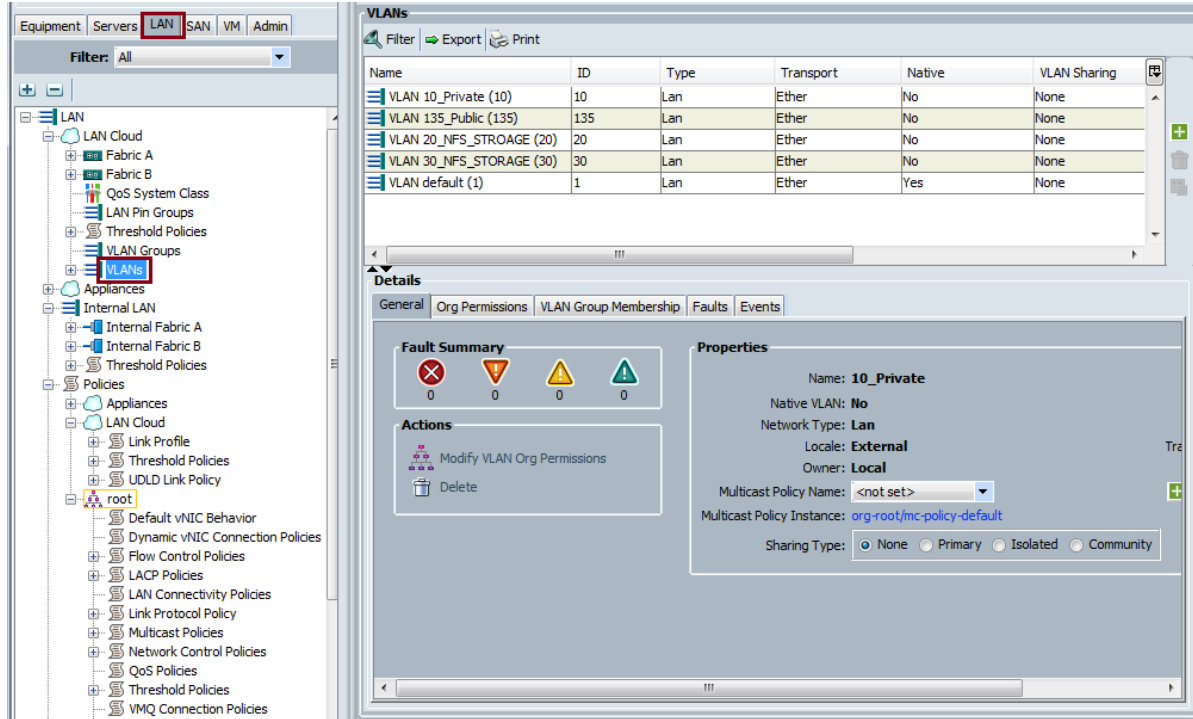
Configure and Create VLANs for Public, Private and Storage Traffic

We created four VLANs as public (VLAN ID 135), private (VLAN ID 10), storageA (VLAN ID 20) and storage (VLAN ID 30) shown below.

Table 5 VLANs

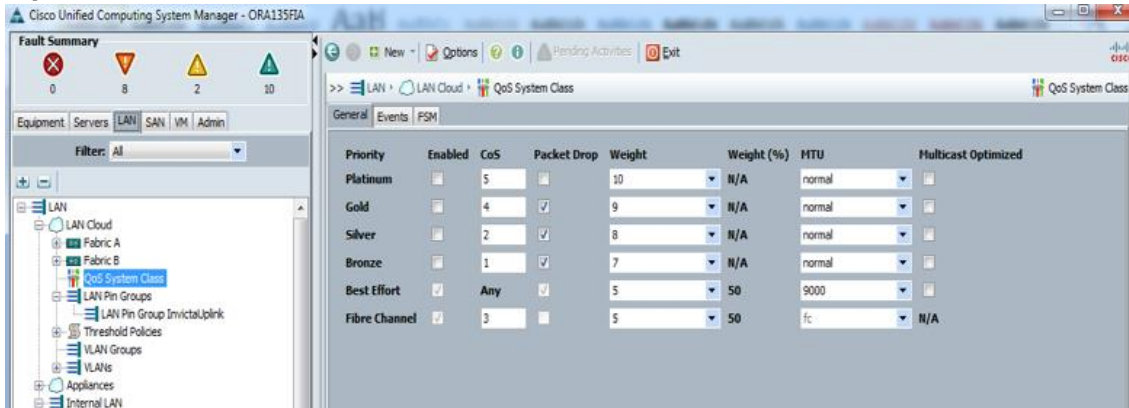
VLAN	VLAN ID	Purpose
Public	135	Public Access
Private	10	Oracle Cluster Interconnect Traffic
StorageA	20	NFS storage access from storage controller A
StorageB	30	NFS storage access from storage controller B

Figure 22 Create VLANs



Enable Jumbo Frames (MTU 9000) on “Best Effort” Class on Fabric Interconnect

Figure 23 Enable Jumbo Frame



Configure UUID and MAC Pools

Figure 24 UUID Pools

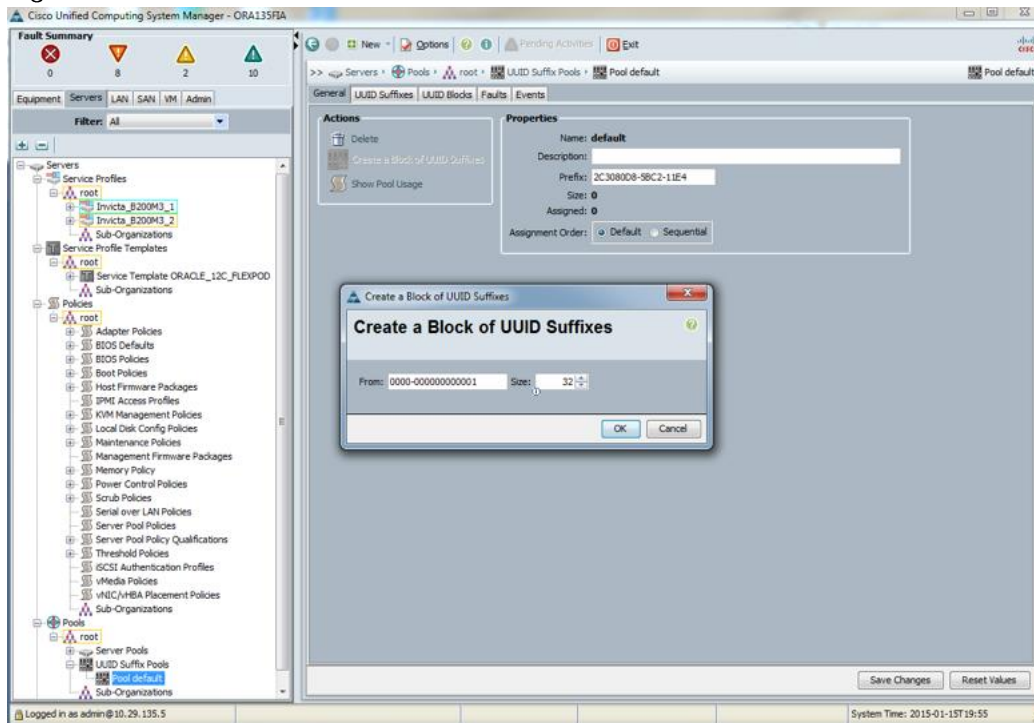
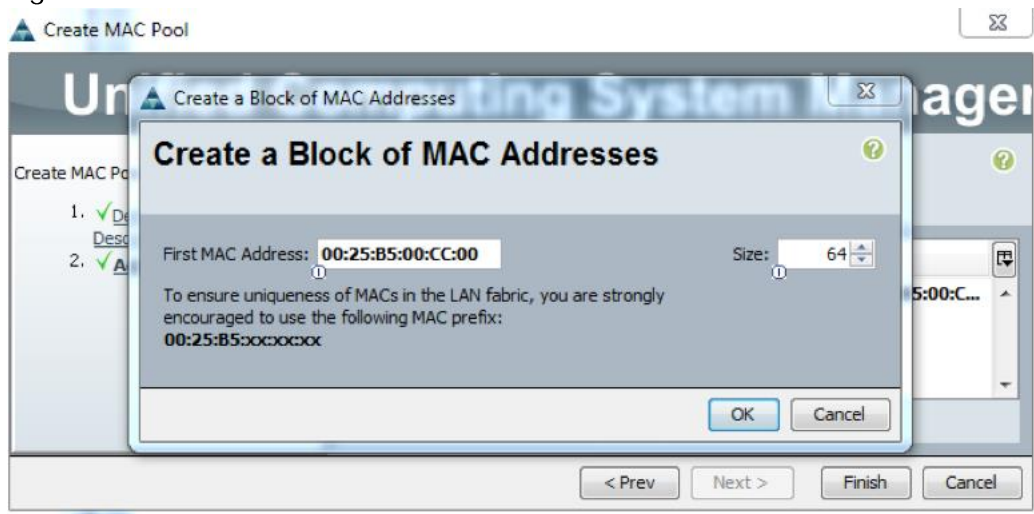


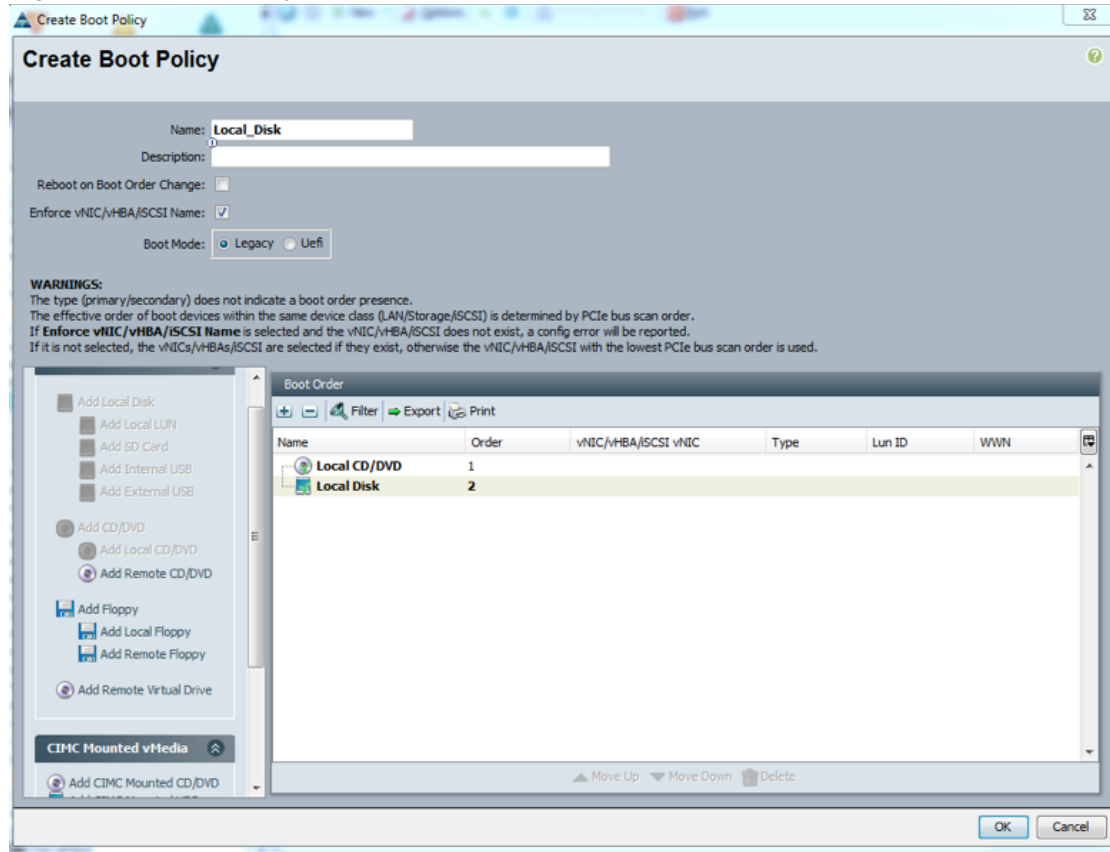
Figure 25 MAC Addresses Pool



Create Server Boot Policy

We strongly recommend to use “Boot from SAN” to realize full benefits of Cisco UCS stateless computing feature such as service profile mobility. However, due to the nature of some of the resiliency tests, we decided to use local disk boot policy for this pools solution. We created a “Local Disk” boot policy as shown below.

Figure 26 Boot Policy



Configure and Create a Service Profile Template

Figure 27 Service Profile Template

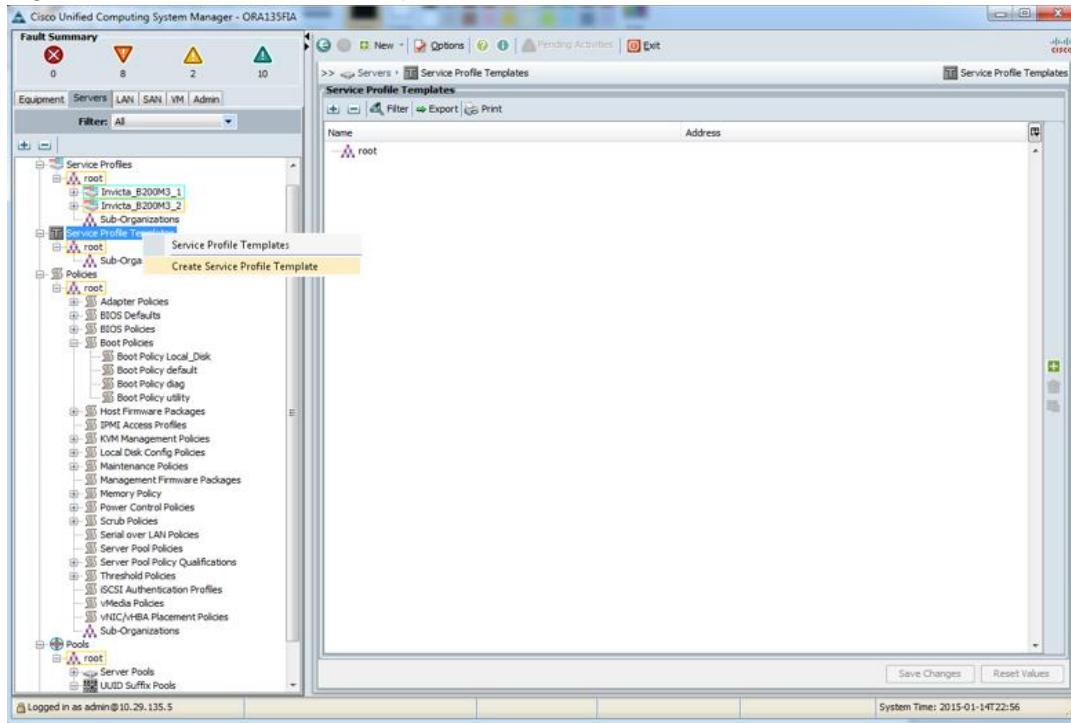


Figure 28 Identify Service Profile Template

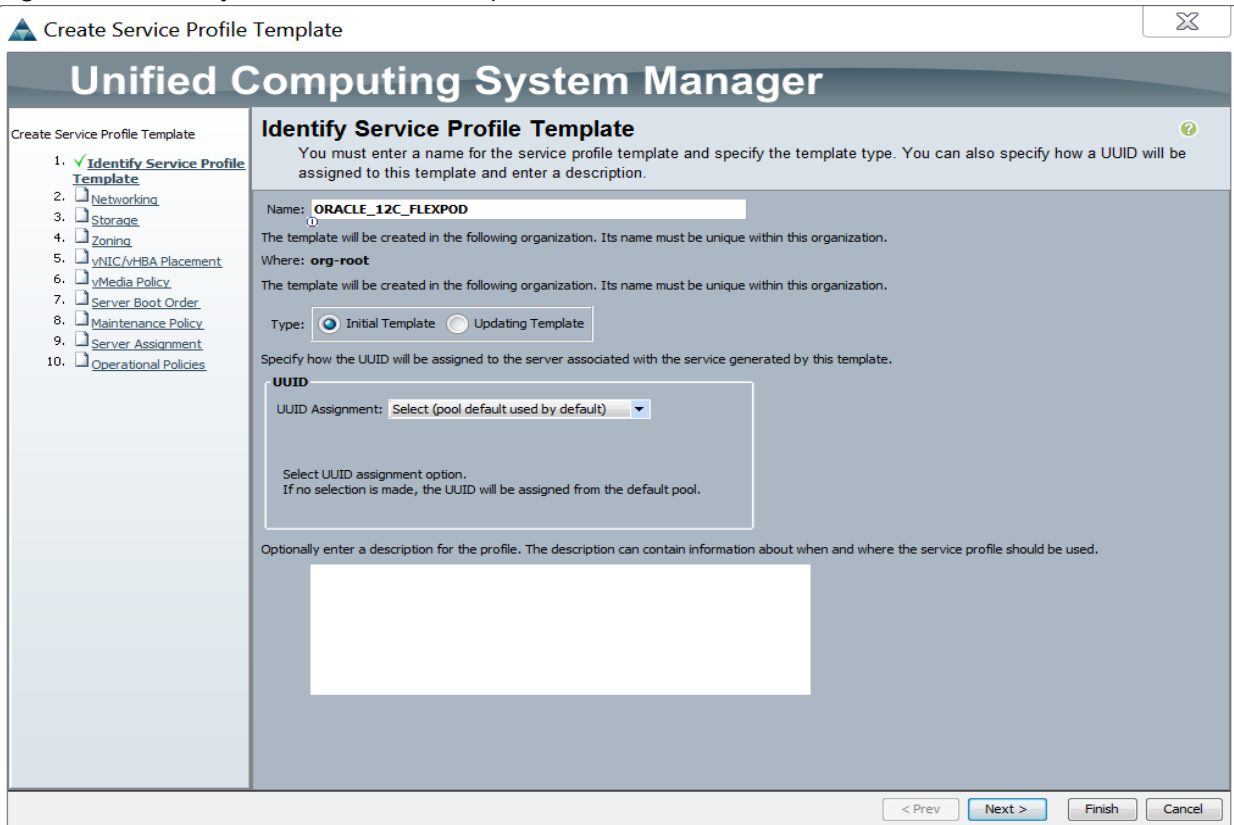


Figure 29 Networking in Service Profile Templates

The screenshot shows the 'Create Service Profile Template' window in the Unified Computing System Manager. The main title is 'Unified Computing System Manager'. On the left, a sidebar lists the steps of the wizard: 1. Identify Service Profile Template (checked), 2. Networking (checked), 3. Storage, 4. Zoning, 5. vNIC/vHBA Placement, 6. vMedia Policy, 7. Server Boot Order, 8. Maintenance Policy, 9. Server Assignment, and 10. Operational Policies.

The main content area is titled 'Networking' and includes the instruction: 'Optionally specify LAN configuration information.' Below this, there is a dropdown menu for 'Dynamic vNIC Connection Policy' with the text 'Select a Policy to use (no Dynamic vNIC Policy by defa...)' and a '+ Create Dynamic vNIC Connection Policy' button.

Underneath, a section asks 'How would you like to configure LAN connectivity?' with four radio button options: 'Simple', 'Expert' (selected), 'No vNICs', and 'Use Connectivity Policy'.

Below the radio buttons, it says 'Click Add to specify one or more vNICs that the server should use to connect to the LAN.' This is followed by a table with the following columns: 'Name', 'MAC Address', 'Fabric ID', and 'Native VLAN'. The table is currently empty.

At the bottom of the table area, there are three buttons: 'Delete', '+ Add', and 'Modify'.

Below the table is a section for 'iSCSI vNICs' with a dropdown arrow.

At the bottom of the window, there are navigation buttons: '< Prev', 'Next >', 'Finish', and 'Cancel'.

Select "Expert" and click "Add" to create vNICs. (Alternatively, you can pre-create vNIC templates for private, public and storage vNICs).

Figure 30 vNIC Public in Service Profile Template

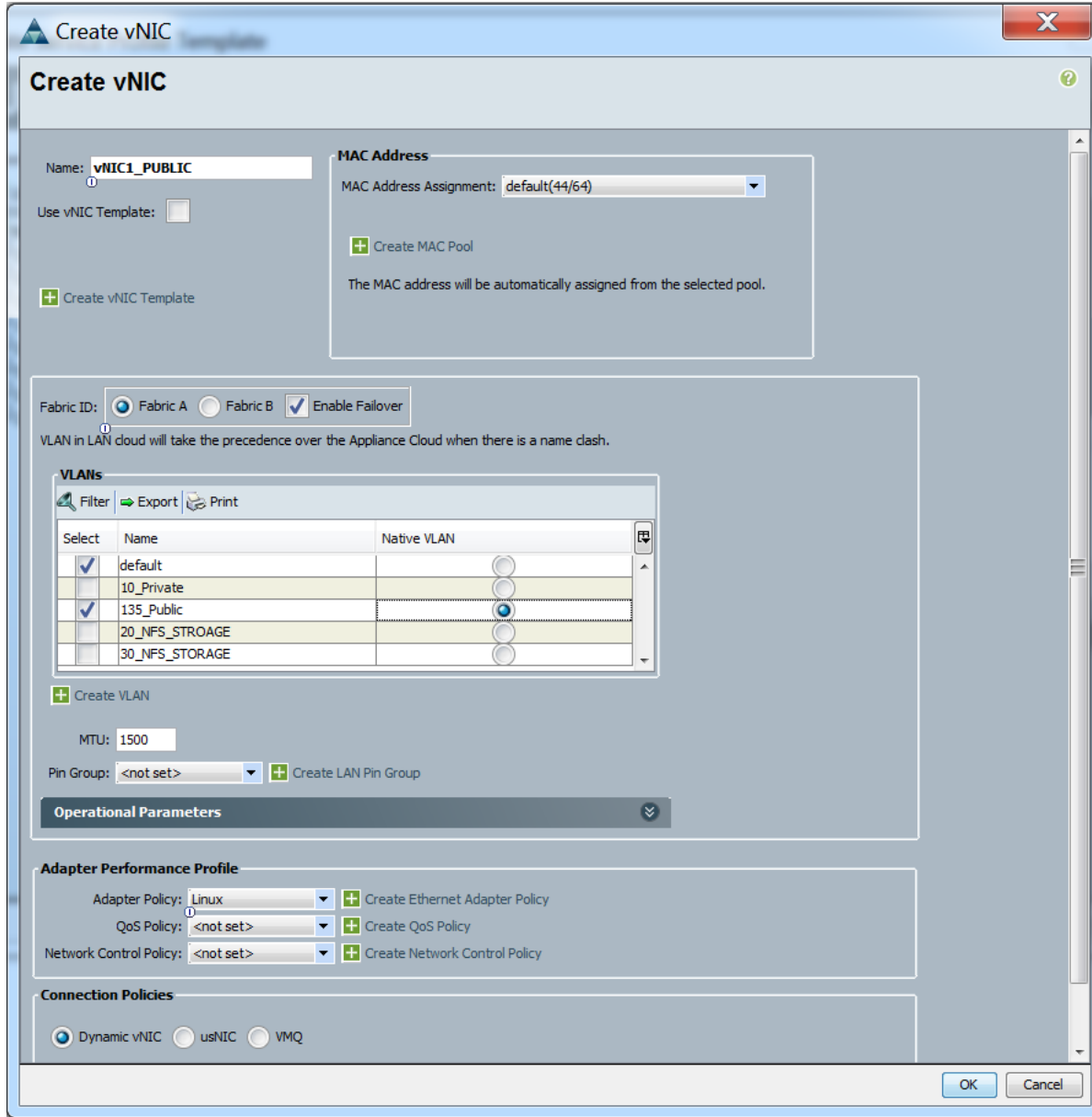


Figure 31 vNIC Private in Service Profile Template

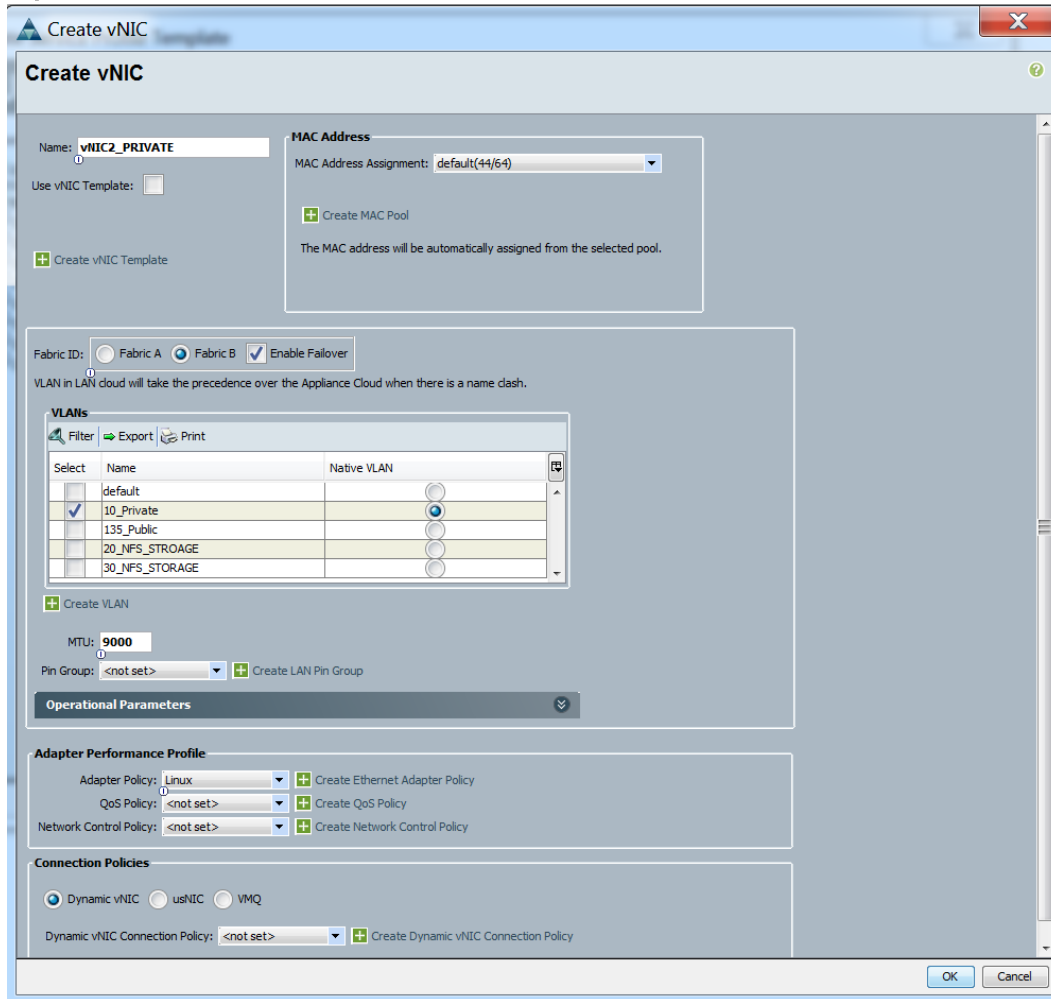


Figure 32 vNIC Storage A in Service Profile Template

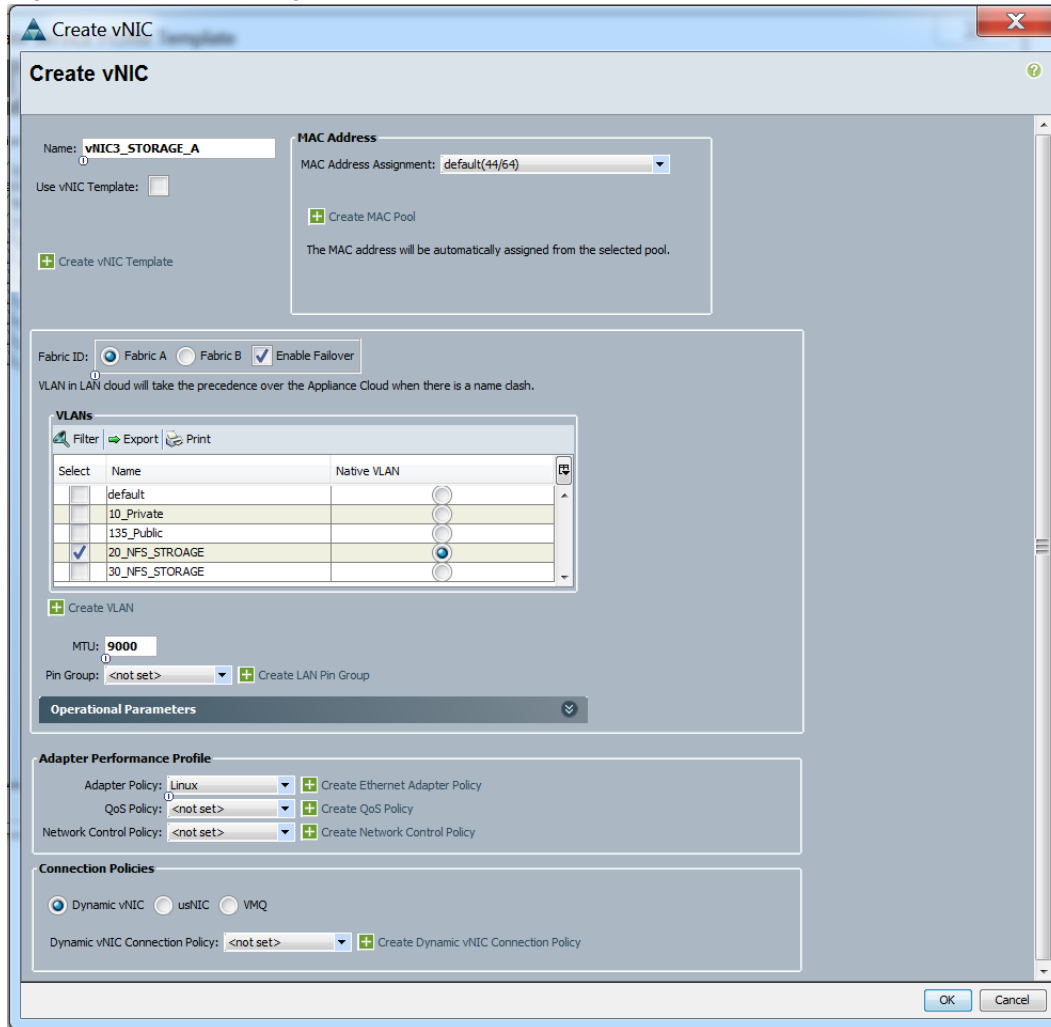


Figure 33 vNIC Storage B in Service Profile Template

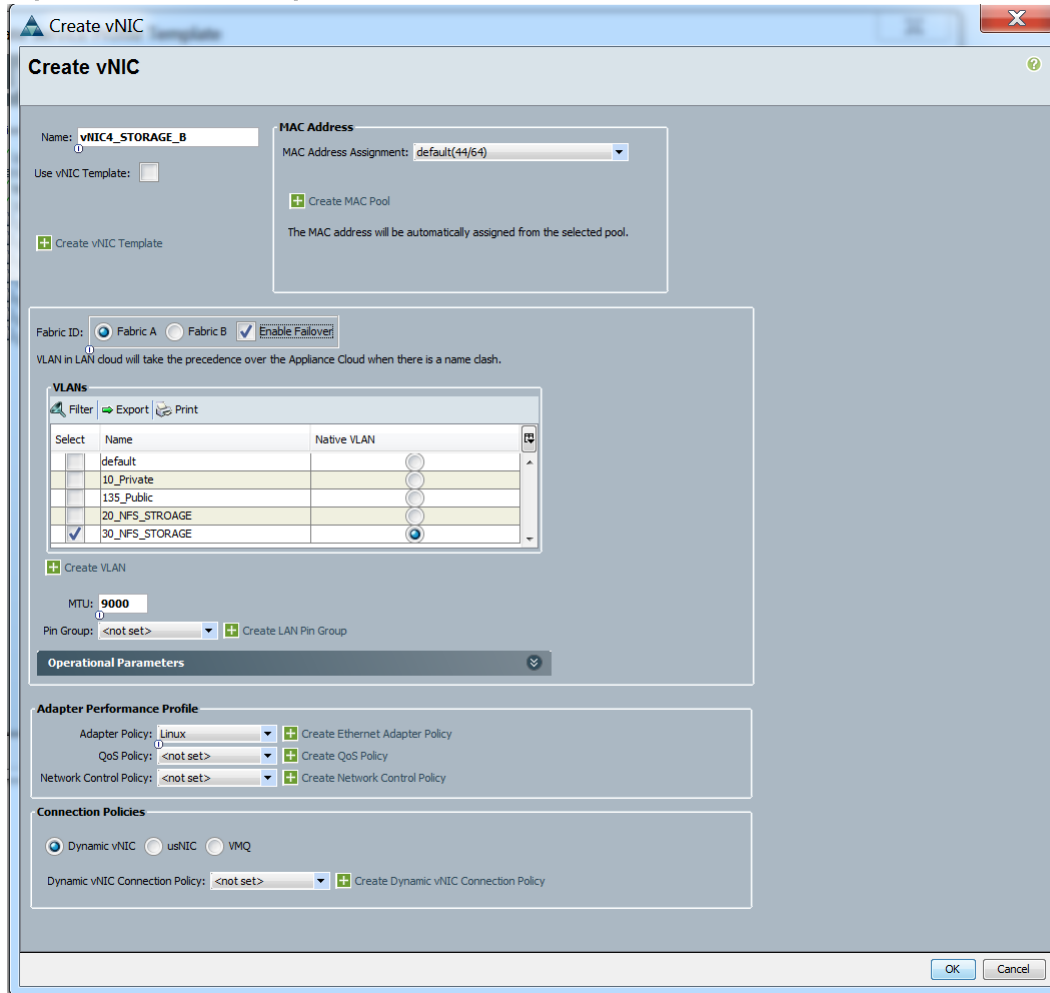


Figure 34 Storage in Service Profile Template



Figure 35 vNICs Placement

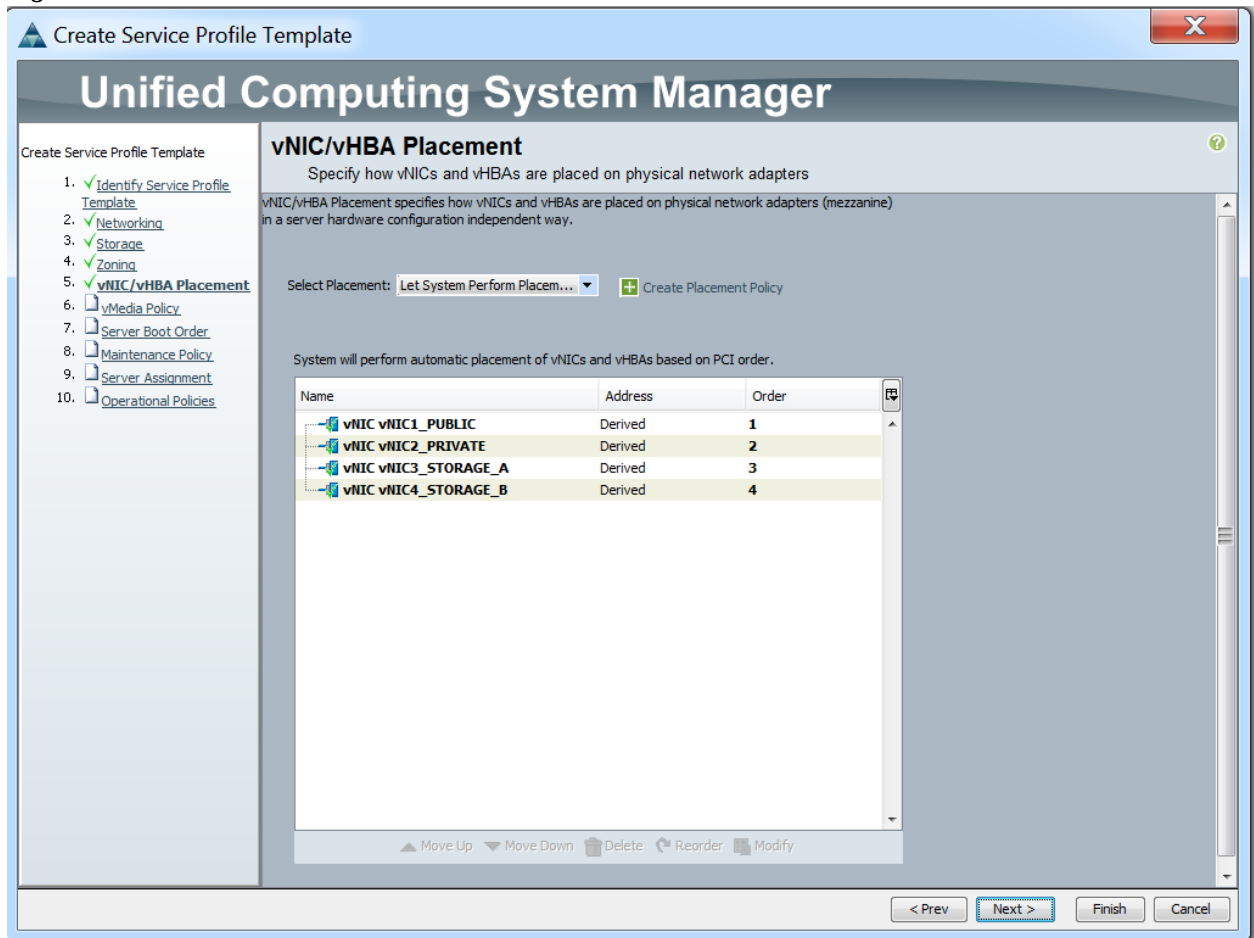


Figure 36 Boot Order in Service Profile Template

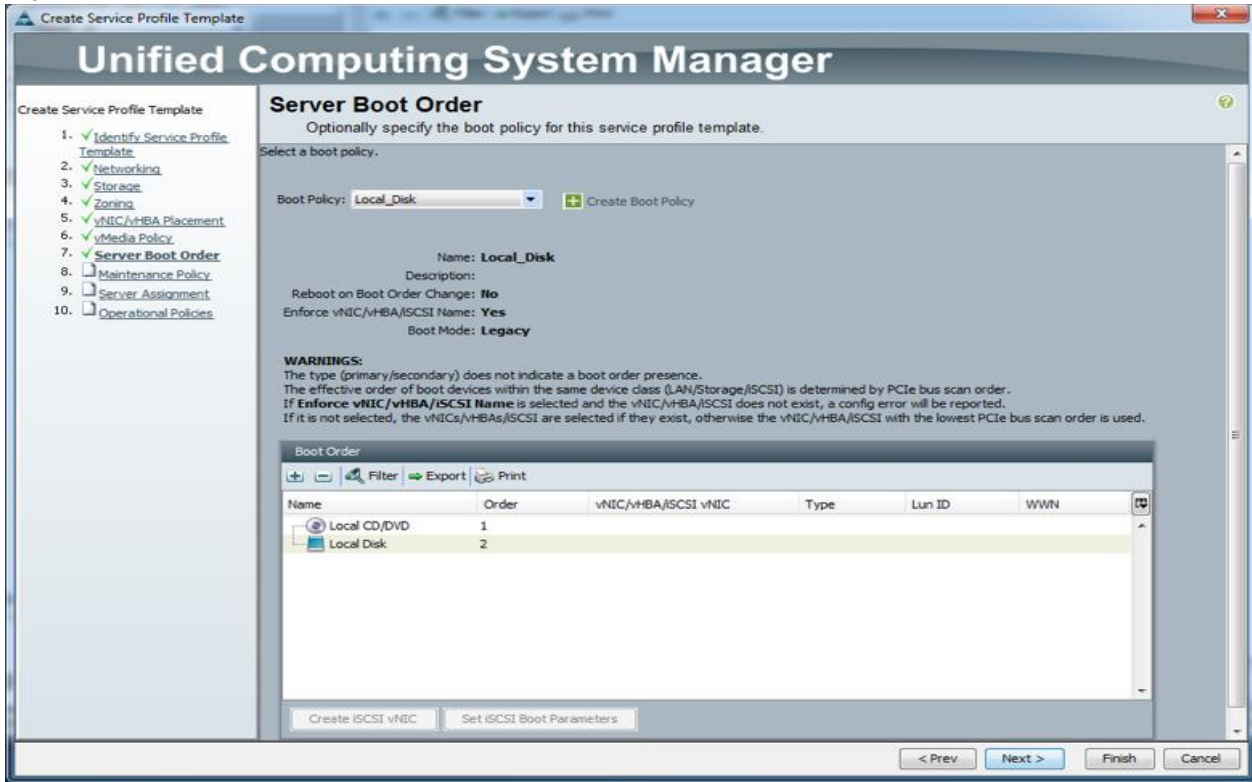
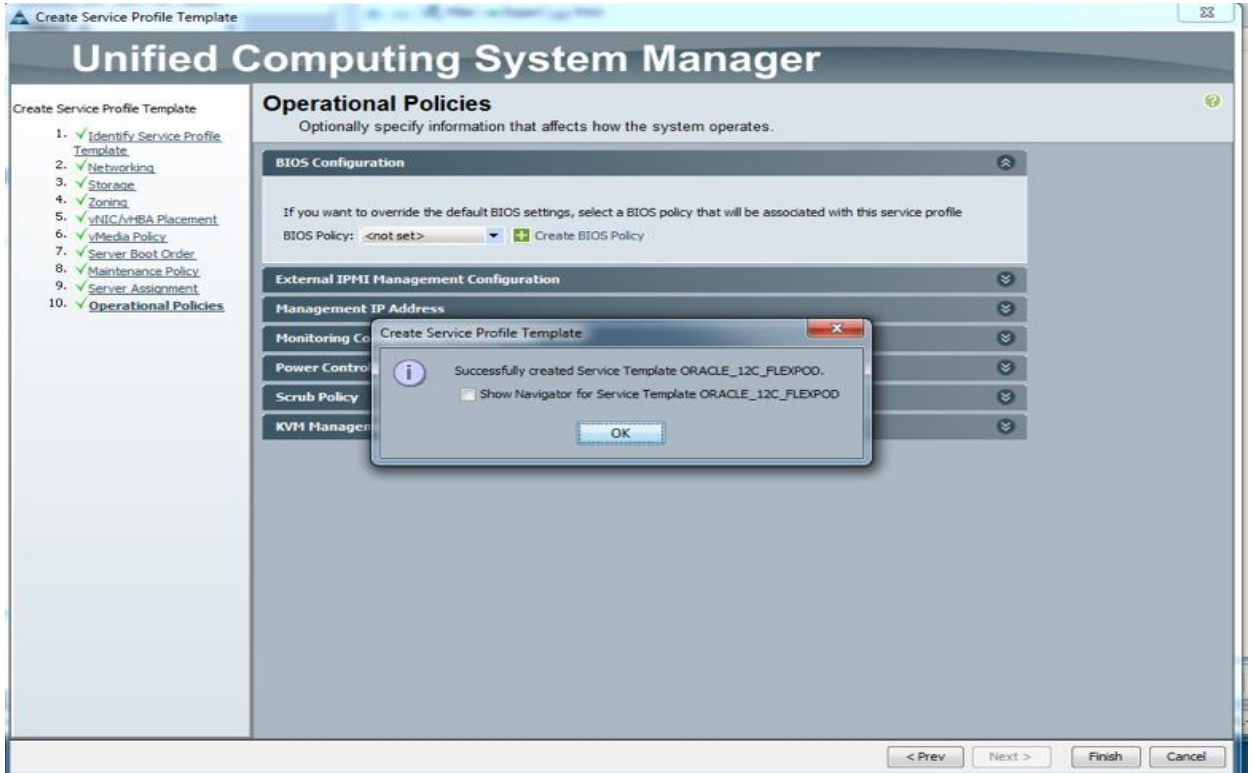
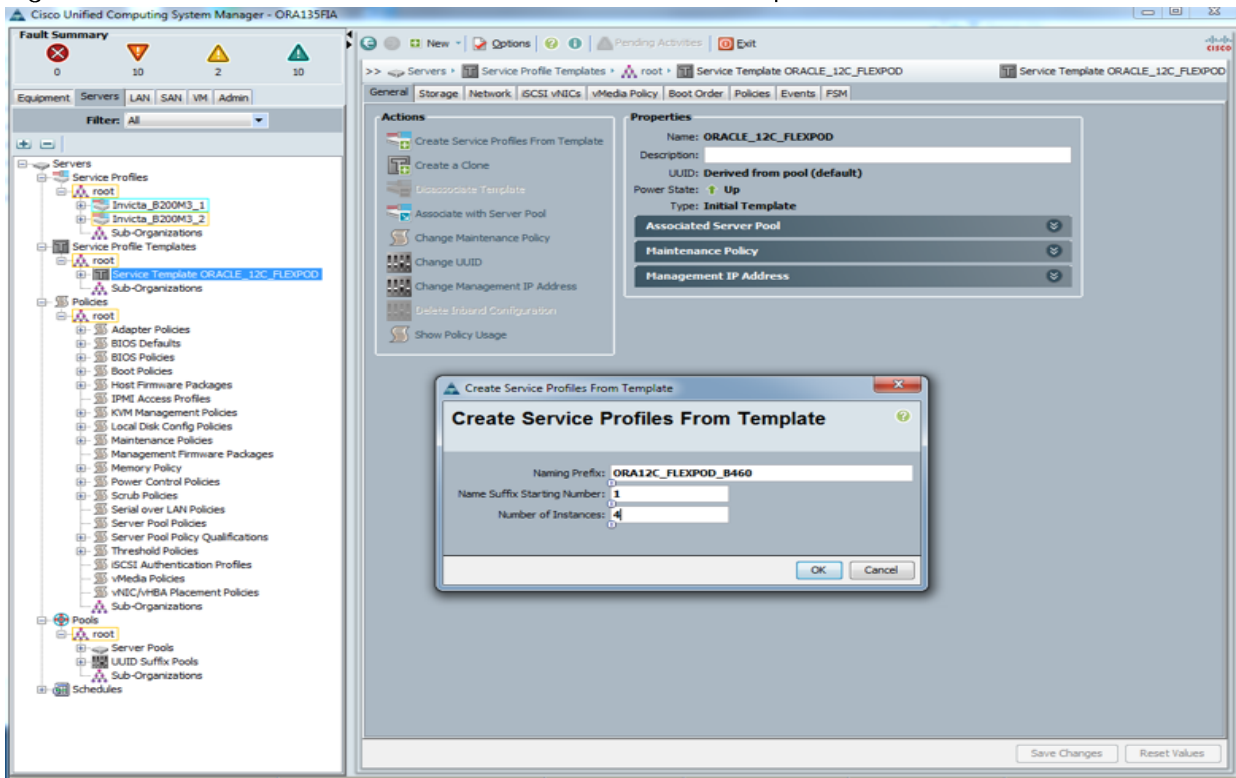


Figure 37 Service Profile Template



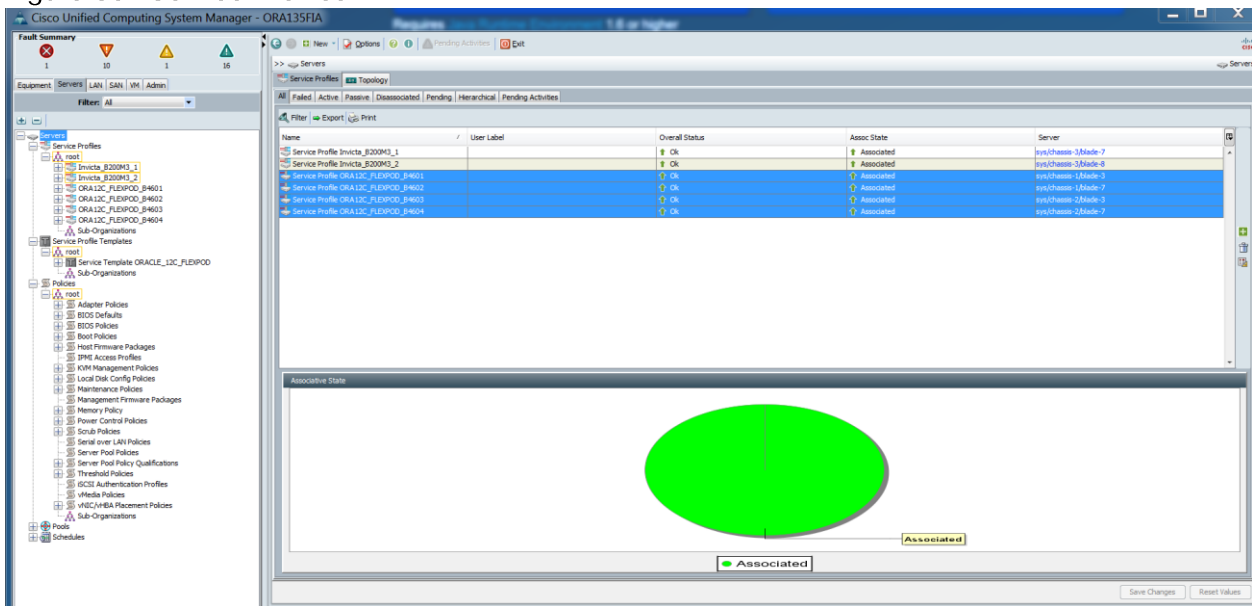
Create Service Profiles from Template and Associate to Servers

Figure 38 Service Profile Creation from Service Profile Template



When the service profiles are created, associate them to servers. Figure 39 shows 4 servers associated with appropriate service profiles.

Figure 39 Service Profiles



This completes the Cisco UCS configuration steps.

Configure Cisco Nexus 9396PX Switches

The following are the steps for Nexus 9396PX switch configuration. The full details of “show run” output is listed in Appendix.

VLAN Configuration

1. Create VLAN 135 for Public traffic

```
N9k-Flexpod-SwitchA# config terminal
N9k-Flexpod-SwitchA(config)# VLAN 135
N9k-Flexpod-SwitchA(config-VLAN)# name Oracle_RAC_Public_Traffic
N9k-Flexpod-SwitchA(config-VLAN)# no shutdown
N9k-Flexpod-SwitchA(config-VLAN)# exit
```

2. Create VLAN 10 for Private traffic

```
N9k-Flexpod-SwitchA(config)# VLAN 10
N9k-Flexpod-SwitchA(config-VLAN)# name Oracle_RAC_Private_Traffic
N9k-Flexpod-SwitchA(config-VLAN)# no ip igmp snooping
N9k-Flexpod-SwitchA(config-VLAN)# no shutdown
N9k-Flexpod-SwitchA(config-VLAN)# exit
```

3. Create VLAN 20 for storage traffic

```
N9k-Flexpod-SwitchA(config)# VLAN 20
N9k-Flexpod-SwitchA(config-VLAN)# name Storage_Traffic for ControllerA
N9k-Flexpod-SwitchA(config-VLAN)# no shutdown
N9k-Flexpod-SwitchA(config-VLAN)# exit
```

4. Create VLAN 30 for storage traffic

```
N9k-Flexpod-SwitchA(config)# VLAN 30
N9k-Flexpod-SwitchA(config-VLAN)# name Storage_Traffic for ControllerB
N9k-Flexpod-SwitchA(config-VLAN)# no shutdown
N9k-Flexpod-SwitchA(config-VLAN)# exit
N9k-Flexpod-SwitchA(config)# copy running-config startup-config
```

Virtual Port Channel (vPC) Summary for Data and Storage Network

Cisco Nexus 9396PX vPC configurations with the vPC domains and corresponding vPC names and IDs for Oracle Database Servers is shown in table below. To provide Layer 2 and Layer 3 switching, a pair of Cisco Nexus 9396PX Switches with upstream switching are deployed, providing HA in the event of failure to Cisco

UCS to handle management, application, and Network storage data traffic. In the Cisco Nexus 9396PX switch topology, a single vPC feature is enabled to provide HA, faster convergence in the event of a failure, and greater throughput.

Table 6 vPC Summary

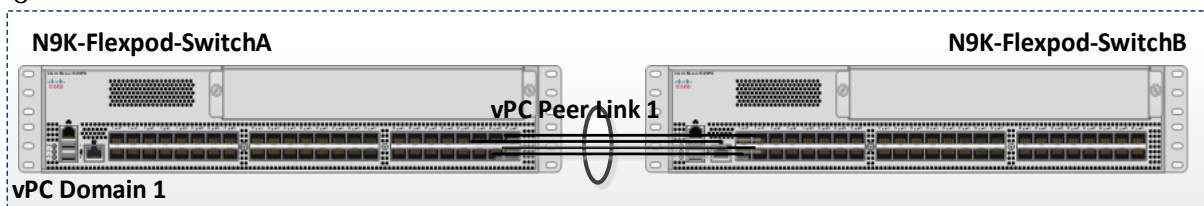
vPC Domain	vPC Name	vPC ID
1	Peer-Link	1
1	vPC Public	17
1	vPC Private	19
1	vPC Storage A	20
1	vPC Storage B	30

As listed in the table above, a single vPC domain with Domain ID 1 is created across two Cisco Nexus 9396PX member switches to define vPC members to carry specific VLAN network traffic. In this topology, we defined a total of 5 vPCs. vPC ID 1 is defined as Peer link communication between two Nexus switches in Fabric A and B. vPC IDs 17 and 19 are defined for public and private traffic from Cisco UCS fabric interconnects. vPC IDs 20 and 30 are defined for NFS Storage traffic to respective NetApp Storage Array controllers. Please follow these steps to create this configuration.

Create vPC Peer-Link Between Two Cisco Nexus Switches

To create a vPC Peer-Link between two Cisco Nexus Switches, complete the following steps:

Figure 40 Cisco Nexus Switch Peer-Link



1. Login Nexus 9396PX-A switch as “admin” user.
2. For vPC 1 as Peer-link, we used interfaces 1-4 for Peer-Link. You may choose the appropriate number of ports for your needs.

```
N9k-Flexpod-SwitchA# config terminal
N9k-Flexpod-SwitchA(config)#feature vpc
N9k-Flexpod-SwitchA(config)#vpc domain 1
N9k-Flexpod-SwitchA(config-vpc-domain)# peer-keepalive destination <Mgmt. IP
Address of peer-N9K-Flexpod-SwitchB>
N9k-Flexpod-SwitchA(config-vpc-domain)# exit
```

```
N9k-Flexpod-SwitchA(config)# interface port-channel 1
N9k-Flexpod-SwitchA(config-if)# switchport mode trunk
N9k-Flexpod-SwitchA(config-if)# vpc peer-link
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed VLAN 1,10,135,20,30
N9k-Flexpod-SwitchA(config-if)# spanning-tree port type network
N9k-Flexpod-SwitchA(config-if)# exit
```

```
N9k-Flexpod-SwitchA(config)# interface Ethernet1/1
N9k-Flexpod-SwitchA(config-if)# description Peer link connected to N9KB-Eth1/1
N9k-Flexpod-SwitchA(config-if)# switchport mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed VLAN 1,10,20,30,135
N9k-Flexpod-SwitchA(config-if)# channel-group 1 mode active
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit
```

```
N9k-Flexpod-SwitchA(config)# interface Ethernet1/2
N9k-Flexpod-SwitchA(config-if)# description Peer link connected to N9KB-Eth1/2
N9k-Flexpod-SwitchA(config-if)# switchport mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed VLAN 1,10,20,30,135
N9k-Flexpod-SwitchA(config-if)# channel-group 1 mode active
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit
```

```
N9k-Flexpod-SwitchA(config)# interface Ethernet1/3
N9k-Flexpod-SwitchA(config-if)# description Peer link connected to N9KB-Eth1/3
N9k-Flexpod-SwitchA(config-if)# switchport mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed VLAN 1,10,20,30,135
N9k-Flexpod-SwitchA(config-if)# channel-group 1 mode active
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit
```

```
N9k-Flexpod-SwitchA(config)# interface Ethernet1/4
```

```
N9k-Flexpod-SwitchA(config-if)# description Peer link connected to N9KB-Eth1/4
N9k-Flexpod-SwitchA(config-if)# switchport mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed VLAN 1,10,20,30,135
N9k-Flexpod-SwitchA(config-if)# channel-group 1 mode active
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit
```

```
N9k-Flexpod-SwitchA(config)# copy running-config startup-config
```

3. Login Nexus 9396PX-B switch as “admin” user and repeat steps to complete second switch configuration.

Create vPC Configuration Between Cisco Nexus 9396PX and Fabric Interconnects

Create and configure vPC 17 and 19 for Data network between Nexus 9396PX switches and Fabric Interconnects.

Figure 41 Configuration between Nexus Switch and Fabric Interconnects

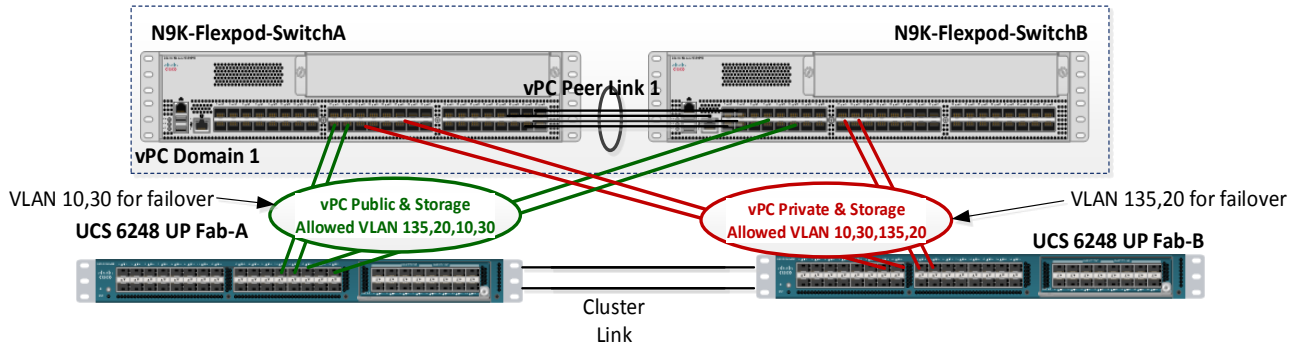


Table 7 summarizes vPC IDs, allowed VLAN IDs and Ethernet uplink ports.

Table 7 VLAN IDs

vPC Description	vPC ID Nexus 9396PX	Fabric Interconnect uplink ports	Nexus 9396PX ports	Allowed VLANs
Port Channel FI-A	17	FI-A P17	N9KA P17	135,20,10,30
		FI-A P18	N9KA P18	Note: VLAN 10 and 30 needed to for failover
		FI-A P19	N9KB P17	
		FI-A P20	N9KB P18	
Port Channel FI-B	19	FI-B P17	N9KA P19	10,30,135,20
		FI-B P18	N9KA P20	Note: VLAN 135 and

vPC Description	vPC ID Nexus 9396PX	Fabric Interconnect uplink ports	Nexus 9396PX ports	Allowed VLANs
		FI-B P19	N9KB P19	20 needed for failover
		FI-B P20	N9KB P20	

The following are the configuration details for Nexus 9396PX.

1. Log in to Nexus 9396PX-A switch as “admin” user and complete the following steps:

```
N9k-Flexpod-SwitchA# config Terminal
N9k-Flexpod-SwitchA(config)# interface Port-channel 17
N9k-Flexpod-SwitchA(config-if)# description Port-Channel FI-A
N9k-Flexpod-SwitchA(config-if)# switchport mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed VLAN 135,10,20,30
N9k-Flexpod-SwitchA(config-if)# spanning-tree port type edge trunk
N9k-Flexpod-SwitchA(config-if)# mtu 9216
N9k-Flexpod-SwitchA(config-if)# vPC 17
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit

N9k-Flexpod-SwitchA(config)# interface Port-channel 19
N9k-Flexpod-SwitchA(config-if)# description Port-Channel FI-B
N9k-Flexpod-SwitchA(config-if)# switchport mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed VLAN 135,10,20,30
N9k-Flexpod-SwitchA(config-if)# spanning-tree port type edge trunk
N9k-Flexpod-SwitchA(config-if)# mtu 9216
N9k-Flexpod-SwitchA(config-if)# vPC 19
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit

N9k-Flexpod-SwitchA# config Terminal
N9k-Flexpod-SwitchA(config)# interface eth 1/17
N9k-Flexpod-SwitchA(config-if)# description Connection FI-A P17 to N9KA P17
N9k-Flexpod-SwitchA(config-if)# switch mode trunk
```



```
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed vlan 135,10,20,30
N9k-Flexpod-SwitchA(config-if)# spanning-tree port type edge trunk
N9k-Flexpod-SwitchA(config-if)# mtu 9216
N9k-Flexpod-SwitchA(config-if)# channel-group 17 mode active
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit
```

```
N9k-Flexpod-SwitchA(config)# interface eth 1/18
N9k-Flexpod-SwitchA(config-if)# description Connection FI-A P18 to N9KA P18
N9k-Flexpod-SwitchA(config-if)# switch mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed vlan 135,10,20,30
N9k-Flexpod-SwitchA(config-if)# spanning-tree port type edge trunk
N9k-Flexpod-SwitchA(config-if)# mtu 9216
N9k-Flexpod-SwitchA(config-if)# channel-group 17 mode active
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit
```

```
N9k-Flexpod-SwitchA(config)# interface eth 1/19
N9k-Flexpod-SwitchA(config-if)# description Connection FI-B P17 to N9KA P19
N9k-Flexpod-SwitchA(config-if)# switch mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed vlan 135,10,20,30
N9k-Flexpod-SwitchA(config-if)# spanning-tree port type edge trunk
N9k-Flexpod-SwitchA(config-if)# mtu 9216
N9k-Flexpod-SwitchA(config-if)# channel-group 19 mode active
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit
```

```
N9k-Flexpod-SwitchA(config)# interface eth 1/20
N9k-Flexpod-SwitchA(config-if)# description Connection FI-B P18 to N9KA P20
N9k-Flexpod-SwitchA(config-if)# switch mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed vlan 135,10,20,30
N9k-Flexpod-SwitchA(config-if)# spanning-tree port type edge trunk
```

```
N9k-Flexpod-SwitchA(config-if)# mtu 9216
N9k-Flexpod-SwitchA(config-if)# channel-group 19 mode active
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit
This completes Switch A side configuration.
```

2. Log in to Nexus 9396PX-B switch as “admin” user and repeat steps to complete second switch configuration.

Create vPC Configuration Between Cisco Nexus 9396PX and NetApp NFS Storage

Create and configure vPC 20 and 30 for data network between Storage Cisco Nexus 9396PX switches and NetApp storage.

Figure 42 Configuration between Cisco Nexus Switch and NetApp Storage

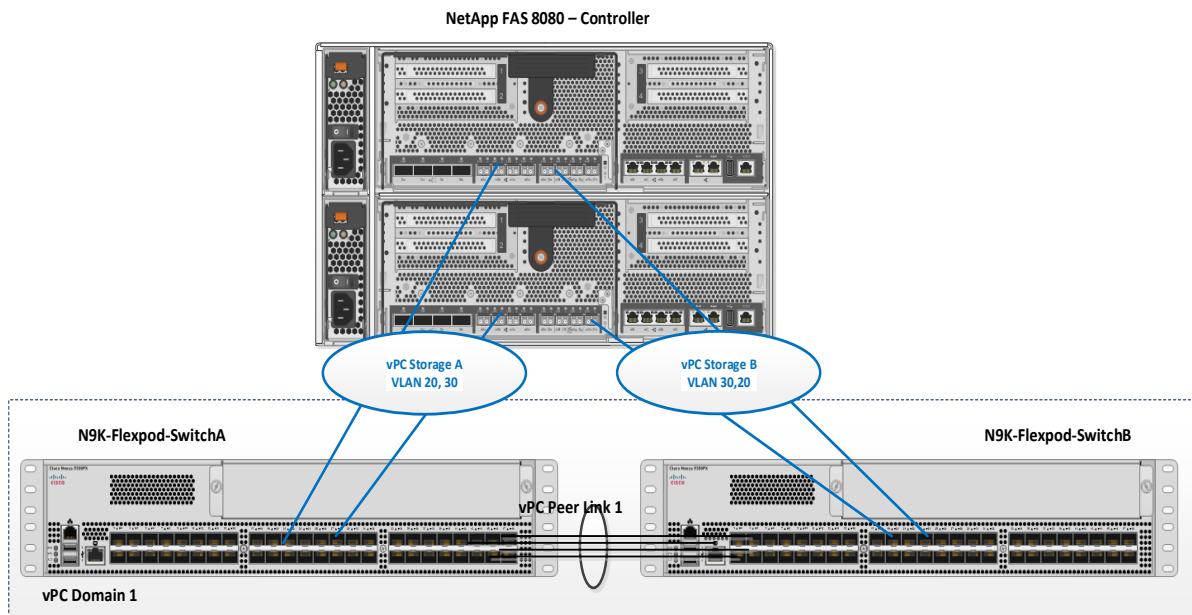


Table 8 summarizes vPC IDs, allowed VLAN IDs and NetApp storage ports.

Table 8 VLAN IDs

vPC Description	vPC ID Nexus 9396PX	NetApp Controller and Storage Port	Nexus 9396PX ports	Allowed VLANs
vPC storage A	20	Controller A port e0e	N9KA P31	20,30
		Controller B port e0e	N9KA P32	
vPC storage B	30	Controller A port e0g	N9KB P31	30,20
		Controller B port e0g	N9KB P32	

The following are the configuration details for Nexus 9396PX.

1. Log in to Cisco Nexus 9396PX-A switch as “admin” user and complete the following steps:

```
N9k-Flexpod-SwitchA# configure terminal
N9k-Flexpod-SwitchA(config)# interface Port-channel20
N9k-Flexpod-SwitchA(config-if)# description Port-channel multimode VIF from
FAS8080-CNTRL-A-10G
N9k-Flexpod-SwitchA(config-if)# switchport mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed VLAN 20,30
N9k-Flexpod-SwitchA(config-if)# spanning-tree port type edge trunk
N9k-Flexpod-SwitchA(config-if)# mtu 9216
N9k-Flexpod-SwitchA(config-if)# vPC 20
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit
```

```
N9k-Flexpod-SwitchA# configure terminal
N9k-Flexpod-SwitchA(config)# interface Port-channel30
N9k-Flexpod-SwitchA(config-if)# description Port-channel multimode VIF from
FAS8080-CNTRL-B-10G
N9k-Flexpod-SwitchA(config-if)# switchport mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed VLAN 20,30
N9k-Flexpod-SwitchA(config-if)# spanning-tree port type edge trunk
N9k-Flexpod-SwitchA(config-if)# mtu 9216
N9k-Flexpod-SwitchA(config-if)# vPC 30
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit
```

```
N9k-Flexpod-SwitchA# configure terminal
N9k-Flexpod-SwitchA(config)# interface Ethernet1/31
N9k-Flexpod-SwitchA(config-if)# description Connection to NetApp CTRL-A e0e
N9k-Flexpod-SwitchA(config-if)# switchport mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed vlan 20,30
```

```

N9k-Flexpod-SwitchA(config-if)# spanning-tree port type edge trunk
N9k-Flexpod-SwitchA(config-if)# mtu 9216
N9k-Flexpod-SwitchA(config-if)# channel-group 20 mode active
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit

N9k-Flexpod-SwitchA# configure terminal
N9k-Flexpod-SwitchA(config)# interface Ethernet1/32
N9k-Flexpod-SwitchA(config-if)# description Connection to NetApp CTRL-B e0e
N9k-Flexpod-SwitchA(config-if)# switchport mode trunk
N9k-Flexpod-SwitchA(config-if)# switchport trunk allowed vlan 20,30
N9k-Flexpod-SwitchA(config-if)# spanning-tree port type edge trunk
N9k-Flexpod-SwitchA(config-if)# mtu 9216
N9k-Flexpod-SwitchA(config-if)# channel-group 30 mode active
N9k-Flexpod-SwitchA(config-if)# no shutdown
N9k-Flexpod-SwitchA(config-if)# exit

```

2. Log in to Cisco Nexus 9396PX-B switch as “admin” user and repeat steps to complete second switch configuration.

Setup Jumbo Frames of Cisco Nexus 9396PX Switches

Jumbo frames with an MTU=9216 have to be setup on both Cisco Nexus 9396PX switches. This is essential for both Oracle NFS storage and Oracle Private Interconnect traffic. Please note that Oracle private interconnect traffic does not leave the Cisco UCS domain (Fabric Interconnect) in normal operating conditions. However, if there is IOM failure or Fabric Interconnect reboot, the private interconnect traffic will need to be routed to the immediate northbound switch (Cisco Nexus 9396PX in our case). Since we are using Jumbo Frames as a best practice for Oracle private interconnect, we need to have jumbo frames configured on the Cisco Nexus 9396PX switches.

To enable jumbo frames on the Cisco Nexus 9396PX switch A, complete the following steps:

```

N9k-Flexpod-SwitchA# config terminal
N9k-Flexpod-SwitchA(config)# policy-map type network-qos jumbo
N9k-Flexpod-SwitchA(config-pmap-nq)# class type network-qos class-default
N9k-Flexpod-SwitchA(config-pmap-nq-c)# mtu 9216
N9k-Flexpod-SwitchA(config-pmap-nq-c)# exit
N9k-Flexpod-SwitchA(config-pmap-nq)# system qos
N9k-Flexpod-SwitchA(config-sys-qos)# service-policy type network-qos jumbo

```

```

N9k-Flexpod-SwitchA(config-sys-qos)# exit
N9k-Flexpod-SwitchA(config)# copy running-config startup-config
[#####] 100%
N9k-Flexpod-SwitchA(config)#

```

Log in to Cisco Nexus 9396PX-B switch as “admin” user and repeat steps to complete second switch configuration.

Verify All vPC Status is up on Both Cisco Nexus 9396PX Switches

Figure 43 Cisco Nexus Switch A Status

```

N9k-Flexpod-SwitchA#
N9k-Flexpod-SwitchA#
N9k-Flexpod-SwitchA# show port-channel summary
Flags:  D - Down          P - Up in port-channel (members)
        I - Individual    H - Hot-standby (LACP only)
        s - Suspended     r - Module-removed
        S - Switched      R - Routed
        U - Up (port-channel)
        p - Up in delay-lacp mode (member)
        M - Not in use. Min-links not met
-----
Group Port-      Type      Protocol  Member Ports
  Chan-
nel
-----
 1   Po1(SU)     Eth       LACP      Eth1/1(P)  Eth1/2(P)  Eth1/3(P)
                Eth1/4(P)
17   Po17(SU)    Eth       LACP      Eth1/17(P) Eth1/18(P)
19   Po19(SU)    Eth       LACP      Eth1/19(P) Eth1/20(P)
20   Po20(SU)    Eth       LACP      Eth1/31(P)
30   Po30(SU)    Eth       LACP      Eth1/32(P)
N9k-Flexpod-SwitchA# █

```

Figure 44 Cisco Nexus Switch B Status

```

N9k-Flexpod-SwitchB#
N9k-Flexpod-SwitchB# show port-channel summary
Flags:  D - Down          P - Up in port-channel (members)
        I - Individual    H - Hot-standby (LACP only)
        s - Suspended     r - Module-removed
        S - Switched      R - Routed
        U - Up (port-channel)
        p - Up in delay-lacp mode (member)
        M - Not in use. Min-links not met
-----
Group Port-      Type      Protocol  Member Ports
  Chan-
nel
-----
 1   Po1(SU)     Eth       LACP      Eth1/1(P)  Eth1/2(P)  Eth1/3(P)
                Eth1/4(P)
17   Po17(SU)    Eth       LACP      Eth1/17(P) Eth1/18(P)
19   Po19(SU)    Eth       LACP      Eth1/19(P) Eth1/20(P)
20   Po20(SU)    Eth       LACP      Eth1/31(P)
30   Po30(SU)    Eth       LACP      Eth1/32(P)
N9k-Flexpod-SwitchB# █

```

Figure 45 vPC Description for Switch A

```

N9k-Flexpod-SwitchA#
N9k-Flexpod-SwitchA# show vpc brief
Legend:
          (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id           : 1
Peer status             : peer adjacency formed ok
vPC keep-alive status   : peer is alive
Configuration consistency status : success
Per-vlan consistency status : success
Type-2 consistency status : success
vPC role                 : primary, operational secondary
Number of vPCs configured : 4
Peer Gateway            : Disabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled
Auto-recovery status    : Enabled, timer is off.(timeout = 240s)
Delay-restore status    : Timer is off.(timeout = 30s)
Delay-restore SVI status : Timer is off.(timeout = 10s)

vPC Peer-link status
-----
id   Port   Status Active vlans
--   -
1    Po1    up     1,10,20,30,135

vPC status
-----
id   Port   Status Consistency Reason           Active vlans
--   -
17   Po17   up     success    success           1,10,20,30,135
19   Po19   up     success    success           1,10,20,30,135
20   Po20   up     success    success           20,30
30   Po30   up     success    success           20,30

```

Figure 46 vPC Description for Switch B

```

N9k-Flexpod-SwitchB#
N9k-Flexpod-SwitchB# show vpc brief
Legend:
      (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id           : 1
Peer status              : peer adjacency formed ok
vPC keep-alive status    : peer is alive
Configuration consistency status : success
Per-vlan consistency status : success
Type-2 consistency status : success
vPC role                 : secondary, operational primary
Number of vPCs configured : 4
Peer Gateway             : Disabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled
Auto-recovery status     : Disabled
Delay-restore status     : Timer is off.(timeout = 30s)
Delay-restore SVI status : Timer is off.(timeout = 10s)

vPC Peer-link status
-----
id   Port   Status Active vlans
--   -
1    Po1    up    1,10,20,30,135

vPC status
-----
id   Port   Status Consistency Reason           Active vlans
--   -
17   Po17   up    success    success           1,10,20,30,
135
19   Po19   up    success    success           1,10,20,30,
135
20   Po20   up    success    success           20,30
30   Po30   up    success    success           20,30

N9k-Flexpod-SwitchB# █

```

This completes the Cisco Nexus 9396PX switch configuration. The next step is to configure the NetApp storage.

Configuring NetApp Storage

Storage connectivity and infrastructure configuration is not covered in this document. For more information, refer to the following Cisco site:

<http://www.cisco.com/c/en/us/solutions/enterprise/design-zone-data-centers/index.html>

This section describes the storage layout and design considerations for the database deployment. Figure 47 through Figure 50 illustrate the SVM (formerly known as Vserver) and LIF setup configurations. In the examples shown in Figure 47 and Figure 48, “a0a” is the interface group for storage nodes ORA12C-01 and ORAC12C-02. The a0a VIF uses 10GbE ports e0e and e0g, with an MTU setting of 9,000.

Figure 47 NetApp Storage Configuration

The screenshot displays the NetApp Storage Configuration interface. The main window shows the 'Network' configuration for the cluster 'ORA12C'. The 'Ethernet Ports' tab is active, showing a table of network interfaces. Two 'Edit Interface Group' dialog boxes are open, one for the 'a0a' interface group on node 'ORA12C-01' and another for the 'a0a' interface group on node 'ORA12C-02'. Both dialog boxes show the following configuration:

IPSpace	Port	Broadcast Domain	Status	Type
Default	a0a	-NA-	Enabled	if_group

The 'Network Port Links' section in both dialog boxes shows a list of ports to include in the interface group. The ports listed are e0h, e0j, e0k, e0l, e0e, and e0g. The 'e0e' and 'e0g' ports are checked, indicating they are included in the interface group. The MTU Size (in Bytes) is set to 9000.

The 'Interface Group Properties' section at the bottom left shows the following details for the 'a0a' interface group:

- Distribution: port
- Create Policy: multimode_lacp
- Member List: [e0h, e0j, e0e, e0g]

VLANs “a0a-20” and “a0a-30” were created for the a0a interface group. Each of the VLANs also has a LIF created; these LIFs are as the mount points for NFS.

Figure 48 Network Interface

```

ORA12C::> network interface show
-----
Vserver      Logical      Status      Network      Current      Current      Is
Interface    Admin/Oper  Address/Mask Node          Port         Home
-----
Cluster
ORA12C-01_clus1  up/up      169.254.30.138/16  ORA12C-01    e0a         true
ORA12C-01_clus2  up/up      169.254.58.146/16  ORA12C-01    e0c         true
ORA12C-02_clus1  up/up      169.254.30.194/16  ORA12C-02    e0a         true
ORA12C-02_clus2  up/up      169.254.109.145/16 ORA12C-02    e0c         true
ORA12C
ORA12C-01_mgmt1  up/up      10.29.135.241/24   ORA12C-01    e0M         true
ORA12C-02_mgmt1  up/up      10.29.135.242/24   ORA12C-02    e0M         true
cluster_mgmt    up/up      10.29.135.245/24   ORA12C-01    e0i         true
ORA12C_SVM
n1_lif_20        up/up      10.20.20.24/24     ORA12C-01    a0a-20     true
n1_lif_30        up/up      10.30.30.25/24     ORA12C-01    a0a-30     true
n2_lif_20        up/up      10.20.20.25/24     ORA12C-02    a0a-20     true
n2_lif_30        up/up      10.30.30.24/24     ORA12C-02    a0a-30     true
11 entries were displayed.

ORA12C::> █

```

Figure 49 and Figure 50 illustrate the storage layout for aggregates, disks, and FlexVol volumes for the databases.

Figure 49 Storage Layout



Figure 50 Storage Volumes

Name	Aggregate	Status	Thin Provisioned	Total Space
n1_data_fin	aggr1_node1	Online	No	2 TB
n1_data_sh	aggr1_node1	Online	No	4 TB
n1_data_soe	aggr1_node1	Online	No	9 TB
n1_log_fin	aggr1_node1	Online	No	100 GB
n1_log_sh	aggr1_node1	Online	No	100 GB
n1_log_soe	aggr1_node1	Online	No	150 GB
n1_ocrvote1	aggr1_node1	Online	No	40 GB
n1_oradata1	aggr1_node1	Online	No	4 TB
n1_oralog1	aggr1_node1	Online	No	150 GB
n1_orarac1bin	aggr1_node1	Online	No	200 GB
n1_orarac3bin	aggr1_node1	Online	No	200 GB
n2_data_fin	aggr1_node2	Online	No	2 TB
n2_data_sh	aggr1_node2	Online	No	4 TB
n2_data_soe	aggr1_node2	Online	No	9 TB
n2_log_fin	aggr1_node2	Online	No	100 GB
n2_log_sh	aggr1_node2	Online	No	100 GB
n2_log_soe	aggr1_node2	Online	No	150 GB
n2_ocrvote2	aggr1_node2	Online	No	40 GB
n2_oradata2	aggr1_node2	Online	No	4 TB
n2_oralog2	aggr1_node2	Online	No	150 GB
n2_orarac2bin	aggr1_node2	Online	No	200 GB
n2_orarac4bin	aggr1_node2	Online	No	200 GB

In summary, each database has a total of four volumes that are distributed evenly across each node. For example, database “soe” has four volumes called “n1_data_soe,” “n2_data_soe,” “n1_log_soe,” and “n2_log_soe.” The first node (ORA12C-01) has volumes “n1_data_soe” and “n1_log_soe.” The second node (ORA12C-02) has volumes “n2_data_soe” and “n2_log_soe.” We used two OLTPs and one DSS database in a multi-tenant configuration for this solution.

The next section covers OS and database deployment configurations.

OS and Database Deployment

OS Configuration

For this solution, we configured a 4-node Oracle Database 12c RAC cluster using Cisco B460 M4 servers. We installed Oracle Linux 6.5 64-bit Operating System on these servers. Table 9 summarizes host configuration details.

Table 9 Host Configuration Details

Component	Details	Description
Server	B460 M4	4 Socket server (60 cores total) with HT enabled
Memory	512 GB	Physical memory for each server
NIC1 (eth0)	Public network	Management and public network interface with MTU=1500
NIC2 (eth1)	Oracle cluster Private Interconnect	Private server-to-server (Cache Fusion) traffic for Oracle RAC. MTU=9000
NIC3 (eth2)	NFS Storage A	Database IO traffic to NetApp Controller A. VLAN 20, MTU=9000
NIC4 (eth3)	NFS Storage B	Database IO traffic to NetApp Controller B. VLAN 30, MTU=9000
OS Configuration	Details	Description
Swap Space	16 GB	Oracle RAC requirement for Swap
Orarac{node#}bin node# = 1,2,3 or 4 as per server	200 GB	NFS mount directory for Oracle GRID, Database software and logs Note: 200 GB is not mandatory requirement for space. Less disk space can be configured as per requirements

Step-by-step OS installation details are not covered in this document, but the following are key steps for OS install:

- Download Oracle Linux 6.5 OS image from <https://edelivery.oracle.com/linux> or as appropriate.
- Launch KVM console on desired server, enable virtual media, map the Linux ISO image and reset the server. It should launch the Oracle Linux installer.
- Select language, fresh installation.

- Apply hostname and click on “configure network” to configure all network interfaces. Alternatively, you can only configure “Public Network” in this step. You can configure additional interfaces as part of post install steps.



Please verify the Device MAC Address and desired MTU settings as you configure the interfaces.

- For disk space and layout, ensure to select “Review and modify partitioning layout” so that you can adjust Swap Space to 16GB.
- As a part of additional RPM package, we recommend to select “customize now” and configure “UEK kernel Repo”.
- After OS install, reboot the server, complete appropriate registration steps. You can choose to synchronize the time with ntp server. Alternatively, you can choose to use Oracle RAC cluster synchronization daemon (OCSSD). Both ntp and OCSSD are mutually exclusive and OCSSD will be setup during GRID install if ntp is not configured.

Miscellaneous Post-Install Steps

Not all of the steps detailed below may be required for your setup. Validate and change as needed. The following changes were made on the test bed where Oracle RAC install was done.

Disable selinux

It is recommended to disable selinux. Edit `/etc/selinux/config` and change to:

```
SELINUX=disabled
#SELINUXTYPE=targeted
```

Modify/Create the DBA Group if Needed

```
groupmod -g 500 dba
```

Disable Firewalls

```
service iptables stop
service ip6tables stop
chkconfig iptables off
chkconfig ip6tables off
```

Make sure `/etc/sysconfig/network` has an entry for hostname.

Install `oracle-rdbms-server-12cR1-preinstall-1.0-11.el6.x86_64`

The actual version of RPM may change. The `oracle-rdbms-server-12cR1-preinstall` RPM packages are accessible through the Oracle Unbreakable Linux Network (ULN, which requires a support contract), from the Oracle Linux distribution media, or from the Oracle public yum repository.

This RPM performs a number of pre-configuration steps, including the following:

- Automatically downloading and installing any additional software packages and specific package versions needed for installing Oracle Grid Infrastructure and Oracle Database 12 c Release 1 (12.1) with package dependencies resolved via yum or up2date capabilities.
- Creating the user oracle and the groups oinstall (for OraInventory) and dba (for OSDBA), which are used during database installation. (For security purposes, this user has no password by default and cannot log in remotely. To enable remote login, please set a password using the passwd tool.)
- Modifying kernel parameters in /etc/sysctl.conf to change settings for shared memory, semaphores, the maximum number of file descriptors, and so on.
- Setting hard and soft shell resource limits in /etc/security/limits.conf, such as the locked-in memory address space, the number of open files, the number of processes, and core file size.
- Setting numa=off in the kernel for x86_64 machines.



The oracle-rdbms-server-12cR1-preinstall and oracle-rdbms-server-11gR2-preinstall parses the existing /etc/sysctl.conf and /etc/security/limits.conf files and updates values only as needed for database installation. Any pre-customized settings not related to database installation are left as is.

Update Sysctl.conf Settings

The following are recommended settings for NFS environment. Please refer to Oracle support note 1519875.1 for more details. Please note that these setting may change as new architectures evolve. Refer to Appendix B for complete sysctl.conf parameter list.

```
net.core.rmem_default = 134217728
net.core.rmem_max = 134217728
net.core.wmem_default = 134217728
net.core.wmem_max = 134217728
##### Additional NFS settings #####
net.ipv4.tcp_rmem = 4096 87380 134217728
net.ipv4.tcp_wmem = 4096 65536 134217728
net.ipv4.tcp_window_scaling = 1
net.ipv4.tcp_syncookies = 0
net.ipv4.tcp_timestamps = 0
net.ipv4.tcp_sack = 0
net.ipv4.tcp_no_metrics_save = 1
#net.ipv4.tcp_moderate_rcvbuf = 0
net.core.netdev_max_backlog = 300000
net.ipv4.tcp_moderate_rcvbuf = 1
sunrpc.tcp_slot_table_entries = 128
```

Update Recommended Cisco VIC Drivers Per Firmware

Go to <http://software.cisco.com/download/navigator.html>

In the download page, select servers–Unified computing. On the right menu, select your class of servers, for example Cisco UCS B-Series Blade server software, and then select Unified Computing System (UCS) Drivers on the following page. Select appropriate ISO image for UCS-related drivers based on your firmware. Extract and install the appropriate eNIC (and fnic if SAN setup) RPM.

Oracle Database 12c GRID Infrastructure Deployment

Prior to GRID and database install, verify all the prerequisites are completed. This document does not cover the step-by-step installation for Oracle GRID but will provide partial summary of details that might be relevant. Oracle Grid Infrastructure ships with the Cluster Verification Utility (CVU) that can be run to validate pre and post installation configurations. Prior to GRID install, complete the steps detailed below.

Configure Private and Storage NICs

Skip this step if you have configured NICs during OS install.

Establish SSH Connectivity

```
./sshUserSetup.sh -user oracle -hosts " orarac1 orarac2 orarac3 orarac4" -noPromptPassphrase -confirm -advanced
```

Run Cluster Verification Utility

```
./runcluvfy.sh stage -pre crsinst -n orarac1,orarac2,orarac3,orarac4 -verbose
```

Create Local Directory Structures on Each Node

We created following local directories on each node:

ocrvote: OCR and voting for GRID infrastructure

u01: GRID and Database software binaries

<dbname>_data1: Data files for database on Controller A

<dbname>_data2: Data files for database on Controller B

<dbname>_log1: Log files database on Controller A

<dbname>_log2: Log files for database on Controller B

Create NFS Mount Points for Each Volume in /etc/fstab

Edit /etc/fstab file in each RAC node and add mount points for all database and GRID NFS volumes with the appropriate mount options. Please note that these mount point directories need to be created first.



The Oracle Direct NFS (dNFS) configuration is completed at a later stage.

When you complete `/etc/fstab` edit, mount those mount points via “`mount -a`” command. The following is a sample output from `mount` command on Node 1 (orac1):

```
[root@orac1 ~]# mount

...

n1_lif_20:/n1_orac1bin on /u01 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.20.20.24)

n1_lif_20:/n1_ocrvote1 on /ocrvote type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.20.20.24)

n1_lif_30:/n1_data_sh on /sh1_data1 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.30.30.25)

n2_lif_20:/n2_data_sh on /sh1_data2 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.20.20.25)

n1_lif_30:/n1_log_sh on /sh1_log1 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.30.30.25)

n2_lif_20:/n2_log_sh on /sh1_log2 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.20.20.25)

n1_lif_20:/n1_log_fin on /fin_log1 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.20.20.24)

n1_lif_20:/n1_data_fin on /fin_data1 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.20.20.24)

n2_lif_30:/n2_data_fin on /fin_data2 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.30.30.24)

n2_lif_30:/n2_log_fin on /fin_log2 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.30.30.24)

n1_lif_20:/n1_log_soe on /log1 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.20.20.24)

n1_lif_20:/n1_data_soe on /data1 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.20.20.24)

n2_lif_30:/n2_data_soe on /data2 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.30.30.24)
```



```
n2_lif_30:/n2_log_soe on /log2 type nfs
(rw,bg,hard,rsize=65536,wsiz=65536,nfsvers=3,actimeo=0,nointr,timeo=600,tcp,addr
=10.30.30.24)
```

Create OCR and Voting Disk Files

Log in as "grid" user from any one node and create the following raw files:

```
dd if=/dev/zero of=/ocrvote/ocr1 bs=1024k count=5120
dd if=/dev/zero of=/ocrvote/ocr2 bs=1024k count=5120
dd if=/dev/zero of=/ocrvote/ocr3 bs=1024k count=5120
dd if=/dev/zero of=/ocrvote/vote1 bs=1024k count=5120
dd if=/dev/zero of=/ocrvote/vote2 bs=1024k count=5120
dd if=/dev/zero of=/ocrvote/vote3 bs=1024k count=5120
```

Configure HugePages

HugePages is a method to have larger page size that is useful for working with very large memory. For Oracle Databases, using HugePages reduces the operating system maintenance of page states, and increases Translation Lookaside Buffer (TLB) hit ratio.

Advantages of HugePages:

- HugePages are not swappable so there is no page-in/page-out mechanism overhead.
- HugePage uses fewer pages to cover the physical address space, so the size of "book keeping" (mapping from the virtual to the physical address) decreases, so it requiring fewer entries in the TLB and so TLB hit ratio improves.
- HugePages reduces page table overhead.
- Eliminated page table lookup overhead: Since the pages are not subject to replacement, page table lookups are not required.
- Faster overall memory performance: On virtual memory systems each memory operation is actually two abstract memory operations. Since there are fewer pages to work on, the possible bottleneck on page table access is clearly avoided.

For our configuration, we used HugePages for all OLTP and DSS workloads.

Please refer to Oracle support (formerly metalink) document 361323.1 for HugePages configuration details.

Install and Configure Oracle GRID

It is not within the scope of this document to include the specifics of an Oracle RAC installation. Please refer to the Oracle installation documentation for specific installation instructions for your environment.

The following are the Oracle GRID install steps summary:

- Launch Oracle GRID installer

- Select “Install and Configure Oracle Grid Infrastructure for a Cluster”
- Next screen, select “Configure Standard Cluster”
- Enter Cluster name, SCAN name and GNS configuration details.
- Select all nodes ROLE as “HUB”. You can check/verify SSH connectivity in this step as well.
- Select and verify Public and Private Network Interface details.
- For storage option, select “Shared File System”. In subsequent screens, select “Normal Redundancy” and provide OCR and Vote disk locations.
- Select to “Automatically run configuration scripts”. This should complete GRID install.

When the GRID install is successful, login to each of the nodes and perform minimum health checks to make sure that Cluster state is healthy.

Install Database Software

After successful GRID install, we recommend to install Oracle Database 12c software only. You can create databases using DBCA or database creation scripts at later stage.

At this point, we are ready to run synthetic IO tests against this infrastructure setup. We used fio and Oracle orion as primary tools for IO tests. Please note that we will configure Direct NFS as we get into the actual database testing with Swingbench later.

Validation

Test Plan

Infrastructure Tests

Before configuring any database for workload tests, it is extremely important to validate that this is indeed a balanced configuration that is capable of delivering expected performance. We use widely adopted synthetic IO generation tools such as Orion and Linux FIO for this exercise. During our testing, we validated that both Orion and FIO generate similar performance results.

8KB Random Read IOPs Tests

Figure 51 8K Node Scalability

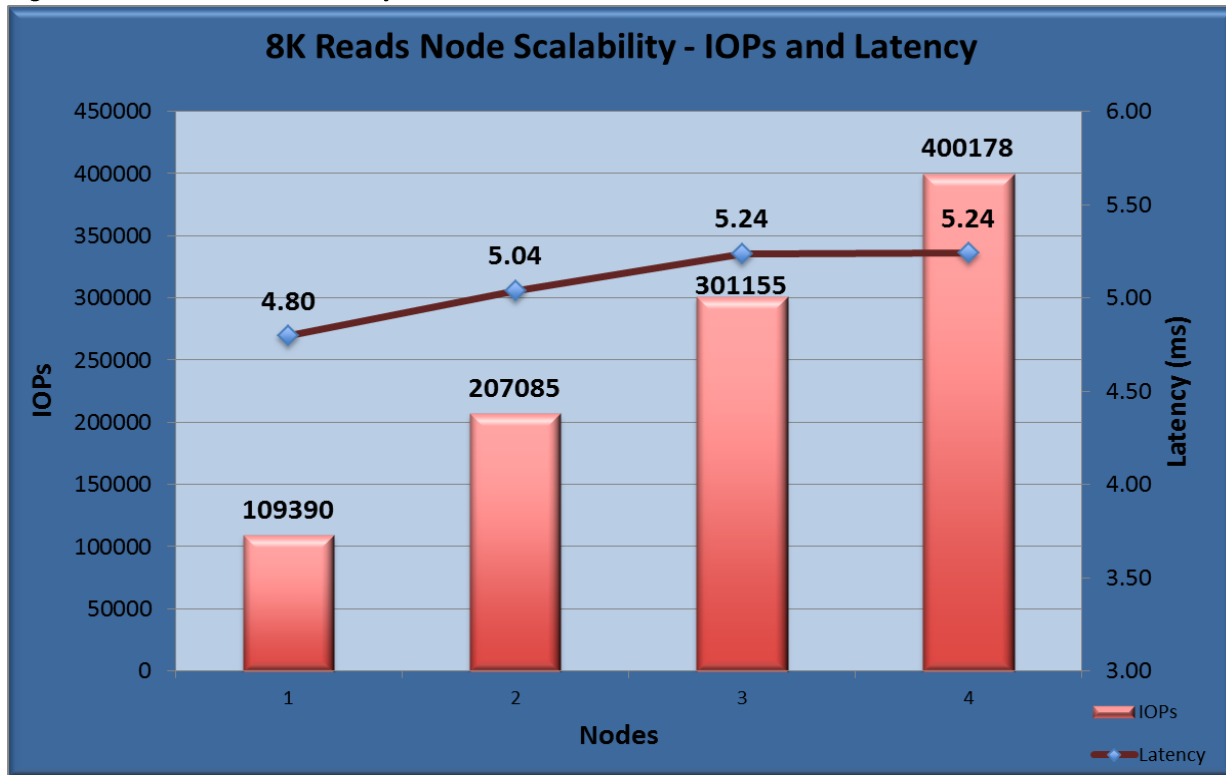
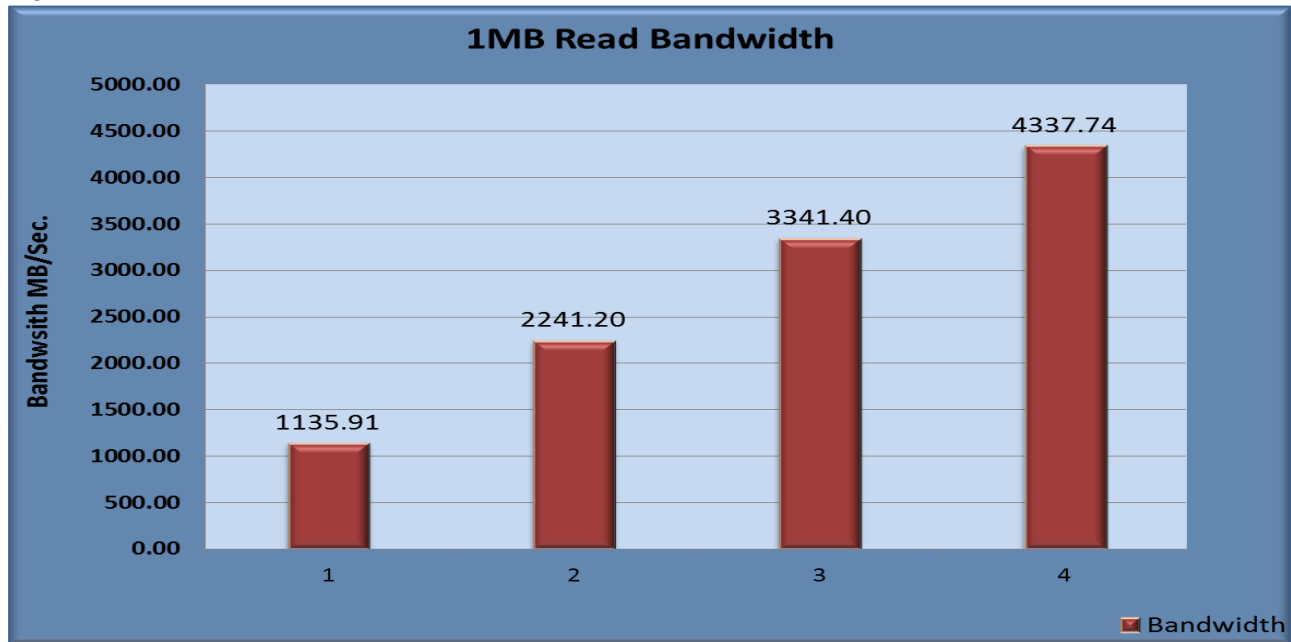


Figure 51 shows the 8KB random read results representing OLTP database. The results showcase linear scalability as we distribute IO generation from Node 1 to Node 4. The latency is also almost constant and does not increase significantly as load increases from Node 1 to Node 4. We also ran the tests for 3 hours to help ensure that this configuration is capable of sustaining this load for longer period of time.

Bandwidth Tests

Figure 52 Bandwidth Tests



The bandwidth tests are carried out with 1MB IO size and represent DSS database workload. As shown above, the bandwidth scaled linearly as we scaled nodes from 1 to 4. With four nodes, we could generate about 4.3 GBytes/sec of sustained bandwidth over 3 hours run period. We did not see any performance dips or degradation over the period of run time. It is also important to note that this is not a benchmarking exercise and the numbers presented are not the peak numbers where there is hardware resource saturation. These are practical and out of box test numbers that can be easily reproduced by any one. At this time, we are ready to create OLTP database(s) and continue with database tests.

Database Creation with DBCA

We used Oracle Database Configuration Assistant (DBCA) to create two OLTP and one DSS databases. Alternatively, you can use Database creation scripts to create the databases as well. Please ensure to place the datafiles, redo logs and control files in appropriate directory paths discussed in the storage layout section. We will discuss OLTP and DSS schema creation along with data population in the workload section.

Direct NFS Configuration

We recommend to configure Oracle Database to access NFS V3 servers directly using an Oracle internal Direct NFS client instead of using the operating system kernel NFS client. To enable Oracle Database to use Direct NFS, the NFS file systems must be mounted and available over regular NFS mounts before you start installation. Direct NFS manages settings after installation. You should still set the kernel mount options as a backup, but for normal operation, Direct NFS will manage NFS mounts.



Some NFS file servers require NFS clients to connect using reserved ports. If your storage system is running with reserved port checking, then you must disable it for Direct NFS to operate. To disable reserved port checking, consult your NetApp documentation.

To enable Direct NFS (dNFS) for Oracle RAC, the orafstab must be configured on all nodes and it must be synchronized on all nodes. The direct NFS client looks for NFS details in the following locations:

- \$ORACLE_HOME/dbs/orafstab
- /etc/orafstab
- /etc/mtab

When the orafstab file is placed in /etc, then it is globally available to all Oracle databases, and can contain mount points used by all Oracle databases running on nodes in the cluster, including single-instance databases. However, on Oracle RAC systems, if the orafstab file is placed in /etc, then you must replicate the file /etc/orafstab file on all nodes, and keep each /etc/orafstab file synchronized on all nodes, just as you must with the /etc/fstab file.

Typical syntax for orafstab is:

```
server: <NFS Server/Storage name or IP>
path: <First path to NFS server or IP of NFS server NIC>
local: <First client-side or RAC node storage NIC IP>
path: <Second path to NFS server or IP of NFS server NIC>
local: <Second client-side or RAC node storage NIC IP>
export: <local server mount directory> mount: <NFS storage mount point>
```

The following is sample orafstab configuration from RAC node 1:

```
[oracle@orarac1 dbs]$ cat /etc/orafstab
server:10.20.20.24
path:10.20.20.24
local:10.20.20.51
local:10.30.30.51
server:10.30.30.24
path:10.30.30.24
local:10.30.30.51
local:10.20.20.51
server:10.20.20.25
path:10.20.20.25
local:10.20.20.51
local:10.30.30.51
server:10.30.30.25
path:10.30.30.25
```

```

local:10.30.30.51
local:10.20.20.51
export:/log1 mount:/n1_log_soe
export:/log2 mount:/n2_log_soe
export:/data1 mount:/n1_data_soe
export:/data2 mount:/n2_data_soe
export:/log1 mount:/n1_log_fin
export:/log2 mount:/n2_log_fin
export:/data1 mount:/n1_data_fin
export:/data2 mount:/n2_data_fin
export:/sh1_data1 mount:/n1_data_sh
export:/sh1_data2 mount:/n2_data_sh
export:/sh1_log1 mount:/n1_log_sh
export:/sh1_log2 mount:/n2_log_sh

```

When oranfstab file is created on all nodes, you need to enable the direct NFS client ODM library on all nodes. Please shutdown all the databases before this step. You can perform this task two ways:

```

cd $ORACLE_HOME/lib
mv libodm12.so libodm12.so_stub
ln -s libnfsodm12.so libodm12.so

```

Or

Enable dNFS through “make -f ins_rdbms.mk dnfs_on” command.

This completes dNFS setup. You can start the database(s) next and validate the dNFS client usage via following views.

- v\$d_nfs_servers
- v\$d_nfs_files
- v\$d_nfs_channels
- v\$d_nfs_stats

The following is sample output from one of the OLTP databases in this solution:

```
SQL> select SVRNAME, DIRNAME from v$d_nfs_servers;
```

```

SVRNAME                                DIRNAME
-----                                -
n1_lif_20                               /n1_data_soe

```

```
n1_lif_20          /n1_log_soe
n2_lif_30          /n2_data_soe
n2_lif_30          /n2_log_soe
```



n1_lif_20 has IP address 10.20.20.24 and n2_lif_30 has IP address 10.30.30.24.

Database Workload Configuration

We used Swingbench for workload testing. Swingbench is a simple to use, free, Java based tool to generate database workload and perform stress testing using different benchmarks in Oracle database environments. Swingbench provides four separate benchmarks, namely, Order Entry, Sales History, Calling Circle, and Stress Test. For the tests described in this paper, Swingbench Order Entry benchmark was used for OLTP workload testing and the Sales History benchmark was used for the DSS workload testing. The Order Entry benchmark is based on SOE schema and is TPC-C like by types of transactions. The workload uses a very balanced read/write ratio around 60/40 and can be designed to run continuously and test the performance of a typical Order Entry workload against a small set of tables, producing contention for database resources. The Sales History benchmark is based on the SH schema and is TPC-H like. The workload is query (read) centric and is designed to test the performance of queries against large tables.

For this solution, we created two OLTP (Order Entry) and one DSS (Sales History) database to demonstrate database consolidation, multi-tenancy capability, performance and sustainability. The OLTP databases are approximately 6 TB and 3 TB in size while DSS database is approximately 4 TB in size.

Typically encountered in the real world deployments, we tested a combination of scalability and stress related scenarios that ran concurrently on a 4-node Oracle RAC cluster configuration.

- OLTP user scalability and OLTP cluster scalability representing small and random transactions
- DSS workload representing larger transactions
- Mixed workload featuring OLTP and DSS workloads running simultaneously for 24 hours

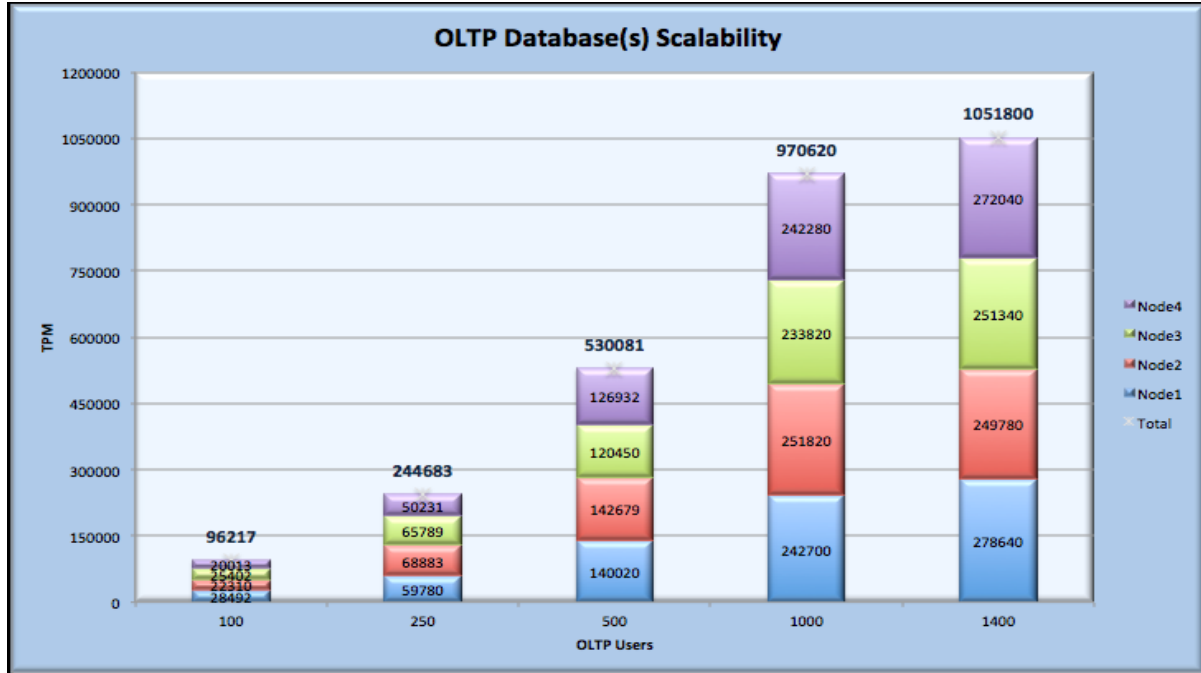
Database Performance

OLTP Performance

The first step after the databases creation is calibration; about the number of concurrent users, OS and database optimization. For OLTP workload featuring Order Entry schema, we used two databases. For the larger database (6TB) OLTP1, we used 64GB SGA and smaller database (3TB) OLTP2, we used 48GB SGA. For OLTP1, we selected 1,000 concurrent users and OLTP2, we used 400 concurrent users. We also ensured that HugePages were in use. Each OLTP scalability test was run for at least 12 hours and ensured that results are consistent for the duration of the full run.

For OLTP workloads, the common measurement metrics are Transactions Per Minute (TPM), and Users and IOPs scalability. Here are the results from scalability testing for Order Entry workload.

Figure 53 OLTP Database Scalability



We ran tests with 100, 250, 500 and 1,000 users across 4-node cluster. We also scaled users further by 400 (total 1,400) by running a second OLTP database in the same cluster. During the tests, we validated that Oracle SCAN listener fairly and evenly load balanced users across all four nodes of the cluster. We also observed appropriate scalability in TPMs as number of users across clusters increased. Next graph shows IO, read/write ratio and node level scalability as number of nodes in the cluster increased.

Figure 54 OLTP Read/Write IOPs

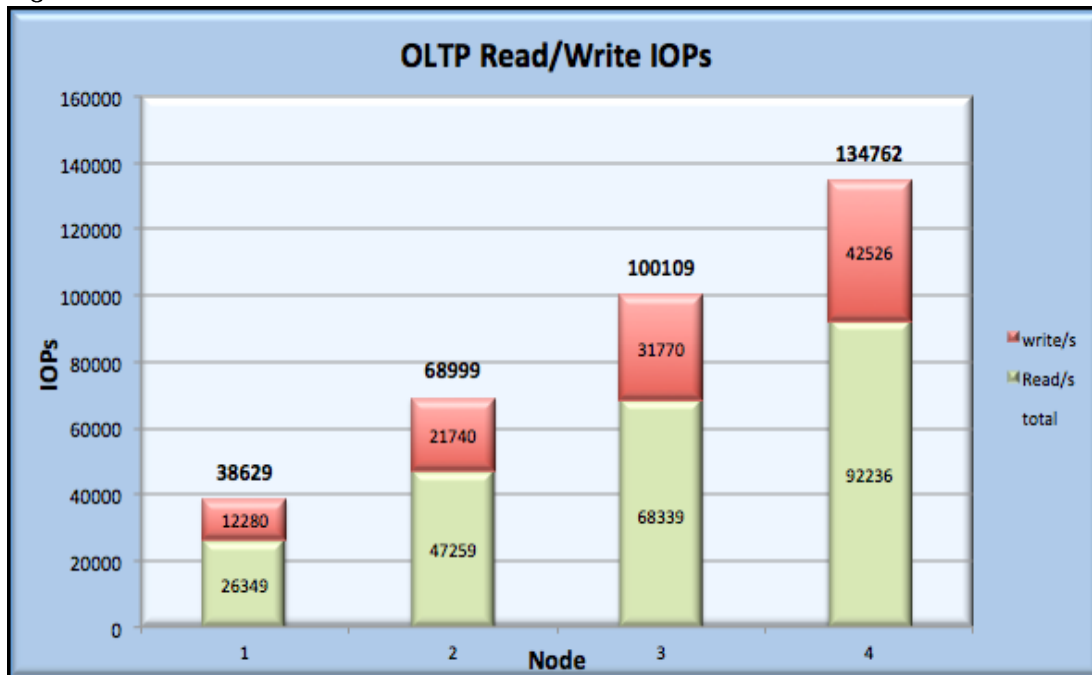


Figure 55 System Statistics

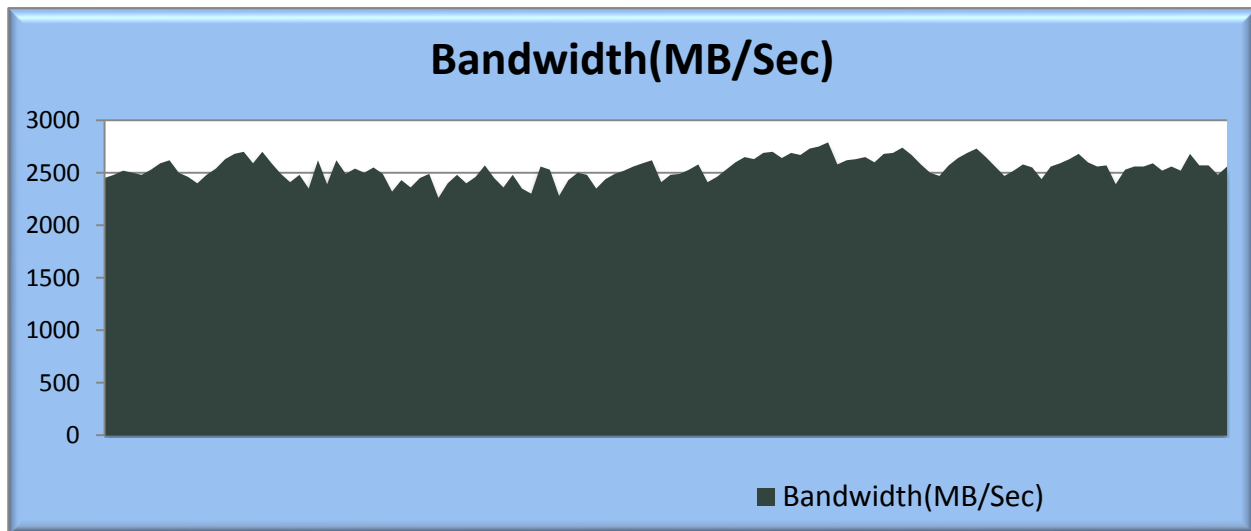
System Statistics - Per Second										
DB/Inst: SOE/soe1 Snaps: 85-89										
I#	Logical Reads/s	Physical Reads/s	Physical Writes/s	Redo Size (k)/s	Block Changes/s	User Calls/s	Execs/s	Parses/s	Logons/s	Txns/s
1	404,638.31	19,132.3	8,766.0	12,770.1	79,432.9	7,393.5	38,348.6	608.4	0.45	3,688.2
2	352,147.31	13,775.6	6,181.5	8,961.7	55,904.9	5,223.0	27,080.2	496.7	0.45	2,603.8
3	359,539.47	14,056.7	6,617.3	9,198.5	57,481.0	5,325.7	27,620.4	506.2	0.44	2,655.3
4	333,359.68	16,017.9	7,102.2	10,426.6	65,282.0	6,074.6	31,502.4	577.3	0.44	3,028.4
Sum	1,449,684.77	62,982.5	28,667.0	41,356.9	258,100.8	24,016.8	124,551.6	2,188.5	1.78	11,975.6
Avg	362,421.19	15,745.6	7,166.7	10,339.2	64,525.2	6,004.2	31,137.9	547.1	0.44	2,993.9
Std	30,225.34	2,468.3	1,130.5	1,743.1	10,750.9	1,001.0	5,195.0	54.4	0.00	500.0

Figure 55 shows a section from 3-hour window of AWR report from the run that highlights Physical Reads/Sec. and Physical Writes/Sec for each instance. This section highlights that IO load is distributed across all the cluster nodes performing workload operations. Due to variations in workload randomness, we conducted multiple runs to ensure consistency in behavior and test results.

DSS Performance

DSS workloads are generally sequential in nature, read intensive and exercise large IO size. DSS workload runs a small number of users that typically exercise extremely complex queries that run for hours. For this test, we ran Swingbench Sales history workload with 60 users. The charts below show DSS workload results.

Figure 56 Bandwidth for DSS Performance



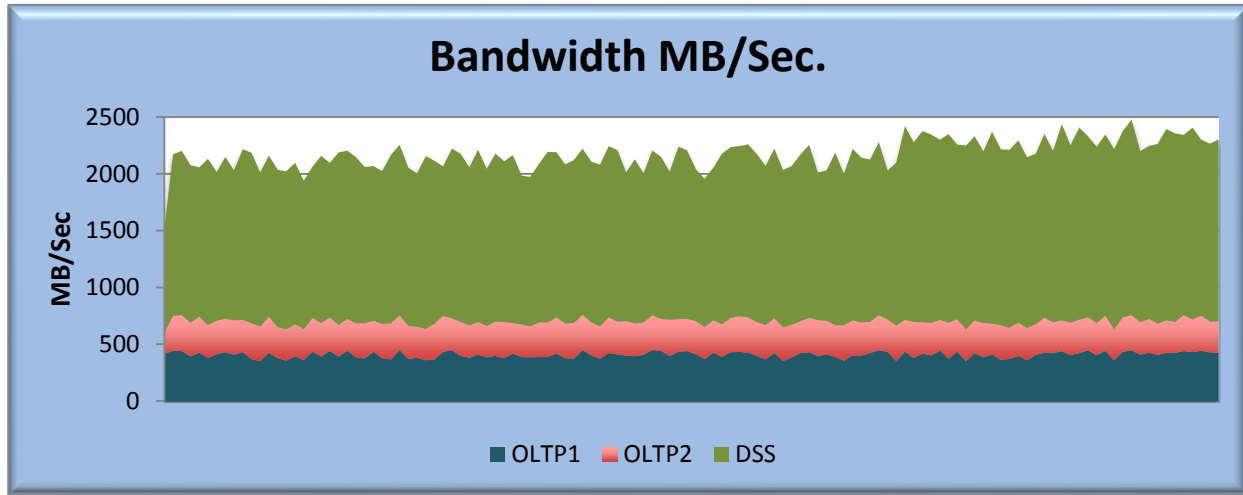
For 24 hours DSS workload test, we observed total sustained IO bandwidth was up to 2.7GB/sec after the initial ramp up workload. As indicated on the charts, the IO was also evenly distributed across both NetApp

FAS storage controllers and we did not observe any significant dips in performance and IO bandwidth for a sustained period of time.

Mixed Workload

The next test is to run both OLTP and DSS workloads simultaneously. This test will ensure that configuration in this test is able to sustain small random queries presented via OLTP along with large and sequential transactions submitted via DSS workload. We ran the test for 24 hours. The results are as shown in the chart below.

Figure 57 Bandwidth for Mixed Workloads

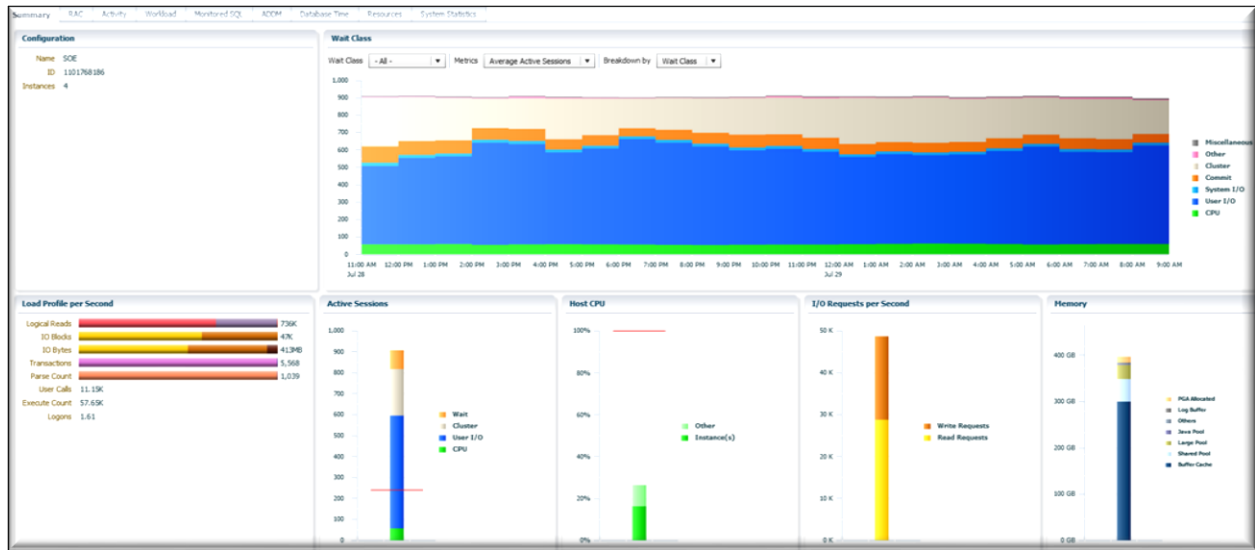


For mixed workloads running for 24 hours, we observed approximately 2GB/Sec IO bandwidth. The OLTP transactions also averaged between 300K and 340K transactions per minute.

OLTP1 Database Performance

OLTP1 Database activity is captured for all four Oracle RAC Instances using Oracle Enterprise Manager for 24 hours mixed workload test.

Figure 58 OLTP1 Database Performance Summary



RAC Global Cache Blocks Received and Global Cache Blocks Get Time is captured for OLTP1 database for 24 hours mixed workload test.

Figure 59 OLTP1 Database RAC Global Cache Blocks Received and Blocks Get Time



Physical I/Os per second is captured for OLTP1 database for 24 hours mixed workload test.

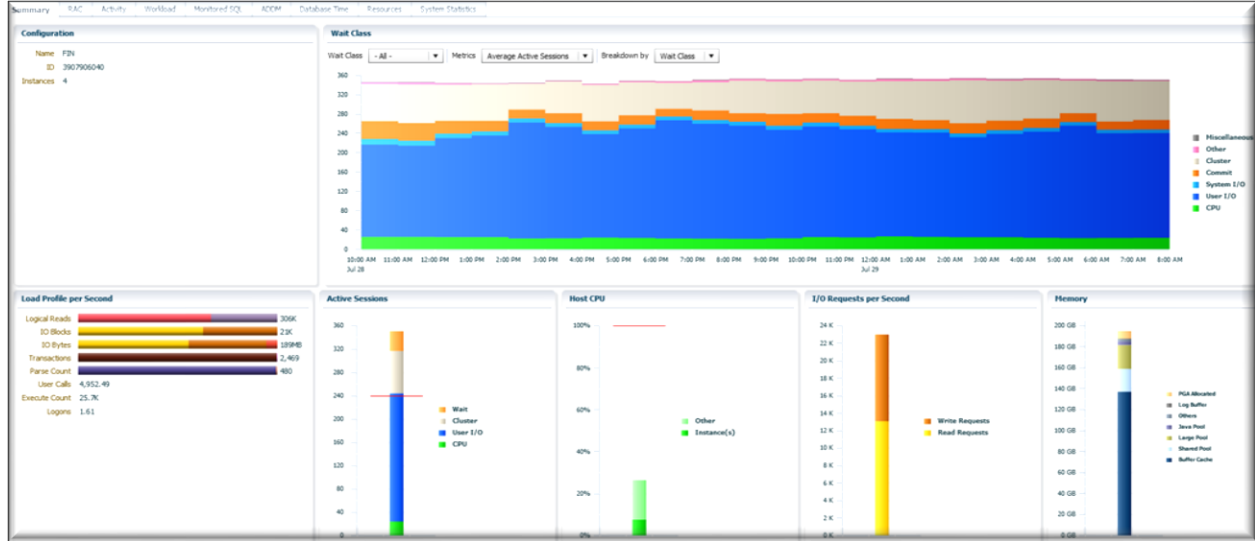
Figure 60 OLTP1 Database Physical I/Os per Second



OLTP2 Database Performance

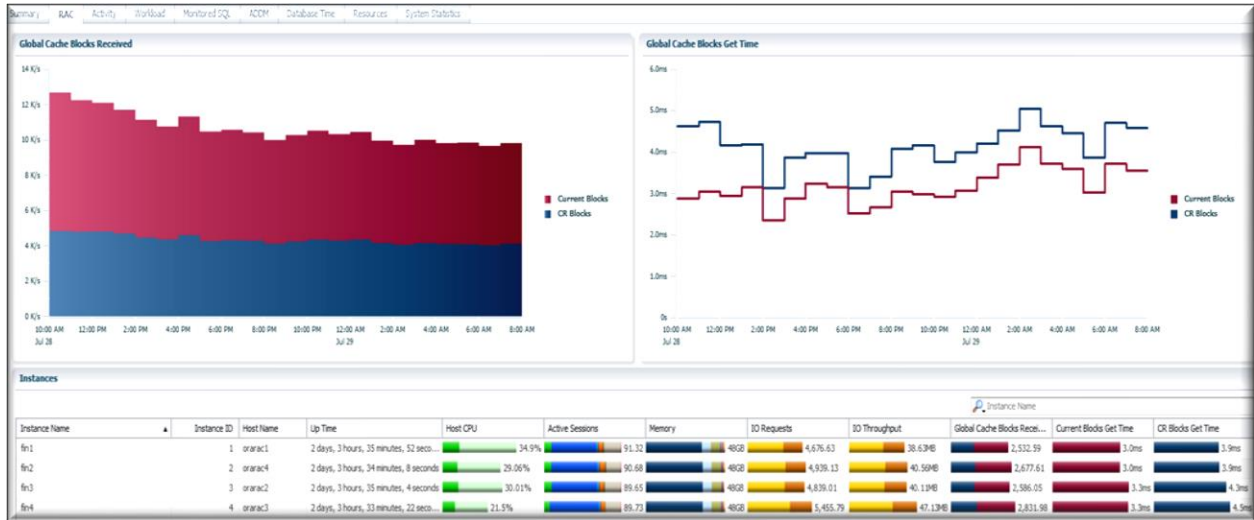
OLTP2 Database activity is captured for all four Oracle RAC Instances using Oracle Enterprise Manager for 24 hours mixed workload test.

Figure 61 OLTP2 Database Performance Summary



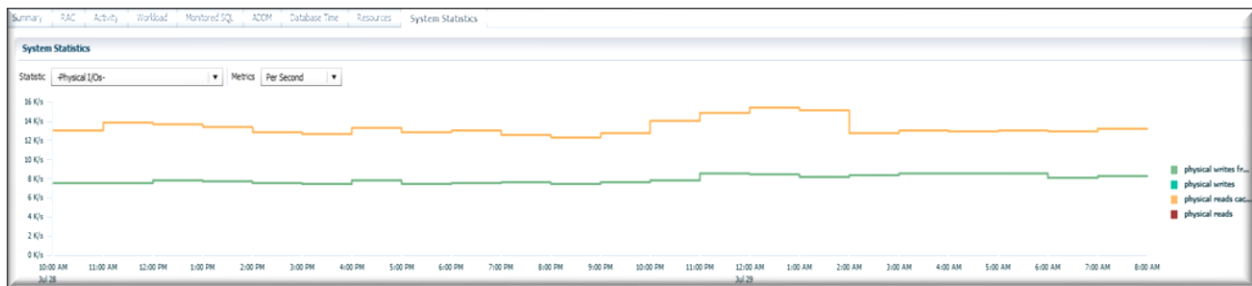
RAC Global Cache Blocks Received and Global Cache Blocks Get Time is captured for OLTP2 database for 24 hours mixed workload test.

Figure 62 OLTP2 Database RAC Global Cache Blocks Received and Blocks Get Time



Physical I/Os per second is captured for OLTP2 database for 24 hours mixed workload test.

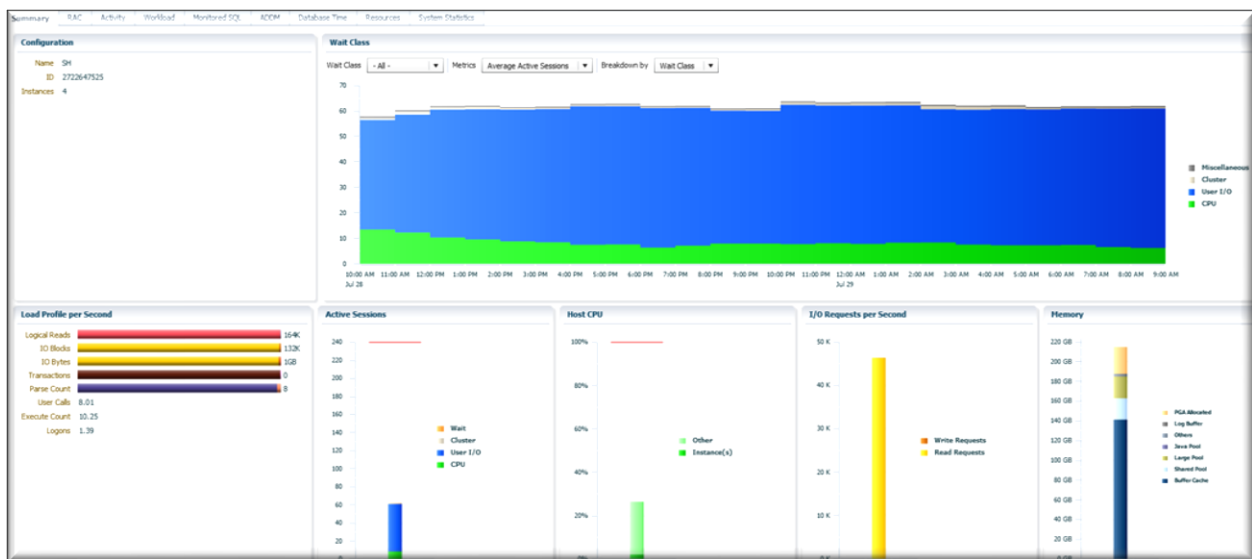
Figure 63 OLTP2 Database Physical I/Os Per Second



DSS Performance

DSS Database activity is captured for all four Oracle RAC Instances using Oracle Enterprise Manager for 24 hours mixed workload test.

Figure 64 DSS Performance



Additional Observation:

- We analyzed storage reports and found that both SSD and HDD were reasonably utilized around 50% and workload was not primarily coming out of SSD. For OLTP workloads, we observed random read latency (db file sequential reads) for 8K block around 8ms on HDD's. For SSDs, we observed latencies around 0.5 millisecond at the consistent load.
- The servers CPU utilization averaged around 35% for the 24 hour run and we did not observe any queuing or bandwidth saturation for the Network interfaces also.

Resiliency Tests

The goal of these tests is to ensure that reference architecture withstands commonly occurring failures either due to unexpected crashes, hardware failures or human errors. We conduct many hardware, software (process kills) and OS specific failures that simulate real world scenarios under stress condition.

Table 10 highlights some of the test cases.

Table 10 Resiliency Tests

Scenario	Test	Status
Test 1 - UCS 6248 Fabric-B Failure Test	Run the system on Full Load. Reboot Fabric-B, let it join the cluster back and check network traffic on Fabric-A.	Fabric Failover did not cause any disruption to Private Interconnect network traffic and StorageB network traffic
Test 2 - UCS 6248 Fabric-A Failure Test	Run the system on Full Load. Reboot Fabric-A, let it join the cluster back and check network traffic on Fabric-B.	Fabric Failover did not cause any disruption to Public network traffic and StorageA network traffic
Test 3 - Nexus 9396PX Switch Failure Test	Run the system on Full Load. Reboot Nexus A, let it join the cluster back and check network traffic on Nexus B. Reboot Nexus B, let it join the cluster back and check network traffic on Nexus A.	Nexus Switch Failover did not cause any disruption to Public network traffic, Private Interconnect network traffic and Storage network traffic

Figure 65 illustrates the FlexPod solution infrastructure diagram under normal conditions. The green lines indicate the public network connecting to Fabric interconnect A and the red lines indicate the private interconnects connecting to Fabric interconnect B. The public and private VLANs spanning the fabric interconnects secure the connectivity in case of link failure. Both green and red links also carry NFS storage traffic.

Figure 65 Flexpod Infrastructure

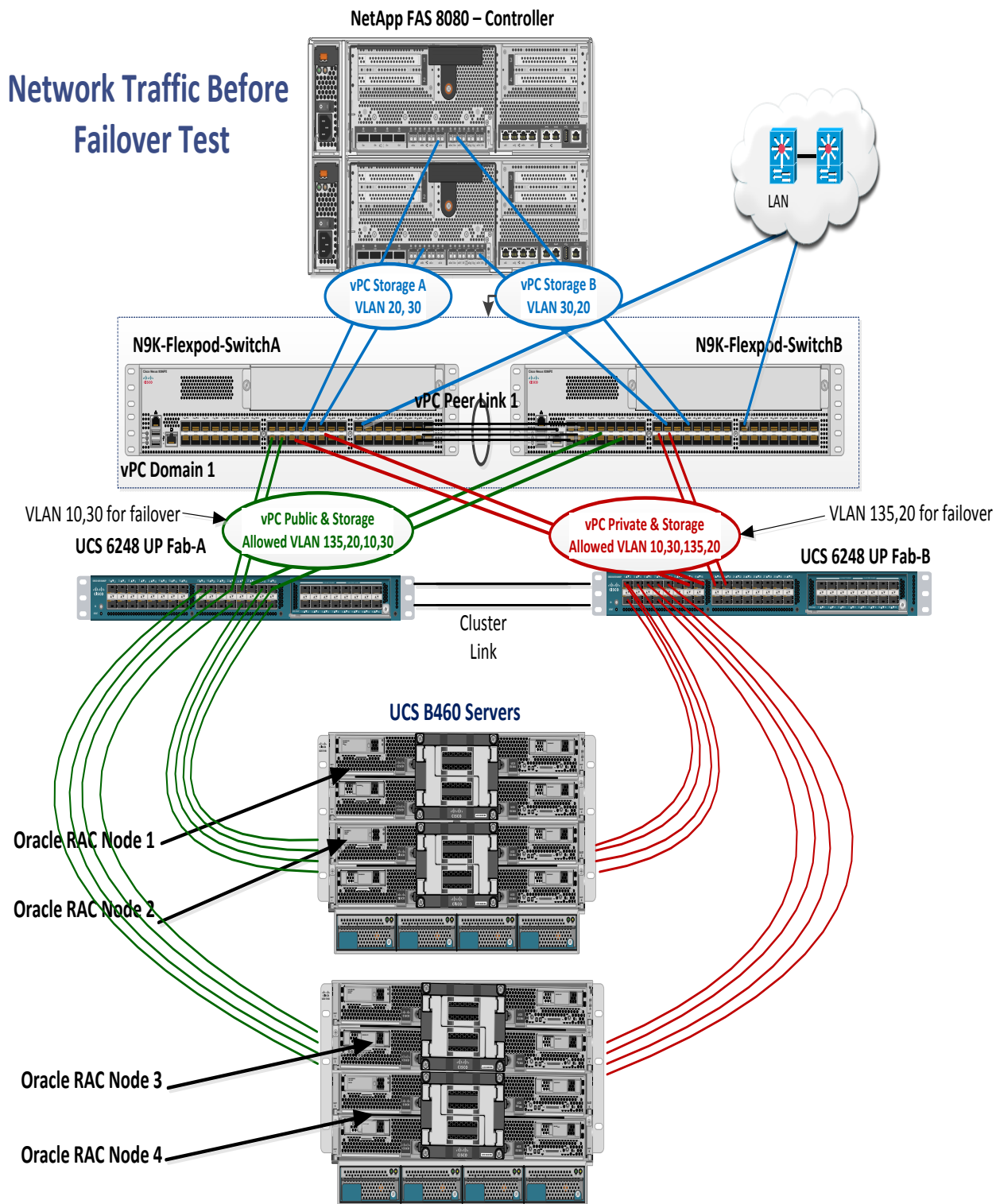


Table 11 shows a complete infrastructure detail of MAC address, OS address, VLAN information and Server connections for Cisco UCS Fabric Interconnect FI-A and Fabric Interconnect FI-B switches before failover test.

Table 11 MAC Addresses and VLAN Information on Fabric Interconnect A and Fabric Interconnect B

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN
1	ORARAC1	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:C C1E	eth0	FI_A	135
			Adapter 1	NIC2	vNIC2_Private	0025:B500:C C0E	eth1	FI_B	10
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C C3F	eth2	FI_A	20
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C C2F	eth3	FI_B	30
	ORARAC2	Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:C C1D	eth0	FI_A	135
			Adapter 1	NIC2	vNIC2_Private	0025:B500:C C0D	eth1	FI_B	10
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C C3D	eth2	FI_A	20
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C C2D	eth3	FI_B	30
2	ORARAC3	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:C C1C	eth0	FI_A	135
			Adapter 1	NIC2	vNIC2_Private	0025:B500:C C0C	eth1	FI_B	10
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C C3C	eth2	FI_A	20
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C C2C	eth3	FI_B	30
	ORARAC4	Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:C C1B	eth0	FI_A	135
			Adapter 1	NIC2	vNIC2_Private	0025:B500:C C0B	eth1	FI_B	10
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C C3B	eth2	FI_A	20
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C C2B	eth3	FI_B	30

Log in to Cisco Fabric Interconnect A and “connect nxos A” then type “show mac address-table” to see all VLAN connection on Fabric-A as shown in the table below.

Table 12 MAC Addresses on Fabric Interconnect – A

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN
1	ORARAC1	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:CC1E	eth0	FI_A	135
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:CC3F	eth2	FI_A	20
	ORARAC2	Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:CC1D	eth0	FI_A	135
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:CC3D	eth2	FI_A	20
2	ORARAC3	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:CC1C	eth0	FI_A	135
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:CC3C	eth2	FI_A	20
	ORARAC4	Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:CC1B	eth0	FI_A	135
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:CC3B	eth2	FI_A	20

Log in to Cisco Fabric Interconnect B and “connect nxos B” then type “show mac address-table” to see all VLAN connection on Fabric-B as shown in the table below.

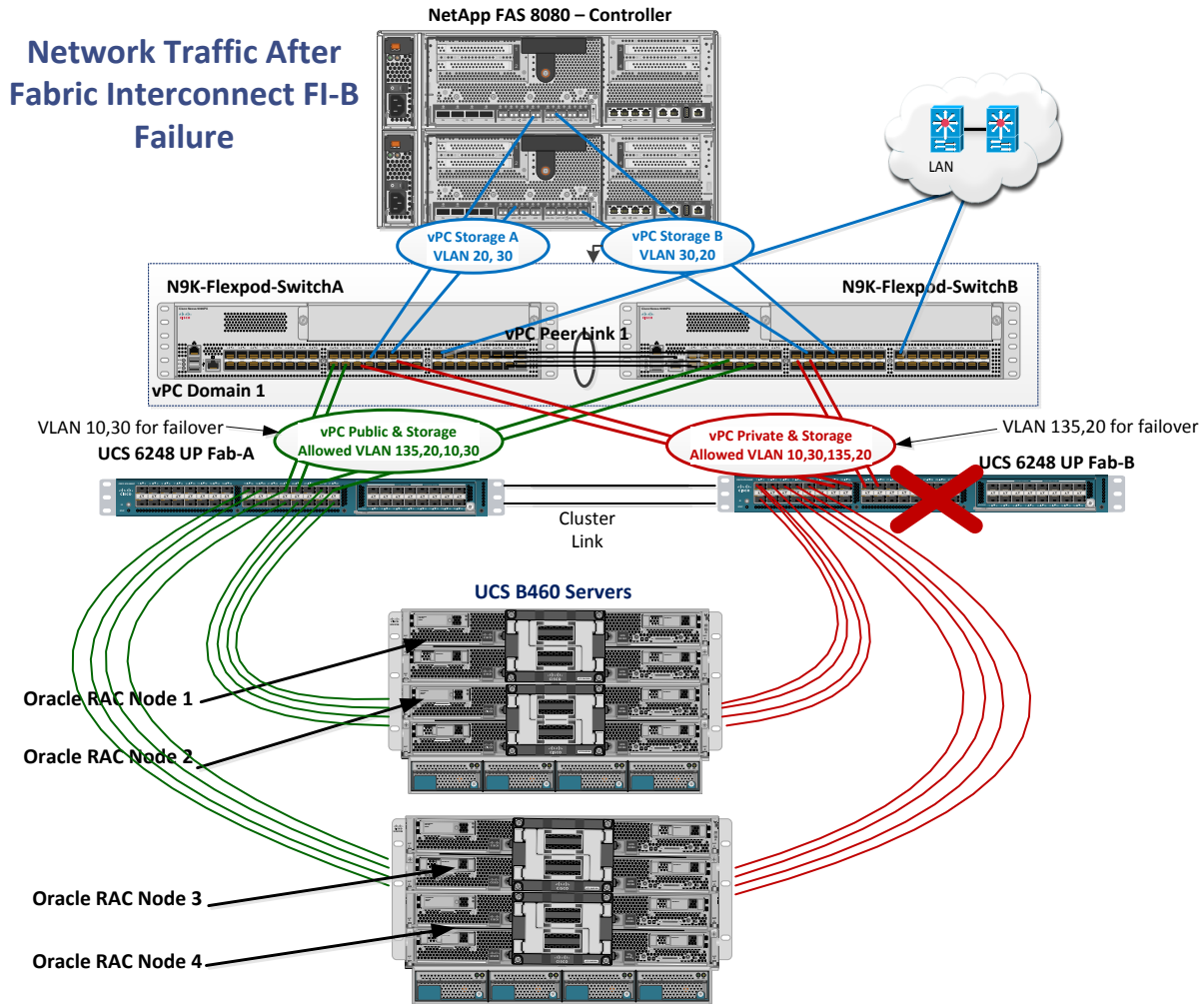
Table 13 MAC Addresses on Fabric Interconnect – B

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN
1	ORARAC1	Server 3	Adapter 1	NIC2	vNIC2_Private	0025:B500:CC0E	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:CC2F	eth3	FI_B	30
	ORARAC2	Server 7	Adapter 1	NIC2	vNIC2_Private	0025:B500:CC0D	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:CC2D	eth3	FI_B	30
2	ORARAC3	Server 3	Adapter 1	NIC2	vNIC2_Private	0025:B500:CC0C	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:CC2C	eth3	FI_B	30
	ORARAC4	Server 7	Adapter 1	NIC2	vNIC2_Private	0025:B500:CC0B	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:CC2B	eth3	FI_B	30

Test 1 - Cisco UCS 6248 Fabric-B Failure Test

Figure 66 illustrates how during fabric interconnect B switch failure, the respective blades (ORARAC1 and ORARAC2) on chassis 1 and (ORARAC3 and ORARAC4) on chassis 2 will fail over the MAC addresses and its VLAN to fabric interconnect A.

Figure 66 Fabric Interconnect - B Failure



The table below shows all the MAC address and VLAN information on Cisco UCS Fabric Interconnect A. In the table, all the switches over VLAN from Fabric Interconnect B to Fabric Interconnect A are shown in green color.

Table 14 MAC Addresses on Fabric Interconnect - A

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN
	ORARAC1	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:CC1E	eth0	FI_A	135
			Adapter 1	NIC2	vNIC2_Private	0025:B500:CC0E	eth1	FI_A	10

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN	
1			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:CC3F	eth2	FI_A	20	
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:CC2F	eth3	FI_A	30	
	ORARAC2	Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:CC1D	eth0	FI_A	135	
			Adapter 1	NIC2	vNIC2_Private	0025:B500:CC0D	eth1	FI_A	10	
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:CC3D	eth2	FI_A	20	
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:CC2D	eth3	FI_A	30	
	2	ORARAC3	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:CC1C	eth0	FI_A	135
				Adapter 1	NIC2	vNIC2_Private	0025:B500:CC0C	eth1	FI_A	10
Adapter 4				NIC1	vNIC3_Storage_A	0025:B500:CC3C	eth2	FI_A	20	
Adapter 4				NIC2	vNIC4_Storage_B	0025:B500:CC2C	eth3	FI_A	30	
ORARAC4		Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:CC1B	eth0	FI_A	135	
			Adapter 1	NIC2	vNIC2_Private	0025:B500:CC0B	eth1	FI_A	10	
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:CC3B	eth2	FI_A	20	
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:CC2B	eth3	FI_A	30	

After replacing Fabric Interconnect B Switch to the new Fabric Interconnect Switch, the respective blades (ORARAC1 and ORARAC2) on chassis 1 and (ORARAC3 and ORARAC4) on chassis 2 will route the MAC addresses and its VLAN to Fabric Interconnect B as shown below.

The table below shows details of MAC address, VLAN information and Server connections for Cisco UCS Fabric Interconnect – A Switch.

Table 15 MAC Addresses on Fabric Interconnect – A

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN
	ORARAC1	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:C C1E	eth0	FI_A	135

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN
1			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C3F	eth2	FI_A	20
	ORARAC2	Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:C1D	eth0	FI_A	135
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C3D	eth2	FI_A	20
2	ORARAC3	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:C1C	eth0	FI_A	135
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C3C	eth2	FI_A	20
	ORARAC4	Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:C1B	eth0	FI_A	135
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C3B	eth2	FI_A	20

The table below shows details of MAC address, VLAN information and Server connections for Cisco UCS Fabric Interconnect B Switch.

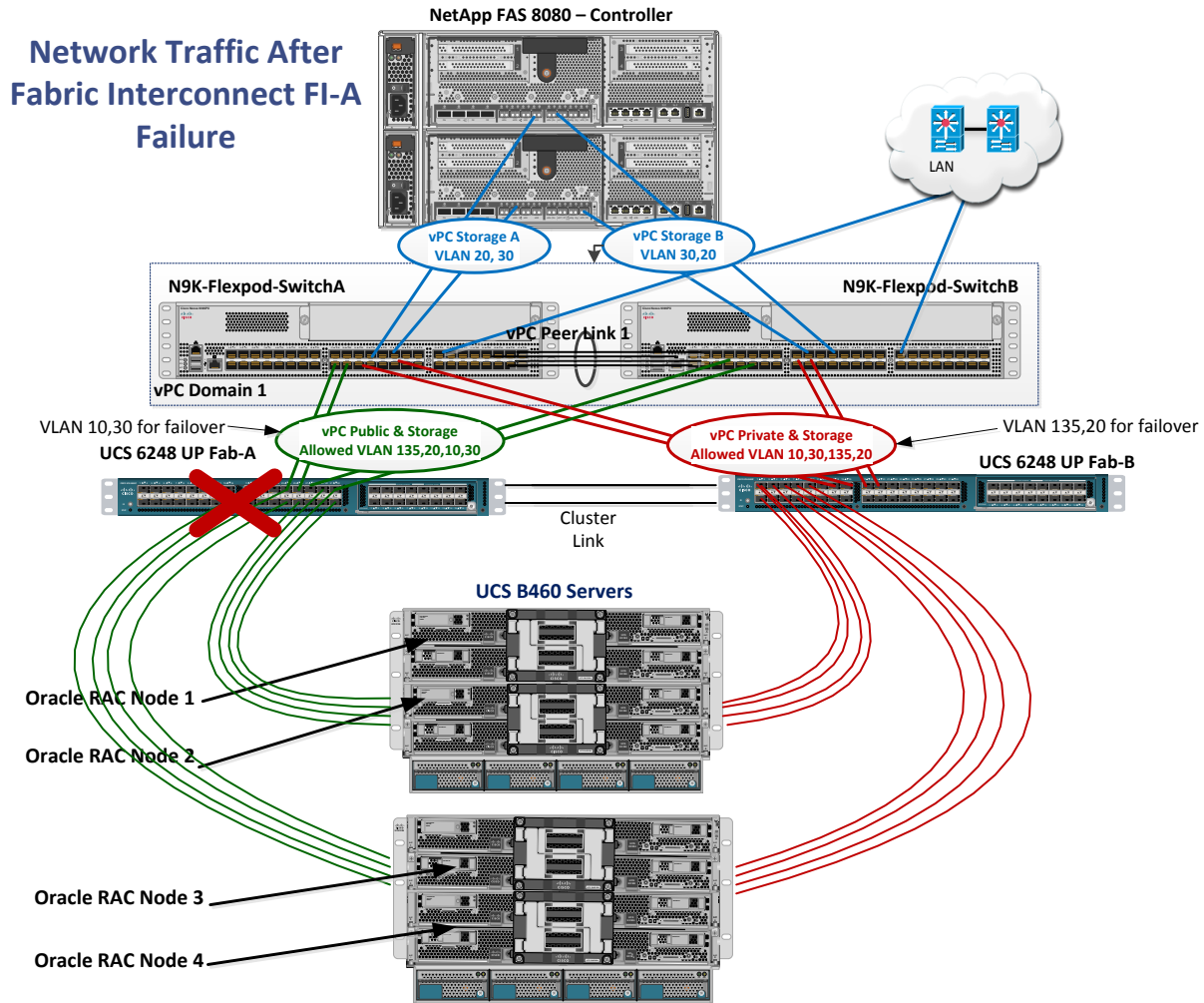
Table 16 MAC Addresses on Fabric Interconnect – B

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN
1	ORARAC1	Server 3	Adapter 1	NIC2	vNIC2_Private	0025:B500:C0E	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C2F	eth3	FI_B	30
	ORARAC2	Server 7	Adapter 1	NIC2	vNIC2_Private	0025:B500:C0D	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C2D	eth3	FI_B	30
2	ORARAC3	Server 3	Adapter 1	NIC2	vNIC2_Private	0025:B500:C0C	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C2C	eth3	FI_B	30
	ORARAC4	Server 7	Adapter 1	NIC2	vNIC2_Private	0025:B500:C0B	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C2B	eth3	FI_B	30

Test 2 - Cisco UCS 6248 Fabric-A Failure Test

Figure 67 illustrates how during fabric interconnect A switch failure, the respective blades (ORARAC1 and ORARAC2) on chassis 1 and (ORARAC3 and ORARAC4) on chassis 2 will fail over the MAC addresses and its VLAN to fabric interconnect B.

Figure 67 Fabric Interconnect - A Failure



The table below shows all the MAC address and VLAN information on Cisco UCS Fabric Interconnect B. In the table, all the switches over VLAN from Fabric Interconnect A to Fabric Interconnect B are shown in green color.

Table 17 MAC Addresses on Fabric Interconnect - B

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN
	ORARAC1	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:C C1E	eth0	FI_B	135
			Adapter 1	NIC2	vNIC2_Private	0025:B500:C C0E	eth1	FI_B	10

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN	
1			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C C3F	eth2	FI_B	20	
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C C2F	eth3	FI_B	30	
	ORARAC2	Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:C C1D	eth0	FI_B	135	
			Adapter 1	NIC2	vNIC2_Private	0025:B500:C C0D	eth1	FI_B	10	
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C C3D	eth2	FI_B	20	
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C C2D	eth3	FI_B	30	
	2	ORARAC3	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:C C1C	eth0	FI_B	135
				Adapter 1	NIC2	vNIC2_Private	0025:B500:C C0C	eth1	FI_B	10
Adapter 4				NIC1	vNIC3_Storage_A	0025:B500:C C3C	eth2	FI_B	20	
Adapter 4				NIC2	vNIC4_Storage_B	0025:B500:C C2C	eth3	FI_B	30	
ORARAC4		Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:C C1B	eth0	FI_B	135	
			Adapter 1	NIC2	vNIC2_Private	0025:B500:C C0B	eth1	FI_B	10	
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C C3B	eth2	FI_B	20	
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C C2B	eth3	FI_B	30	

After replacing Fabric Interconnect A Switch to new Fabric Interconnect Switch, the respective blades (ORARAC1 and ORARAC2) on chassis 1 and (ORARAC3 and ORARAC4) on chassis 2 will route the MAC addresses and its VLAN to Fabric Interconnect A as shown below.

The table below shows details of MAC address, VLAN information and Server connections for Cisco UCS Fabric Interconnect FI-A Switch.

Table 18 MAC Addresses on Fabric Interconnect – A

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN
	ORARAC1	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:C C1E	eth0	FI_A	135

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN
1			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C3F	eth2	FI_A	20
	ORARAC2	Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:C1D	eth0	FI_A	135
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C3D	eth2	FI_A	20
2	ORARAC3	Server 3	Adapter 1	NIC1	vNIC1_Public	0025:B500:C1C	eth0	FI_A	135
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C3C	eth2	FI_A	20
	ORARAC4	Server 7	Adapter 1	NIC1	vNIC1_Public	0025:B500:C1B	eth0	FI_A	135
			Adapter 4	NIC1	vNIC3_Storage_A	0025:B500:C3B	eth2	FI_A	20

The table below shows details of MAC address, VLAN information and Server connections for Cisco UCS Fabric Interconnect FI-B Switch.

Table 19 MAC Addresses on Fabric Interconnect – B

CHASSIS	NAME	UCS MANAGER				MAC Address	OS Address	FI	VLAN
1	ORARAC1	Server 3	Adapter 1	NIC2	vNIC2_Private	0025:B500:C0E	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C2F	eth3	FI_B	30
	ORARAC2	Server 7	Adapter 1	NIC2	vNIC2_Private	0025:B500:C0D	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C2D	eth3	FI_B	30
2	ORARAC3	Server 3	Adapter 1	NIC2	vNIC2_Private	0025:B500:C0C	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C2C	eth3	FI_B	30
	ORARAC4	Server 7	Adapter 1	NIC2	vNIC2_Private	0025:B500:C0B	eth1	FI_B	10
			Adapter 4	NIC2	vNIC4_Storage_B	0025:B500:C2B	eth3	FI_B	30

Test 3 – Nexus 9396PX Switch Failure Test



We conducted the hardware failure test on Nexus Switch A and Nexus Switch B.

During Nexus Switch B failure, the respective blades (ORARAC1 and ORARAC2) on chassis 1 and (ORARAC3 and ORARAC4) on chassis 2 will failover the MAC addresses and its VLAN to Nexus Switch A same way as Fabric Switch failure. Similarly, during Nexus Switch A failure, the respective blades (ORARAC1 and ORARAC2) on chassis 1 and (ORARAC3 and ORARAC4) on chassis 2 will failover the MAC addresses and its VLAN to Nexus Switch B same way as Fabric Switch failure.

Summary

Cisco and NetApp have partnered to deliver the FlexPod solution, which uses best-in-class storage, server, and network components to serve as the foundation for a variety of workloads, enabling efficient architectural designs that can be quickly and confidently deployed. FlexPod Datacenter is predesigned to provide agility to the large enterprise data centers with high availability and storage scalability. With a FlexPod-based solution, customers can leverage a secure, integrated, and optimized stack that includes compute, network, and storage resources that are sized, configured, and deployed as a fully tested unit running industry standard applications such as Oracle Database 12c RAC over D-NFS (Direct NFS).

The following factors make the combination of Cisco UCS with NetApp storage so powerful for Oracle environments:

- Cisco UCS stateless computing architecture provided by the Service Profile capability of Cisco UCS allows for fast, non-disruptive workload changes to be executed simply and seamlessly across the integrated UCS infrastructure and Cisco x86 servers.
- Cisco UCS, combined with NetApp's highly-scalable FAS storage system with NetApp's Clustered Data ONTAP®, provides the ideal combination for Oracle's unique, scalable, and highly available FAS technology.
- Hardware level redundancy for all major components using Cisco UCS and NetApp availability features.

FlexPod is a flexible infrastructure platform composed of pre-sized storage, networking, and server components. It's designed to ease your IT transformation and operational challenges with maximum efficiency and minimal risk.

FlexPod differs from other solutions by providing:

- Integrated, validated technologies from industry leaders and top-tier software partners.
- A single platform built from unified compute, fabric, and storage technologies, allowing you to scale to large-scale data centers without architectural changes.
- Centralized, simplified management of infrastructure resources, including end-to-end automation.
- A flexible Cooperative Support Model that resolves issues rapidly and spans across new and legacy products.

Appendix

Cisco Nexus 9396PX Running Configuration



Only relevant fragments are shown.

```
N9k-Flexpod-SwitchA# show run

!Command: show running-config
!Time: Thu Sep  3 23:38:45 2015

version 7.0(3)I1(2)
switchname N9k-Flexpod-SwitchA
policy-map type network-qos jumbo
  class type network-qos class-default
    mtu 9216
vdc N9k-Flexpod-SwitchA id 1
  limit-resource vlan minimum 16 maximum 4094
  limit-resource vrf minimum 2 maximum 4096
  limit-resource port-channel minimum 0 maximum 511
  limit-resource u4route-mem minimum 248 maximum 248
  limit-resource u6route-mem minimum 96 maximum 96
  limit-resource m4route-mem minimum 58 maximum 58
  limit-resource m6route-mem minimum 8 maximum 8

cfs eth distribute
feature interface-vlan
feature hsrp
feature lacp
feature vpc
feature lldp
...
```

```
...
ip domain-lookup
system qos
  service-policy type network-qos jumbo
...
...
vlan 1,10,20,30,135
vlan 10
  name ORACLE_RAC_PRIVATE_TRAFFIC
vlan 20
  name Storage-A-vNIC3
vlan 30
  name Storage-B-vNIC4
vlan 135
  name ORACLE_RAC_PUBLIC_TRAFFIC

spanning-tree port type network default
vrf context management
  ip route 0.0.0.0/0 10.29.135.1
vpc domain 1
  role priority 10
  peer-keepalive destination 10.29.135.4
  auto-recovery

interface Vlan1
  no shutdown

interface Vlan20
  description Storage-A-Subnet
  no shutdown
  no ip redirects
  ip address 10.20.20.2/24
```

```
hsrp version 2
hsrp 20
  preempt
  ip 10.20.20.1
```

```
interface Vlan30
  description Storage-B-Subnet
  no shutdown
  no ip redirects
  ip address 10.30.30.2/24
  hsrp version 2
  hsrp 30
    preempt
    ip 10.30.30.1
```

```
interface Vlan135
  no shutdown
  ip address 10.29.135.2/24
  hsrp version 2
  hsrp 135
```

```
interface port-channel1
  switchport mode trunk
  switchport trunk allowed vlan 1,10,20,30,135
  spanning-tree port type network
  vpc peer-link
```

```
interface port-channel17
  description Port-Channel FI-A
  switchport mode trunk
  switchport trunk allowed vlan 1,10,20,30,135
  spanning-tree port type edge trunk
```

```
mtu 9216
```

```
vpc 17
```

```
interface port-channel19
```

```
switchport mode trunk
```

```
switchport trunk allowed vlan 1,10,20,30,135
```

```
spanning-tree port type edge trunk
```

```
mtu 9216
```

```
vpc 19
```

```
interface port-channel20
```

```
description Port-channel multimode VIF from FAS8080-CNTRL-A-10G
```

```
switchport mode trunk
```

```
switchport trunk allowed vlan 20,30
```

```
spanning-tree port type edge trunk
```

```
mtu 9216
```

```
vpc 20
```

```
interface port-channel30
```

```
description Port-channel multimode VIF from FAS8080-CNTRL-B-10G
```

```
switchport mode trunk
```

```
switchport trunk allowed vlan 20,30
```

```
spanning-tree port type edge trunk
```

```
mtu 9216
```

```
vpc 30
```

```
interface Ethernet1/1
```

```
description Peer Link connected to N9KB-Eth1/1
```

```
switchport mode trunk
```

```
switchport trunk allowed vlan 1,10,20,30,135
```

```
channel-group 1 mode active
```

```
interface Ethernet1/2
  description Peer Link connected to N9KB-Eth1/2
  switchport mode trunk
  switchport trunk allowed vlan 1,10,20,30,135
  channel-group 1 mode active
```

```
interface Ethernet1/3
  description Peer Link connected to N9KB Eth1/3
  switchport mode trunk
  switchport trunk allowed vlan 1,10,20,30,135
  channel-group 1 mode active
```

```
interface Ethernet1/4
  description Peer Link connected to N9KB Eth1/4
  switchport mode trunk
  switchport trunk allowed vlan 1,10,20,30,135
  channel-group 1 mode active
```

...

...

```
interface Ethernet1/17
  description Connection FI-A P17 to N9KA P17
  switchport mode trunk
  switchport trunk allowed vlan 1,10,20,30,135
  spanning-tree port type edge trunk
  mtu 9216
  channel-group 17 mode active
```

```
interface Ethernet1/18
  description Connection FI-A P18 to N9KA P18
  switchport mode trunk
  switchport trunk allowed vlan 1,10,20,30,135
  spanning-tree port type edge trunk
```

```
mtu 9216
channel-group 17 mode active

interface Ethernet1/19
description Connection FI-B P17 to N9KA P19
switchport mode trunk
switchport trunk allowed vlan 1,10,20,30,135
mtu 9216
channel-group 19 mode active

interface Ethernet1/20
description Connection FI-B P18 to N9KA P20
switchport mode trunk
switchport trunk allowed vlan 1,10,20,30,135
mtu 9216
channel-group 19 mode active
...
...
interface Ethernet1/31
description Connection to NetApp CTRL-A e0e
switchport mode trunk
switchport trunk allowed vlan 20,30
spanning-tree port type edge trunk
mtu 9216
channel-group 20 mode active

interface Ethernet1/32
description Connection to NetApp CTRL-B e0e
switchport mode trunk
switchport trunk allowed vlan 20,30
spanning-tree port type edge trunk
mtu 9216
```

```
channel-group 30 mode active
...
...
interface mgmt0
    vrf member management
    ip address 10.29.135.3/24
line console
line vty
boot nxos bootflash:/n9000-dk9.7.0.3.I1.2.bin
```

Sysctl.conf settings

```
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 0

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Do not accept source routing
net.ipv4.conf.default.accept_source_route = 0

# Controls the System Request debugging functionality of the kernel
kernel.sysrq = 0

# Controls whether core dumps will append the PID to the core filename.
# Useful for debugging multi-threaded applications.
kernel.core_uses_pid = 1
```



```
# Controls the use of TCP syncookies
net.ipv4.tcp_syncookies = 1

# Disable netfilter on bridges.
#net.bridge.bridge-nf-call-ip6tables = 0
#net.bridge.bridge-nf-call-iptables = 0
#net.bridge.bridge-nf-call-arptables = 0

# Controls the default maximum size of a message queue
kernel.msgmnb = 65536

# Controls the maximum size of a message, in bytes
kernel.msgmax = 65536

# Controls the maximum shared segment size, in bytes

# Controls the maximum number of shared memory segments, in pages
##### Oracle 12C settings and NFS parameters Metalink note 1519875.1 #####
#net.core.rmem_default = 4194304
net.core.rmem_default = 134217728
net.core.rmem_max = 134217728
net.core.wmem_default = 134217728
#net.core.wmem_default = 4194304
net.core.wmem_max = 134217728
##### Additional NFS settings #####
net.ipv4.tcp_rmem = 4096 87380 134217728
net.ipv4.tcp_wmem = 4096 65536 134217728
net.ipv4.tcp_window_scaling = 1
net.ipv4.tcp_syncookies = 0
net.ipv4.tcp_timestamps = 0
net.ipv4.tcp_sack = 0
net.ipv4.tcp_no_metrics_save = 1
```

```
#net.ipv4.tcp_moderate_rcvbuf = 0
net.core.netdev_max_backlog = 300000
net.ipv4.tcp_moderate_rcvbuf = 1

# oracle-rdbms-server-12cR1-preinstall setting for fs.file-max is 6815744
fs.file-max = 6815744

# oracle-rdbms-server-12cR1-preinstall setting for kernel.sem is '250 32000 100 128'
kernel.sem = 250 32000 100 128

# oracle-rdbms-server-12cR1-preinstall setting for kernel.shmmni is 4096
kernel.shmmni = 4096

# oracle-rdbms-server-12cR1-preinstall setting for kernel.shmall is 1073741824 on
x86_64
kernel.shmall = 1073741824

# oracle-rdbms-server-12cR1-preinstall setting for kernel.shmmax is 4398046511104 on
x86_64
kernel.shmmax = 4398046511104

# oracle-rdbms-server-12cR1-preinstall setting for net.core.rmem_default is 262144

# oracle-rdbms-server-12cR1-preinstall setting for net.core.rmem_max is 4194304

# oracle-rdbms-server-12cR1-preinstall setting for net.core.wmem_default is 262144

# oracle-rdbms-server-12cR1-preinstall setting for net.core.wmem_max is 1048576

# oracle-rdbms-server-12cR1-preinstall setting for fs.aio-max-nr is 1048576
fs.aio-max-nr = 1048576
```

```
# oracle-rdbms-server-12cR1-preinstall setting for net.ipv4.ip_local_port_range is  
9000 65500
```

```
kernel.panic_on_oops = 1
```

```
vm.nr_hugepages=105000
```

About Authors

Tushar Patel, Principle Engineer, Data Center Group, Cisco Systems, Inc.

Tushar Patel is a Principle Engineer for the Cisco Systems Data Center group. Tushar has nearly eighteen years of experience in database architecture, design and performance. Tushar also has strong background in Intel x86 architecture, converged systems, storage technologies and virtualization. He has helped a large number of enterprise customers evaluate and deploy mission-critical database solutions. Tushar has presented to both internal and external audiences at various conferences and customer events.

Niranjan Mohapatra, Technical Marketing Engineer, Data Center Group, Cisco Systems, Inc.

Niranjan Mohapatra is a Technical Marketing Engineer for Cisco Systems Data Center group and specialist on Oracle RAC RDBMS. He has over 16 years of extensive experience with Oracle RDBMS and associated tools. Niranjan has worked as a TME and a DBA handling production systems in various organizations. He holds a Master of Science (MSc) degree in Computer Science and is also an Oracle Certified Professional (OCP) and Storage Certified Professional.

John Elliott, Senior Technical Marketing Engineer, Data Fabric Eco Systems Solutions Group, NetApp

John Elliott is a Senior Technical Marketing Engineer for the NetApp Data Fabric Eco Systems Solutions group. John has 14 years of experience in storage performance, storage architecture, storage protocols, and database administration and performance. He has also authored a number of NetApp technical documents related to Oracle databases and NetApp storage.

Acknowledgements

- Hardik Vyas, Data Center Group, Cisco Systems, Inc.