ORACLE UNIVERSITY

Oracle SPARC M7 Server System Administration

Student Guide
D97486GC10
Edition 1.0 | November 2016 | D98575

Learn more from Oracle University at **oracle.com/education/**

ORACLE

Contents

1

Introduction

Course Objectives

After completing this course, you should be able to:

- List the functions and features of the SPARC M7-8 and M7-16 Servers
- Identify their components and architecture
- Install and configure the M7 Servers
- Upgrade the firmware and software
- Perform administration tasks
- Configure LDoms
- Perform basic troubleshooting of the M7 Servers

Course Map

The following course map enables you to see what you have accomplished and where you are going with reference to the course goals:

- Les 01 • Introduction
- Les 02 • Platform Configuration and Administration Using ILOM
- Les 03 • Domain Configuration
- Les 04 • Logical Domain Configuration
- Les 05 • Data Collection and Fault Analysis

Topics Not Covered

This course does not cover the following topics. These topics are covered in other courses offered by Oracle University.

- OS Administration (See the website at: http://education.oracle.com/pls/web_prod-plq-dad/ou_product_category.getPage?p_cat_id=434.)
  - *Oracle Solaris 11 System Administration*
  - *Oracle Solaris 11 Advanced System Administration*
- Server and Storage Administration
  - Storage Administration (See the website at http://education.oracle.com/pls/web_prod-plq-dad/ou_product_category.getFamilyPage?p_family_id=357.)
  - Server Administration (See the website at http://education.oracle.com/pls/web_prod-plq-dad/ou_product_category.getPage?p_cat_id=301.)
- Database Administration
  - Oracle Database 12*c* (See the website at http://education.oracle.com/pls/web_prod-plq-dad/ou_product_category.getPage?p_cat_id=385.)

Additional Resources for Installation and Maintenance

- Oracle Technology Network
  http://otn.oracle.com
- Oracle by Example (OBE)
  http://www.oracle.com/technetwork/index.html
- My Oracle Support
  http://support.oracle.com
- Oracle Learning Library
  https://apexapps.oracle.com/pls/apex/f?p=44785:OLL_HOME
- My Oracle Support Community
  http://communities.oracle.com

Lesson Objectives

After completing this lesson, you should be able to:

- List the main components that comprise the M7-8 and M7-16 servers
- Provide a comparison with previous products
- Define the hardware features
- Define the software features
- Describe the M7-8 and M7-16 servers
- Describe the functions and features of the M7-8 and M7-16 servers
- Describe the RAS features of these Servers with comparison to previous products

ORACLE

Additional Resources

The following references provide additional information about the topics described in this lesson:

- Technology Portals on OTN:
  https://www.oracle.com/servers/sparc/m7-8/index.html
  https://www.oracle.com/servers/sparc/m7-16/index.html
- Whitepapers:
  https://www.oracle.com/servers/sparc/resources.html

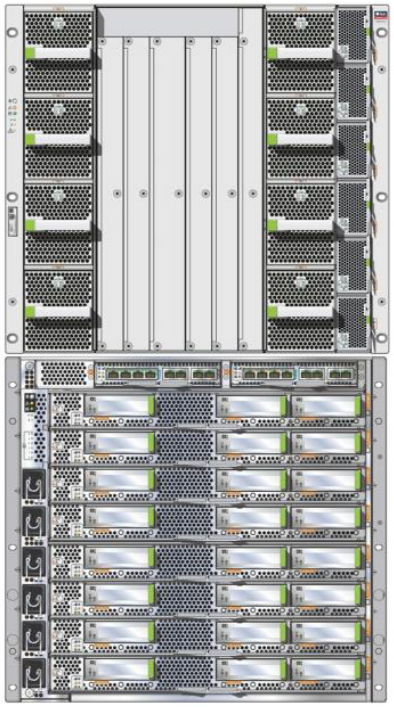
Introduction

The M7-8 and M7-16 Servers are data center servers that were designed with the highest reliability and availability features. They have a high redundancy of components and perform numerous health checks.

Component/Function	Characteristics
Processor	**M7-8:** 4 to 8 SPARC M7 processors, each with 32 cores and 8 threads/core
	M7-16: 8 to 16 SPARC M7 processors, each with 32 cores and 8 threads/core
Memory	16 DIMM slots per processor, 16GB or 32GB DIMMs
I/O expansion	**M7-8:** 12 to 24 low-profile PCIe Generation 3 (PCIe3) card slots
	M7-16: 24 to 48 low-profile PCIe Generation 3 card slots
Storage	One or more optional flash accelerator PCIe cards
Service processors	Two redundant SPs. The M7-16 has an additional four redundant SPPs.
	The SPs and SPPs are used to monitor and control the server remotely.

M7-8: Overview

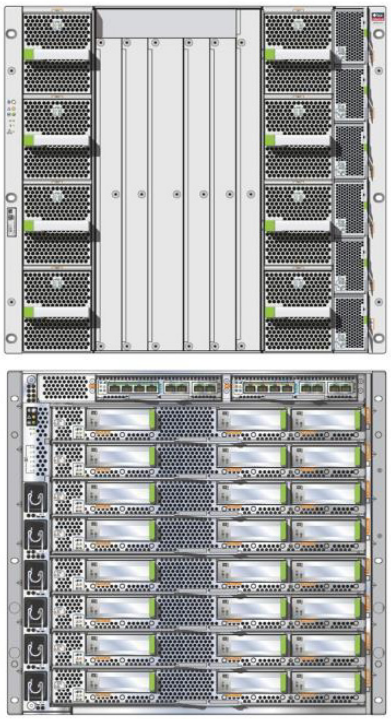
- 10 RU chassis, 31" deep (approximate)
- 8-socket M7 (4.133GHz, 32 Cores, 8 Threads/Core) Total 2048 Threads
- Single or dual physical partitions (PDoms)
- Eight CMIOU Modules. Each CMIOU contains:
  - 1x M7 processor
  - 16x DDR4 Memory DIMM Slots total, supporting up to 512GB
  - 3 PCIe Gen 3 x16 Hot Plug LP "Carrier" slots
- 6X Hot-plug N+N redundant PSUs (3KW each @ 200VAC-240VAC)
- 8X Hot-plug fan modules
- ILOM3.X with advanced power management features



M7-8 Partitions

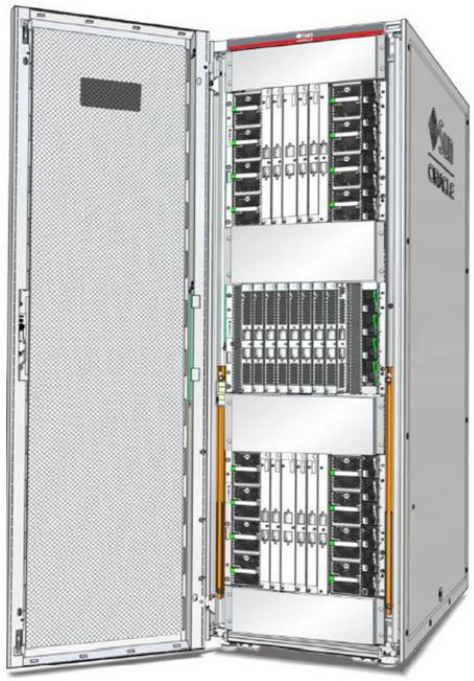
The M7-8 ships in two different configurations:

- One configuration is as a single hard partition, with the maximum number of PDoms is 1
- The other configuration is as two physical partitions (that is, PDoms). The SPARC M7-8 with two PDom servers will use the same chassis as the M7-8 server, but it will have different front interconnects that will divide the eight CMIOUs into two 4 CMIOU halves. It also has a different service processor configuration.



M7-16: Overview

- 16-socket M7 (4.133GHz, 32 Cores, 8 Threads/Core) Total 4096 Threads
- One to four physical partitions (PDoms)
- Sixteen CMIOU Modules. Each CMIOU contains:
  - 1x M7 processor
  - 16x DDR4 Memory DIMM Slots total, supporting up to 512GB
  - 3 PCIe Gen 3 x16 Hot Plug LP "Carrier" slots
- 12X Hot-plug N+N redundant PSUs (3KW each @ 200VAC-240VAC), six per CMIOU chassis
- 16X Hot-plug fan modules, eight per CMIOU chassis
- 1Switch chassis
- ILOM3.X with advanced power management features



Key Differences from Earlier M-Series

	M6-32	M7-16
SP	SP, running ILOM. Each DCU has its own SP Proxy (SPP).	Each CMIOU chassis has dual SPPs and the switch chassis has dual SPs all running ILOM.
Domains	Physical domains: Up to 4	Physical domains: Up to 4
Memory Clock	DDR3 at 1066MHz	DDR4 at 2133MHz
Lowest Latency	158 ns (Bounded Domain), 160 ns (Regular Domain)	132 ns
I/O	PCIe Gen 3 (PCIe switch limits to Gen 2 bandwidth)	PCIe Gen 3
Smallest PDom	2 CPUs + 64 DIMMs (1 CMU board)	1 CPU + 16 DIMMs (1 CMIOU board)
Internal network	10/100 Ethernet	10/100 Ethernet
Clock and Interconnect	Hot-plug replacement of failed clock and SBB boards	Redundant (one clock per M7 module); Redundant Clock Synthesizers (one per CMIOU/Switch)

The M6-32 has an internal midplane. The M7-8 servers have Internal Interconnects and the M7-16 server has External Interconnect consisting of wire trusses.

Key Differences from Earlier M-Series

	M6-32	M7-16
Power Feed	3-phase, DPF standard Cable is orderable.	Single or 3-phase, DPF standard Cable is orderable.
Airflow	Front to back	Front to back
Base I/O Card	Dedicated slots just for Base I/O card	No internal storage
KVMS	Must use remote media of rKVMS	Must use remote media of rKVMS
Cable Routing	Top or bottom	Top or bottom

Terminology

Term	Definition
CPU memory I/O board Unit (CMIOU)	Board equipped with one CPU module, up to 16 DIMMs of memory, and 3 PCIe Gen 3 x 16 Hot-plug LP slots
Chip Multi-Processor (CMP)	Another term for multi-core processor
Buffer on Board (BoB)	An ASIC that interfaces between the DIMMs and the MCU on the M7. Each BoB can control two DIMMs, which is how we reach 16DIMMs total for an M7.
Memory Controller Unit (MCU)	Each M7 processor has four MCUs and each MCU communicates with two BoBs.
Domain configurable unit (DCU)	A grouping of up to 4 CMIOUs, controlled by its own SP Proxy in an M7-16 server
Physical domain (PDom)	An electrically fault-isolated hard domain that comprises one or more DCUs
Hypervisor	A small firmware layer that provides a stable virtualized machine architecture to which an operating system can be written

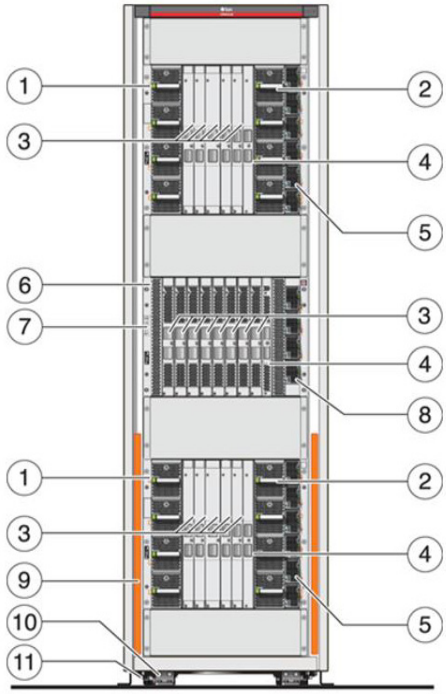
It is very important to understand the terminology used in these servers before going any further. Some of the terms may be familiar from other server lines.

The M7-8 and M7-16 servers consist of CPU, memory, and I/O, which reside on CMIOU boards. Each CMIOU board contains one CPU module and 16 Dual Inline Memory Module (DIMM) slots, as well as 3 PCIe Gen 3 Hot Plug LP slots.

Domain configurable units (DCUs) are groups of four CMIOUs in the M7-16, connected by the local coherency interconnect. A physical domain consists of one or more DCUs.

Components: M7-16 Front Cabinet

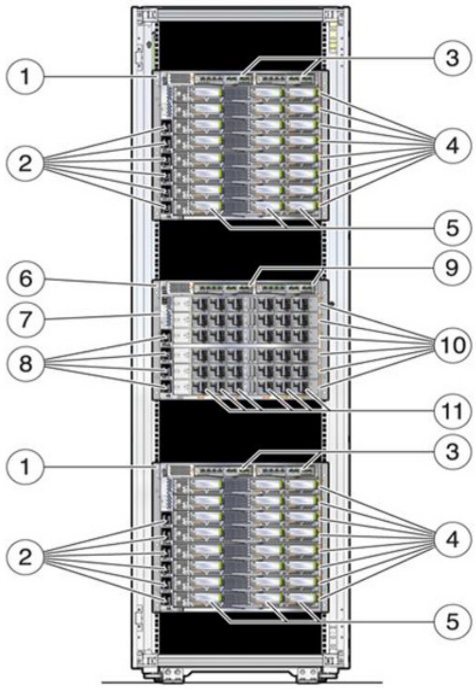
1. CMIOU chassis
2. CMIOU fan modules
3. Interconnects
4. SP interconnects
5. CMIOU power supplies
6. Switch chassis
7. Front indicator panel
8. Switch power supplies
9. Anti-tilt bars
10. Mounting brackets
11. Leveling feet



The M7-8 server consists of one CMIOU chassis. The M7-16 server consists of two CMIOU chassis with a switch chassis connecting them.

Components: M7-16 Rear Cabinet

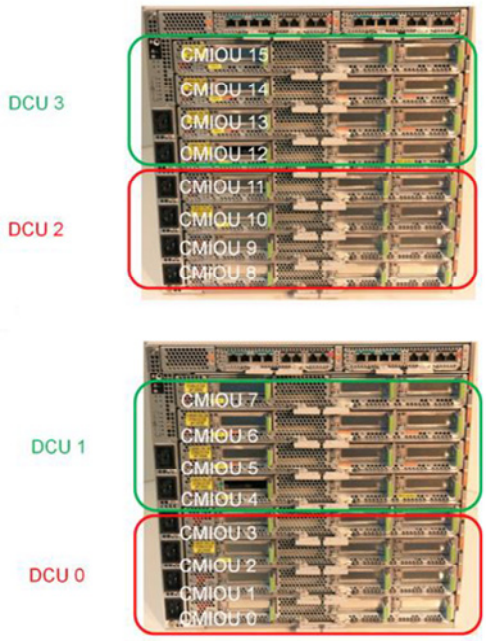
1. CMIOU chassis
2. CMIOU chassis AC inputs
3. SPPs
4. CMIOUs
5. PCIe hot-plug carrier for low-profile PCIe cards
6. Switch chassis
7. Rear indicator panel
8. Switch chassis AC inputs
9. SPs
10. Switch units
11. Switch unit fan modules



Note: What is labeled as SPPs (3) in the diagram for M7-16 will be SPs in M7-8.

M7-16 DCUs

A physical domain (PDom) can be made from one or more DCUs.



A SPARC M7-16 server is divided into four configurable units called DCUs. These DCUs have four CMIOUs each. Two of the DCUs are in the top CMIOU chassis, and two are in the bottom CMIOU chassis. You can configure the DCUs in up to four PDomains.

Each CMIOU chassis also has two service processor proxies. SPP2 and SPP3 are in the top chassis, and SPP0 and SPP1 are in the bottom chassis. These SPPs have two service processor modules each. To achieve redundancy, each SPM (SPM0 and SPM1) on an SPP is assigned to a different DCU. SPM0 on each SPP manages one DCU in the CMIOU chassis, whereas SPM1 on each SPP manages the other DCU in the chassis.

Service Processors on CMIOU and Switch Chassis

There are two redundant SPs in a SPARC M7-8 server: SP0 and SP1.

In a SPARC M7-16 server, the SPs in each CMIOU chassis are referred to as SPPs. SPP2 and SPP3 are in the top CMIOU chassis, and SPP0 and SPP1 are in the bottom CMIOU chassis. A SPARC M7-16 server also has two SPs in the switch chassis. The SPPs in the CMIOU chassis manage DCU activity, and the SPs in the switch chassis manage system activity.

Service Processor

The service processor (SP):

- Runs Integrated Lights Out Management (ILOM)
- Is redundant in the server
- Provides primary platform configuration and management
- Works with the SPPs in the M7-16 server to configure and monitor the components in each DCU

Service Processor Software (ILOM)

ILOM is the software that runs on the service processor and it:

- Looks and behaves just like ILOM on other platforms
- Contains a simple (user-visible) set of extensions to support physical domains
- Has extensions to support service processor proxies and redundant hot-swappable service processors
  - Has minimal impact on user experience

ILOM on the M7 servers has extensions to support Enterprise features.

Oracle ILOM: Key Functions

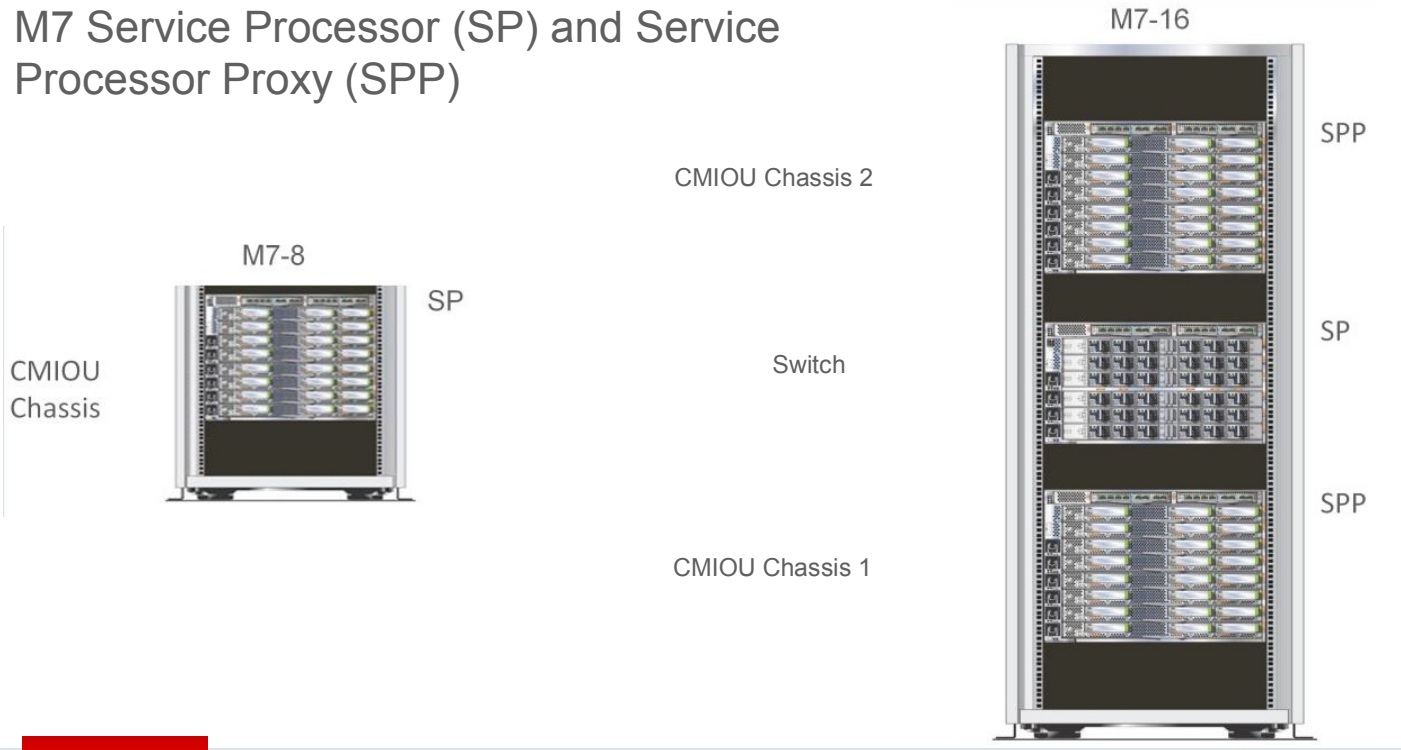
Key ILOM functions include:

- Management Interfaces
  - Command-line interface (CLI), browser user interface (BUI), intelligent platform management interface (IPMI), simple network management protocol (SNMP)
- Firmware updates
- Remote host management
- Inventory and component management
- System monitoring and alert/fault management
- User account management
- Power consumption management

Oracle Integrated Lights Out Manager (Oracle ILOM) is the system management firmware that is preinstalled on the M7 Servers. It enables you to actively manage and monitor components installed in your server. Oracle ILOM provides a browser-based interface and a command-line interface, as well as SNMP and IPMI interfaces.

The Oracle ILOM SP runs independently of the server and regardless of the server power state as long as AC power is connected to the server. When you connect the server to AC power, the ILOM service processor immediately starts up and begins monitoring the server. All environmental monitoring and control are handled by Oracle ILOM.

M7 Service Processor (SP) and Service Processor Proxy (SPP)

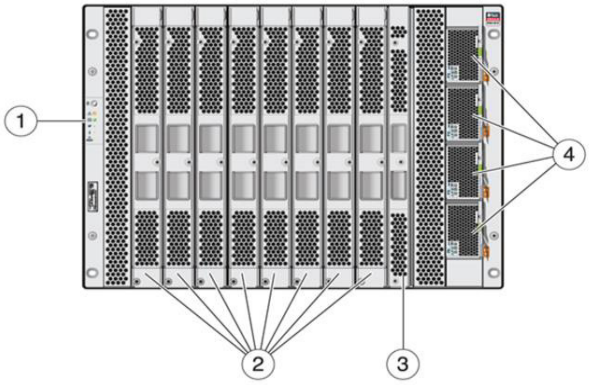


In the M7-8 servers, each CMIOU chassis has two service processors.

In the M7-16 server, the service processors reside on the Switch chassis. Each CMIOU chassis has two service processor proxies. SPP2 and SPP3 are in the top chassis, and SPP0 and SPP1 are in the bottom chassis. These SPPs have two service processor modules each. To achieve redundancy, each SPM (SPM0 and SPM1) on an SPP is assigned to a different DCU. SPM0 on each SPP manages one DCU in the CMIOU chassis, whereas SPM1 on each SPP manages the other DCU in the chassis.

Components: Switch Chassis - Front

1. Front indicator panel
2. External interconnects
3. SP internal interconnect
4. Power supplies

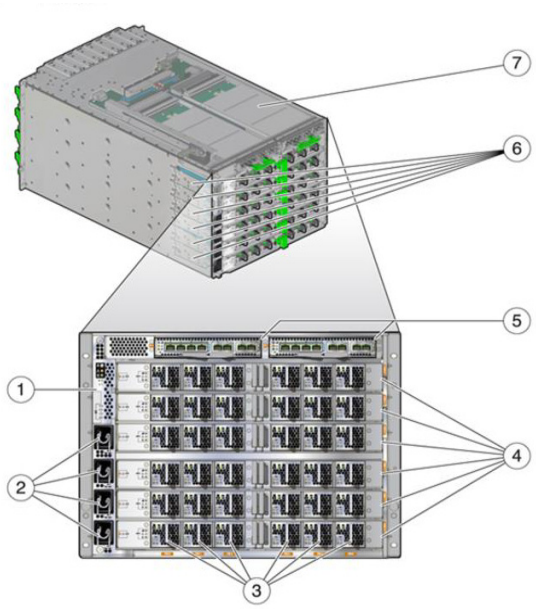


The M7-16 server consists of two CMIOU chassis with a switch chassis connecting them.

In SPARC M7-16 servers, switch units are part of the scalability feature that allows a PDomain to control more than one DCU. Switch units are configured to work together as a single unit.

Components: Switch Chassis - Rear

1. Rear indicator panel
2. AC inlets
3. Fan modules
4. Switch units
5. SPs
6. PDECBs
7. SP tray



Remote Keyboard / Video / Mouse / Storage (rKVMS)

- rKVMS is created by the domain as a USB keyboard, mouse, and storage (as well as graphics) console window.
- The KVMS software that is preinstalled on these servers allows for both video-redirection and serial-redirection connections to the Oracle Solaris OS.
- Only the serial-redirection connection supports the Oracle Solaris console.
- Video redirection provides a graphical display to the Oracle Solaris OS.

Oracle ILOM Remote System Console Plus is a Java application that enables you to remotely redirect and control the following devices on the host server. This group of devices is commonly abbreviated as KVMS.

- Keyboard
- Video display
- Mouse
- Storage devices or images (CD/DVD)

Virtualization

- The M7-8 and M7-16 Servers have a high degree of virtualization including:
  - Physical domains
    - Granularity of physical domains is a single DCU.
  - Logical domains
    - Includes support of LDoms within physical domains
  - Oracle Solaris Zones for OS virtualization
- Oracle Enterprise Manager Ops Center provides an administrator-friendly integration of these different virtualization levels.

What Is a Physical Domain?

- There can be up to four dynamic physical domains (PDoms or PDomains) in the M7-16 server, two static PDoms in the M7-8 with two PDoms, and one static PDom in the M7-8 server.
- Each PDom operates like an independent server that has full hardware isolation from other PDoms in the chassis.
- Domain configurable units (DCUs) are the building blocks of PDoms.
- Each PDom is represented as ***PDomain_x*** or ***HOSTx*** in Oracle ILOM where *x* ranges from 0 through 3 (PDomain_0, PDomain_1, PDomain_2, PDomain_3).

`/Servers/PDomains/PDomain_x/HOST` is the equivalent of `/HOSTx`.

Logical Domains

- **Logical domains (LDoms)** is the server virtualization and partitioning technology from Oracle that has been rebranded as Oracle VM Server for SPARC.
- Each domain is a virtualized environment with a reconfigurable subset of hardware resources and an OS that can be started, stopped, and rebooted independently of the host system or any other domains.

Hypervisor and Logical Domains

- A *logical domain* (LDom) is a virtual machine composed of a discrete logical grouping of resources.
- Each LDom runs its instance of Oracle Solaris.
- Each LDom can be created, destroyed, reconfigured, and rebooted independently.
- The Hypervisor enforces the partitioning of the server's resources, and the OS and applications running in those partitions (that is, LDoms).
- The Hypervisor allocates a subset of the overall CPU, memory, and I/O resources of a server to a given logical domain.

Hypervisor implements the software component of the sun4v virtual machine, providing low overhead hardware abstraction. It enforces hardware and software resource access restrictions for guests, including inter-guest communication, to provide isolation and security. It also performs initial triage and correction of hardware errors.

SPARC M7 System RAS: Overview

- Designed to minimize part count and operating temperature to enhance reliability
- End-to-end data protection, detecting and correcting errors throughout server
- Processor and memory protection
  - CPU core and thread off-lining
  - Memory with ECC, DRAM Extended ECC, page retirement, and lane
- Major components redundant and hot-pluggable
  - Fans and Power Supplies
- Fault Management Architecture (FMA) support on ILOM

FMA is part of the ILOM.

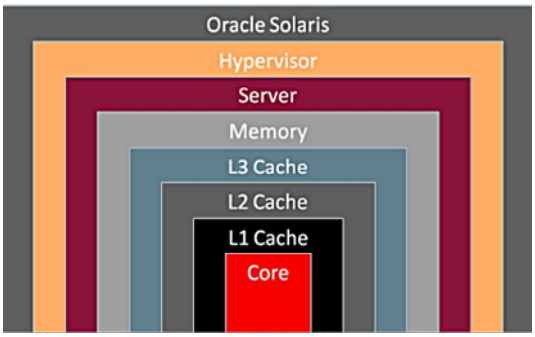
M7 Systems RAS

M7 Processor
- L1 cache tag, status and data
  - Parity protection
  - Retry on error
- L2 & L3 cache data
  - SEC/DED protection
  - Inline correction
  - Cache-line sparing
- L2 & L3 cache status and directory
  - SEC/DED protection
  - Inline correction
- Architectural registers L2 Cache
  - SEC/DED protection
  - Precise trap
  - Hypervisor correction/retry
- Message retry in hardware
- Dynamic processor core deconfiguration
- Independently managed power for each quadrant of the processor
- Link-level RAS:
  - Automatic frame retry, link retrain
  - Link-level multipathing
  - Single-lane failover
- Dynamic voltage and frequency scaling

Memory
- SDRAM Soft Errors
  - ECC Protection and Correction
  - Extended ECC Protection
  - 4-bit Correction
  - DIMM Sparing
- Channel Interconnect
  - CRC protection/Message Retry
  - Lane Sparing

S11 FMA
- Diagnosis engine on SP & Oracle Solaris
- Auto reconfigure on failure
- Soft error rate discrimination (SERD)
- Bad page retirement
- OS & SP watchdogs
- FMA component hot-upgradeable

Hypervisor
- Enables software partitioning (LDoms), virtualization, and failure containment
- Dynamic PCIe bus assignment
- Processor support for error clearing, correction and collection



New to M7

Processor Features

L1Cache Protection

- Because the L1 caches are clean, there is no need for the stronger protection of ECC that is used on the larger caches. So parity protection suffices.
- If a parity error is detected, the affected cacheline is invalidated and a new copy is fetched from the L2 cache. The new fetch uses a bypass path so it is guaranteed to not hit the same location again, if by chance there is a persistent failure there.

L2Cache Protection

- Because L2 can have modified (that is, dirty) data, ECC is used for protection. ECC provides Single Error Correct (SEC)/Double Error Detect (DED) coverage so that data can always be recovered in case of a single-bit error. Data is arranged in the physical SRAM of the cache to ensure that the cosmic rays and alpha particles affect only a single bit of data covered by the same ECC checkbits.
- For performance reasons, error correction is not done in the critical access path of the cache. So a detected error will cause a trap to Hypervisor (part of the system firmware), which will then flush the line from the cache and allow the user process to resume. Again, a bypass path ensures forward progress in the presence of a persistent failure.
- If the system firmware determines that a specific location in the cache is generating a lot of errors, it will update the state of the cache so that that location is no longer used. This process is called either cacheline "retire" or cacheline "sparing."

L3Cache Protection

- Much the same as the L2 but with inline correction, which means that if a cacheline is loaded to the L2 from the L3, the data will be corrected when it's placed in the L2 if it contains an error when read from the L3.

Status and Directory Protection

- The Status and Directory arrays in both the L2 and L3, which are used for maintaining coherence, are protected by ECC and have automatic correction built into the HW, to ensure forward progress.

Register File Protection

- The architectural registers that are used by executing processes are also protected by ECC, so that any form of upset will not cause loss of data. The HW does not automatically correct, but a detected error will cause a trap to Hypervisor, which will then correct the data and allow the process to resume.

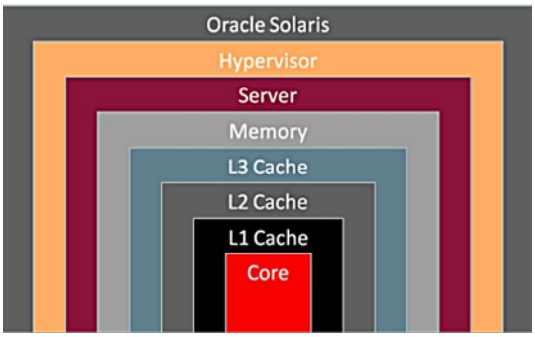
M7 Systems RAS

System
- Diagnosis to the level of a field replaceable unit (FRU) on first fault
- Redundant hot-serviceable service processors (SPs) and service processor proxies (SPPs) with automatic failover
- Plesiochronous Clocking

System I/O
- PCIe end-to-end CRC
- PCIe link retry
- Hot-pluggable low-profile PCIe cards

Central Directory and Switch
- SEC/DED protection with inline correction
- Physical domain isolation
- CRC protected System Interconnect with message retry and lane sparing
- Deconfigurable directory chips, no loss of functionality, minimized bandwidth loss

Power and Cooling
- Advanced power management
- Redundant hot-swappable fans and fan modules
- Redundant hot-swappable power supplies
- Dual-grid power
- Redundant CPU core power supply



New to M7

System Infrastructure Features

There are two Service Processors (SPs) in the system, which are used in an active-standby relationship. So only one of the two SPs is the “active” SP. But the other SP can take over as the “active” SP if it detects that the first active SP is no longer functioning properly. This is done with no loss of functionality to the rest of the system.

Memory Subsystem Features

DRAM Protection

- Main memory, which resides in DRAM devices, is protected by ECC, but it is a much stronger version of ECC than is used elsewhere in the system.
- When using x4 DRAM devices (which means each device provides 4 bits of data on every access cycle), the ECC used for memory can correct errors even in the presence of a complete failure of a single DRAM device. This is known as Extended ECC protection. Note that all DIMMs shipped with the M7 systems use x4 devices.

Memory Channel Protection

- Between the CPU chip and the memory buffer chip, there is a high-speed interconnect similar to the interconnect that goes between the CPUs and the BX ASICs. This interconnect has the same properties of CRC protection, retry, and the capability of performing lane sparing.

New to M7

Redundant SP Proxies

On the M7-16 system, each DCU of (up to) 4 CPUs has an SP Proxy (SPP). While an SPP is not critical to the runtime functionality of the four CPUs, a functioning SPP must be present to boot a physical domain using that DCU. Having redundant SPPs ensures that a DCU will continue to be bootable in the circumstance of a failed SPP.

It also has the advantage of ensuring the ongoing management of a single-DCU physical domain that incurs the loss of an SPP.

To enable this feature, the FPGA that resides on the CMIOU has been designed with two ports, one for each SPP. This means both SPPs within a CMIOU chassis have access paths to all CMIOUs in the chassis, which in turn means that either SPP can assume responsibility for managing the CMIOUs.

DIMM Sparing

The physical address space provided by the memory DIMMs controlled by an individual CPU node are typically interleaved for performance reasons. An N-way interleave means that each successive cacheline resides in a different DIMM, from DIMM 0 to DIMM N-1. Cacheline N resides in DIMM 0 and the pattern repeats.

Interleave support is usually implemented in powers of 2. The plan of record for fully configured M7 systems is to use a 16-way interleave. The M7 processor also supports a 15-way interleave. Which means that if a physical domain is started with an unusable DIMM, that CPU node can still provide 15 DIMMs worth of physical address space rather than having to drop to an 8-way interleave and thus provide only 8 DIMMs worth of physical address space.

In addition, the M7 processor provides support for dynamically (with no interruption of service to the user) switching from a 16-way interleave to a 15-way interleave. Because it is not possible to fit 16 DIMMs worth of address into only 15 DIMMs, this feature will be possible only if the user has specified, at the time the physical domain is started, that the capability is desired. In this case, the platform FW will ensure that only 15 DIMMs worth of physical address space will be made available to the system, even though the DIMMs are configured for 16-way interleave operation.

Support of New RAS Features

Plesiochronous Clocking:

- A plesiochronous system is one where different parts of the system are almost, but not quite, perfectly synchronized.
- In general, plesiochronous systems behave similarly to synchronous systems.
- A plesiochronous system should employ means to cope with “sync slips,” which will occur at intervals due to the plesiochronous nature of the system.

Support of New RAS Features

DIMM Sparing:

- DIMM sparing is enabled by default for any node with a full population of 16 DIMMs.
  - 1/16th of each full pop node's memory is set aside in reserve to allow for sparing.
- DIMM faults due to correctable errors will invoke sparing when possible:
  - Defined as non-serviceable faults, so no notification
  - Fault classnames use a "-spare" suffix, for example, Fault.memory.dram-spare
- `fmadm faulty -a` will be the only way to know that a spare has occurred.
- A second faulty DIMM on a node will generate a serviceable fault, which will put both DIMMs out of service.

Customers have the ability to enable or disable DIMM sparing although it is not documented and they will need to open a service request (SR).

DIMM sparing is a mode that protects against persistent DRAM failure. Sparing monitors for an excessive amount of correctable errors and, when it detects the errors, copies the contents of an unhealthy portion of memory to an available spare. It increases the system reliability and uptime.

Reference Doc ID 2037793.1 – SPARC T7/M7 Servers : DIMM sparing FAQ

Support of New RAS Features

DIMM Sparing

- DIMM sparing has resulted in the demise of pin sparing.
  - Pin sparing conflicts with the SSM (Silicon Secured Memory) functionality; uses the same spare bits in DRAM.
  - Because almost all system will ship with full pop memory, DIMM sparing is possible and, therefore, will be used instead of pin sparing.
  - Even though pin sparing could be used in half pop configurations, or after a DIMM spare, the momentum behind SSM has led to a decision to make sure it is always available.

Redundant M7 Components

- A261 Power Supply (N+N)
- Switch and CMIOU Fan Modules (N+1)
- Service Processor/Service Processor Proxy
- Clock Synthesizers - per board (CMIOU/Switch)
- Power
  - CMIOU - 3.3V Clock Power, 1.2V DIMM VDDQ
  - CPU Mezzanine - VDD0-VDD3, VDDSOC, VDDT
  - Switch Unit - VDDA, VDDREF,VDD

Configuration Rules for M7-8

- Memory will be 16 DIMMs per CMIOU with 16 GB and 32 GB DIMM size offerings, half or fully populated.
- M7-8 (racked) or M7-16 requires two 26KVA/33KVA (LV/HV) PDUs.
- M7-8:
  - Minimum configuration is two CMIOUs in slots 0 and 1 to maintain rKVMS redundancy.
  - Either a Twinville or Powerville network card is required. Alternate is Niantic (optical).
- M7-8 with two PDoms:
  - Minimum configuration is four CMIOUs in slots 0,1,4,5 to maintain SP-HOST communication. Alternate is four CMIOUs in slots 0–3 if the customer wants to use CMIOU slots 4–7 for expansion later.
  - Either a Twinville or Powerville network card is required. Alternate is Niantic (optical).

Configuration Rules for M7-16

- CMIOU
  - Minimum 8 CMIOU/maximum 16 CMIOU
  - Factory configured CMIOU are ordered in pairs.
  - Existing systems can be upgraded by adding boards in single increments.
- Memory
  - Memory is half or fully populated.
  - Memory DIMM density must all be the same in a CMIOU.
  - Memory DIMM density and quantity can vary from board to board; however, it must be the same in each domain configurable unit (DCU).
- Domains
  - DCU is 4 CMIOU.
  - Minimum 2 CMIOU per DCU or PDom (physical domain)
  - PDom must include whole DCUs.

Summary

In this lesson, you should have learned how to:

- List the main components that comprise the servers
- Provide a comparison with previous products
- Define the hardware features
- Define the software features
- Describe the M7-8 and M7-16 Servers
- Describe the functions and features
- Describe the RAS features of these Servers with comparison to previous products

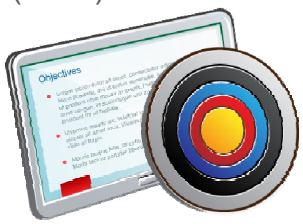
2

Platform Configuration and Administration Using ILOM

Objectives

After completing this lesson, you should be able to:

- Compare and contrast the differences between ILOM and the previous server's service processor
- Review the functions of the SPs, SPPs, and SPMs
- List ILOM's key functions, features, and highlights
- Compare the available physical network configurations
- Review the changes in the ILOM CLI within the Simplified Data Mode (SDM)
- Review the changes in the ILOM BUI within the Simplified Data Mode (SDM)
- Explain dual SP behavior



Objectives

After completing this lesson, you should be able to:

- Identify the chassis by using ILOM
- Configure the SP Shell (SPSH)
- Configure the ILOM network addresses
- Locate the server physically and by using ILOM
- Display server information by using ILOM
- View server and component information by using ILOM
- Update ILOM firmware

Additional Resources

The following references provide additional information about the topics described in this lesson:

- M7-8 and M7-16 Technology Portal on OTN:
  https://www.oracle.com/servers/sparc/m7-8/index.html
  https://www.oracle.com/servers/sparc/m7-16/index.html
- M7-8 and M7-16 white papers:
  https://www.oracle.com/servers/sparc/resources.html

M7 Systems Compared to M6-32

- SP is not a Single Point of Failure (SPOF) to system operation, but it is redundant and is hot pluggable.
- The SPPs (/SYS/SPPx) on an M6-32 are now SPMs (/SYS/SPPx/SPMy) on M7 platforms.
- M7 systems provide a redundant hot-plug SPP FRU to minimize system SPOFs.
- Redundant SPs are supported:
  - Active/Standby roles negotiated similarly to the way Active/Standby roles are negotiated on MS/M6
  - Two SPMs on each SP on the M7-8 with 2 PDOMs
  - One SPM on each SP on the M7-8 with 1 PDOM and M7-16 Switch Unit
  - SPs connected via an externally connected VLAN

Service Processor (SP)

- SPs are located on the CMIOU chassis on the M7-8.
  - The M7-8 with 2 PDOMs SP has two SPMs.
  - The M7-8 with 1 PDOM SP has one SPM.
- SPs are located on the Switch Chassis on the M7-16.
  - The SPPs are located on the CMIOU chassis.
- One SP serves as the Active SP and the other SP serves as the Standby SP.
- The Active SP manages system resources unless it fails when the Standby SP assumes the active role.
- On the M7-16, the active SP “aggregates” all SPPs and SPMs, and presents a unified system view.

Service Processor Proxy (SPP)

- The SPPs are available only on the top and bottom chassis of the M7-16.
- The SPPs in the CMIOU chassis manage DCU activity, whereas the SPs in the switch chassis manage system activity.
- Each CMIOU chassis also has two service processor proxies. SPP2 and SPP3 are in the top chassis, and SPPO and SPPl are in the bottom chassis.

Note that SPMs are not serviceable components. If an SPM fails, you must replace the SP or SPP that contains it.

Service Processor Module (SPM)

- The M7-8 with 2 PDOMs supports 2 SPMs per SP, whereas the M7-8 with 1 PDOM and M7-16 supports 1 SPM per SP.
- The SPM that is currently controlling the DCU is called “DCU-SPP” by analogy with the “PDOM-SPP” that controls one PDOM.
- The DCs are controlled by a pair of redundant SPMs.
- Oracle ILOM identifies one of the SPM pairs as the DCU-SPM to manage DCU activity.
- The other SPM pair runs Oracle ILOM, but remains inactive (Standby) unless the DCU-SPM pair can no longer manage the hardware.
- Oracle ILOM also identifies one of the DCU-SPMs from the pool of DCU-SPMs on the same PDomain as the PDomain-SPM to manage activity on that host.

Oracle ILOM Key Functions

- Management interface:
  - CLI, BUI, IPMI, SNMP
- Firmware upgrades
- Remote host management
- Inventory and component management
- System monitoring and alert/fault management (see the lesson titled ”Data Collection and Fault Analysis”)
- User account management (see the lesson titled “Platform Configuration and Administration”)
- Power consumption management

ILOM can be accessed via the command-line interface (CLI), browser user interface (BUI), intelligent platform management interface (IPMI), and simple network management protocol (SNMP). System firmware upgrades are performed through ILOM. ILOM is also used for the management of the host remotely as well as for system monitoring and power consumption management.

SP ILOM Highlights

Oracle ILOM looks and behaves just like Oracle ILOM on other platforms. It provides:

- Simple (user-visible) set of extensions to support physical domains
- Extensions to support service processor proxies and redundant service processors
  - Minimal impact on user experience

Oracle ILOM 3.2.5 or later will be released with the M7-8 and M7-16. The SP looks and behaves very similar to ILOM that runs on other platforms. There is a set of extensions that was added to support PDoms and SPPs.

Simplified Data Model (SDM) Command-Line Interface

The CLI is reorganized:

- **/System** and **/Servers** are among the new targets (trees) introduced.
- The different components of a system are grouped and organized into sub-targets of **/System** and **/Servers**.
- At every level of the tree, the critical properties are shown along with any sub-targets.
- The applicable CLI commands are supported at different levels of these trees.
- All targets and properties under **/System** and **/Servers** are case-insensitive.

SDM Command-Line Interface

- Health and health_details are two of the common properties that are shown at every level to indicate the overall health of that sub-tree.
- The Open_Problems target shows detailed descriptions of the faults in the system.
- The /SYS and /Storage targets have been made legacy.
  - They continue to exist but are hidden by default.
  - The legacy targets can be made visible by enabling the `/SP/cli/legacy_targets` property.

SDM CLI: Summary-Level Targets

```
-> show /System -d targets
 /System
 Targets :
     Open_Problems (11)
     DCUs
     Processors
     Memory
     Power
     Cooling
     Firmware
     Other Removable Devices
     Log
```

SDM CLI: Summary-Level Properties

```
-> show /System -d properties
/System
  Properties:
    health = Service Required
    health details = PS3 (Power Supply 3), PS2 (Power Supply 2), PSl
           (Power Supply 1), PSS (Power Supply 5)
     are faulty. Type 'show /System/Open Problems' for details.
    open_yroblems_count = 11 -  type = Domained Server
    model = SPARC M7-16
    qpart id = Ql0776
    part umber = 32397701+2+1 serial number =
    AK00192121 system-identifier = (none)
    system-fw version = Sun System Firmware 9.4.0.build 30 2015/01/29
    14:42 ilom address = 10 .129.89.30
    ilom-mac address = 00:10 :E0 :37 :27 :A4
    locator indicator = Off power state = On
    actual_power_consumption 1933 watts
    action = (none)
```

SDM CLI: Processor Subsystem

```
-> show /System/Processors/CPUs/CPU_x modei
 /System/Processors/CPUs/CPU 0
  Properties :
       model     Oracle SPARC M7
```

SDM CLI: Processor Subsystem

```
-> show /System/Processors/CPUs/CPU_0
/System/Processors/CPUs/CPU 0
 Targets :
 Properties :
   health = OK health details
   requested_state = Enabled
   part number = Not Available
   serial   number   =   00000000000000000000b9078a392f186
   location = CMIOUO/CM/CMP (CPU Memory IO Unit 0)
   model - Oracle SPARC M7
   max clock_speed = 4 .13 GHz
   total cores = 32
   enabled_cores = 20
   temperature = 46 degrees C
```

ILOM 3.2.5 Web Redesign

- The navigation tree on the left replaces the old tabs that were on top.
- The top levels of the tree include the new Summary and Subsystem pages.
- The pages are organized by purpose:
  - System Information
  - Remote Control
  - Power, Host, System, and ILOM Administration

Web Summary Page

- General Information table: Basic information about the server and SP
- Action table: Quick access to common ILOM actions [power on and off, Oracle System Assistant (OSA), and firmware update]
- Subsystem Status table: Quick summary of the various subsystems

ILOM BUI

There are some key differences in the BUI from ILOM 3.0.

- The old components, sensors, and indicators pages have been removed and replaced with new subsystem pages.
- The new subsystem pages focus less on raw sensor data and more on the status of the components.
- The old RAID pages are now replaced by the storage subsystem page.
- The Session Timeout page has been merged with the Web Server Configuration page.
- The Open Problems page brings together all system problems in a single spot, and replaces the old Fault Management page.
- There is also a new Comprehensive Summary page.

Dual SP Behavior

- Race to become ACTIVE_SP:
  - View system status and errors by making connections to the SP0 and SP1 serial ports.
  - Chassis configuration is done on ACTIVE_SP.
  - DHCP is not supported at revenue release (RR).
- On the ACTIVE_SP:
  - The top-level targets are: Servers, System, and SP.
- On the Standby:
  - Only /SP is available.
- To see which SP you are on:

```
-> SHOW /SP/redundancy status
Properties:
Status = Active
```


Before powering on the server for the first time, make a serial connection to the two SPs. By making these serial connections, you will be able to view the system status and error messages when you switch on the power grid circuit breakers. If you power-on the server without first connecting the terminals to the SER MGT port, you will not see the system power-on messages.

When the system is powered on, there is a race to become the active SP. All chassis configuration is done from the active SP. Its top-level targets are servers, system, and SP. On the standby SP, only the SP target is available.

SP Shell (SPSH): Dual SP

- Manual failover properties include:
  - Target /SP/redundancy
  - Properties
    - `fru_name = /SYS/SPx`
    - `Initiate_failover_action = (none)`
    - `Status = Active`
- To switch the ACTIVE_SP responsibilities:
  `-> reset /SP`
  - Reboots SPs and SPPs
  - No impact on the running domain

SP failovers can be performed manually from the SP shell. The target that is used to perform the failover is `/SP/redundancy`. When the `initiate_failover_action` property is set to `true`, the information is propagated to the SP and the SPPs. A reset of the active SP is required, which will reboot the SPs and the SPPs. There is no impact on the running domains. To confirm that you are connected to the active SP, execute `-> show /SP/redundancy status`. If `status = Active`, you are connected to the active SP. If `status = Standby`, you are logged in to the standby SP. If `status = Standalone`, only one of the SPs is responding.

SPSH: Chassis Identification

- To display the system's model:

```
-> show /System model
   /System
     Properties:
      model = SPARC M7
```

- To display the system's part number:

```
-> show / System part_number
   /System
     Properties:
      part_number =
```

- To display the system's serial number

```
-> show / System serial_number
   /System
     Properties:
      Serial_number = AK001802287
```

To continue with the installation, you must log in to the ILOM software on the active SP through a local serial connection. Log in to the active SP as `root` with a password of `changeme`. You can run the commands shown in the slide to perform chassis identification for your system.

SPSH: Setup

- Set the host name:
  `-> set /SP hostname=`*`text string`*
- Configure the network:
  `-> set /SP/network `*`property=value`*
  Where property can be:

```
commitpending
dhcp_clientid
dhcp_server_ip
ipdiscovery
ipgateway
ipnetmask
pendingipdiscovery
pendingipgateway
pendingipnetmask
```


If the SP host name is not set, the login prompt displays ORACLE-PSN. When configuring the SP network, you need at least three IP addresses. One for the active SP that will float to whichever SP is currently active, one for SP0, and one for SP1.

Configuring ILOM Network Addresses

- Set the gateway IP address for all SP addresses:
  `-> set /SP/network pendingipgateway=xxx.xxx.xxx.xxx`
- Set the netmask IP address for all SP addresses:
  `-> set /SP/network pendingipnetmask=255.255.xxx.0`
- Assign the IP address for the active SP:
  `-> set /SP/network/ACTIVE_SP pendingipaddress=xxx.xxx.xxx.xxx`
- Assign the IP address for SP0:
  `-> set /SP/network/SP0 pendingipaddress=xxx.xxx.xxx.xxx`
- Assign the IP address for SP1:
  `-> set /SP/network/SP1 pendingipaddress=xxx.xxx.xxx.xxx`

The SPs do not support DHCP. You must assign static network addresses to the following components:

- **SP0:** NET MGT port on SP0
- **SP1:** Net MGT port on SP1
- **ACTIVE_SP:** Active SP. If the active SP fails, the standby SP will be assigned this address.
- **HOST0:** The IP address for the PDomain0-SPP host. The server is configured as one PDomain; so only one host requires an address. If you reconfigure the server to have multiple PDomains, you must assign network addresses to the other hosts.

These network addresses must be configured before you can access the ILOM software over a network connection.

Configuring the ILOM Network Addresses

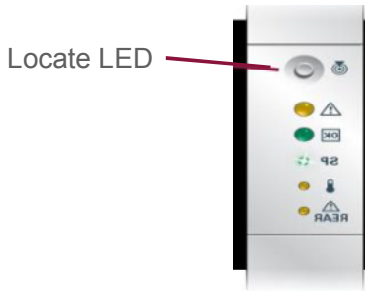
- Assign the IP address for HOST0 (PDomain0-SPP):
```
-> set /SP/network/HOST0 pendingipaddress=xxx.xxx.xxx.xxx
```
- Verify that you have set the network addresses correctly:
```
-> show/SP/network -level 2 -output table pendingipaddress pendingipnetmask
   pendingipgateway
   Target                    I  Property           I Value
   ------------------------+--------------------+---------------------
   /SP/network               | pendingipgateway   | 10.141.246.1
   /SP/network               | pendingipnetmask   | 255.255.255.0
   /SP/network/ACTIVE_SP     | pendingipaddress   | 10.141.246.194
   /SP/network/HOST0         | pendingipaddress   | 10.141.246.182
   /SP/network/HOSTl         | pendingipaddress   | 10.141.246.183
   /SP/network/H0ST2         | pendingipaddress   | 10.141.246.184
   /SP/network/H0ST3         | pendingipaddress   | 10.141.246.185
   /SP/network/SP0           | pendingipaddress   | 10.141.246.180
   /SP/network/SPl           | pendingipaddress   | 10.141.246.181
```
- Commit the addresses:
```
-> set/SP/network commitpending=true
  set 'commitpending' to 'true'
```


After setting the IP addresses, you must commit them for the new addresses to take effect. If you change the IP address that you are connected into, you will need to re-establish your connection after you set `commitpending=true`.

Updating the Firmware

- The SP firmware can be updated for all SPs simultaneously.
- The firmware image includes the Oracle ILOM firmware, OpenBoot PROM firmware, POST firmware, and miscellaneous files.
- The firmware image is installed in the SP flash memory.
- A user must have the administrator (a) or field engineering (s) role to update the firmware.
- Firmware can be upgraded from both the CLI and the BUI.

Locating the Server (Web Interface)

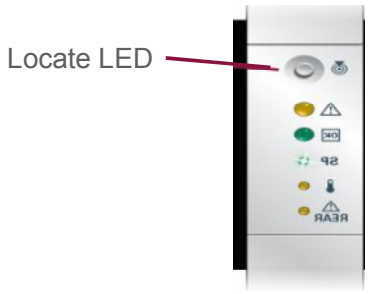


Front and rear of the CMIOU chassis on the left side of the rack

To locate the server by using the BUI, log in to the ILOM web interface. View the System Information > Summary page. Click the Locator Indicator button in the Actions panel. When prompted, click Yes to confirm the action. The server's Locate LED becomes illuminated on both the front and rear of the server so that you can physically identify the server. To turn off the Locate LED, you can press the Locate LED button if you are near the server. You can also turn it off remotely by using the web interface. On the Summary page, click the Locator Indicator button.

Locating the Server (CLI)

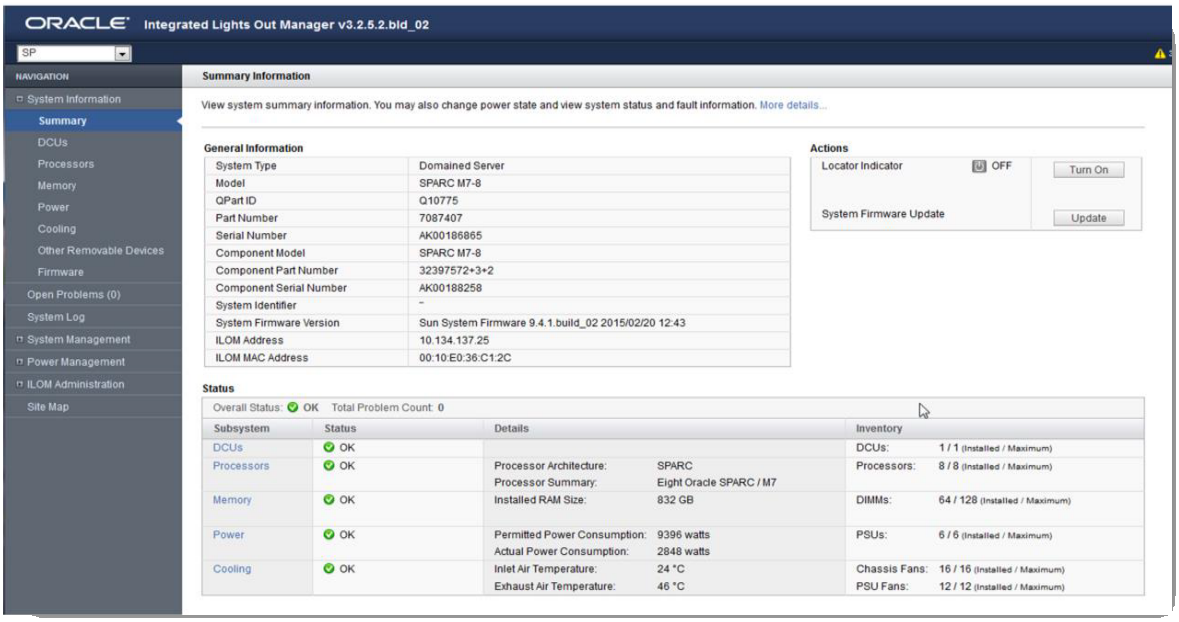
- To flash the Locate LED:
  ```
  -> set /SYS/LOCATE value=Fast_Blink
  ```
- To turn off the Locate LED:
  ```
  -> set /SYS/LOCATE value=off
  ```
- To display the status of the Locate LED:
  ```
  -> show /SYS/LOCATE
  ```



Front and rear of the CMIOU chassis on the left side of the rack

If you need to service a component, turning on the system's Locate LED helps in easily identifying the correct server. You do not need administrator permissions to use these commands. The server's Locate LED blinks on both the front and rear of the server so that you can physically identify it.

Displaying Server Information (Web Interface)



To display server information through the BUI, log in to the ILOM web interface and view the Summary page. The Summary page provides the following information:

- **General Information panel:** Provides general information such as the serial number, firmware version, primary OS, host MAC address, SP IP addresses, and MAC addresses
- **Actions panel:** Indicates the power state of the host
- **Status panel:** Indicates the overall status of the server components

To see more detailed information, click the specific components listed under System Information.

Displaying Server Information (CLI)

```
-> show components
Target                                I  Property               I Value
--------------------------------------+-----------------------+----------------
/SYS/CMIOU0                           | current_config_state   | Enabled
/SYS/CMIOU0/CM/CMP                    | current_config_state   | Enabled
/SYS/CMIOU0/CM/CMP/BOB00              | current_config_state   | Enabled
/SYS/CMIOU0/CM/CMP/BOB00/CH0          | current_config_state   | Enabled
/SYS/CMIOU0/CM/CMP/BOB00/CH0/DIMM     | current_config_state   | Enabled
/SYS/CMIOU0/CM/CMP/BOB00/CH1          | current_config_state   | Enabled
/SYS/CMIOU0/CM/CMP/BOB01              | current_config_state   | Enabled
/SYS/CMIOU0/CM/CMP/BOB01/CH0          | current_config_state   | Enabled
/SYS/CMIOU0/CM/CMP/BOB01/CH0/DIMM     | current_config_state   | Enabled
<cont>
```

Viewing Server and Component Information

- In addition to the systemwide and subcomponent statuses that you can access with ILOM, for this server, you can view the state of individual PDomains or specific components (DCUs, CMIOUs, or CMPs).
- To view the information, the user accounts for each component must be assigned read-only operator (o) user roles.

Verifying Hardware Status

- Verify the hardware status and investigate any issues:

```
-> show /System/Open_Problems
Open Problems (11)
Date/Time                    I  Subsystems     I Component
---------------------------+---------------+----------------------
Mon Jan 26 22 :24: 13 2015    Power             PS3 (Power Supply 3)
         A power supply AC input voltage failure has occurred.
         (Probability:100,  UUID:55cf03ba-33cd-e4e6-f968-cff3e8d33a22,
         Resource:/SYS/PS3/SUPPLY, Part Number:7068817,
         Serial Number:465776G+l348B20C56, Reference
         Document:http://support.oracle.com/msg/SPT-8000-5X)
…
Paused: press any key to continue, or 'q' to quit
```


In the example in the slide, there is an issue with CMIOU8 (which is in DCU2).

Verifying Components

- Verify that there are no degraded components:
  ```
  -> show /System health
  ```
- When everything is OK, an example output would be:
  ```
  /System
  Properties:
  health = OK
  ```
- When there is a problem, an example output would be:
  ```
  /System
  Properties:
  health = Service Required
  ```
- If Service Required is shown, as in the preceding example, investigate further:
  ```
  -> show /System health_details
     /System
     Properties:
     health_details = PS3 (Power Supply 3), PS2 (Power Supply 2),
                      PSl (Power Supply 1), PSS (Power Supply 5) are faulty.
                      Type 'show /System/Open_Problems' for details.
  ```


If more details are required:

```
-> show -l all -o table /System
```

This produces a long table of output.

Viewing the State of a Specified PDomain (1/2)

- To view information about a specific domain:

```
/Servers/PDomains/PDomain_0/HOST
    Targets:
        VPS
        VPS_CPUS
        VPS_MEMORY
        bootmode
        console
        diag
        domain
        status_history
        tpm
        verified_boot
    Properties:
        autorestart = reset
        autorunonerror = none
        bootfailrecovery = poweroff
        bootrestart = none
        boottimeout = 0
        dcus_assigned = /SYS/DCU0
        dimm sparing = enabled
        GM 1.5.1.build_02 2015/02/20 11:28 [m7-glueless:debug]
```

Viewing the State of a Specified PDomain (2/2)

```
<cont>
      hostconfig_version = Hostconfig l.5.1.build_02 2015/02/20 11:15 (m7-glueless:debug}
      hypervisor_version = Hypervisor 1.14.1.build_02 2015/02/20 10:49
      {greatlakes:m7-platform:debug]
      keyswitch_state =  Normal
      mac address = 00:10:e0:47:89:04
      maxbootfail = 3
      obp_version = OpenBoot 4.37.l.build_02 2015/02/20 10:20
      operation_in_progress = none
      post_version = POST 5.2.l.build_02 2015/02/20 10:48
      power_state = On
      send_break_action = (Cannot show property)
      sp_name = /SYS/SPl/SPM0
      state_capture_mode = default
      state_capture_on_error = enabled
      state_capture_status = enabled
      status = OpenBoot Running OS Boot
      status_detail 20150304 14:42:26: Host status updated
      sysfw_version = Sun System Firmware 9.4.1.build_02 2015/02/20 12:43
<cont>
```

Viewing System-Level DCU Properties

- To view information and health status details for all DCUs in the system:

```
-> show /System/DCUs
        /System/DCUs
            Targets:
              DCU_0
              DCU_1
              DCU_2
              DCU_3
            Properties:
              health = OK
```

Viewing System-Level DCU Properties with a Fault

```
-> show /System/DCUs
        /System/DCUs
            Targets:
              DCU_0
              DCU_1
              DCU_2
              DCU_3
            Properties:
              health = Service Required
              health details = CMIOU0 is faulty.
              Type 'show /System/Open_Problems' for details.
              installed dcus = 4
              max dcus = 4
```

Viewing Individual DCU Properties

- To view the health status and properties for a specific DCU:

```
-> show /System/DCUs/DCU_0/
  /System/DCUs/DCU_O
         Targets:
            CMIOU_O
        Properties:
            health = OK
            health_details = -
            power_state = Off
            cpu_summary = One Oracle SPARC M7
            memory_summary = 128 GB
            location = DCU0 (Domain Configuration Unit 0)
            host assigned = /HOST0
            fan_list = FM0/F0 (Fan Module 0), FM0/Fl (Fan Module 0), FMl/F0 (Fan Module l),
                       FMl/Fl (Fan Module l), FM2/F0 (Fan Module 2), FM2/Fl (Fan Module 2),
                       FM3/F0 (Fan Module 3), FM3/Fl (Fan Module 3), FM4/F0 (Fan Module 4),
                       FM4/Fl (Fan Module 4), FM5/F0 (Fan Module 5), FM5/Fl (Fan Module 5),
                       FM6/F0 (Fan Module 6), FM6/Fl (Fan Module 6), FM7/F0 (Fan Module 7),
                       FM7/Fl (Fan Module 7}
            sp_name = /SYS/SPPl/SPM0
            initiate_sp_failover = (none)
<cont>
```

Viewing Individual DCU Properties with a Fault

```
-> show /System/DCUs/DCU_3
   /System/DCUs/DCU_3
      Targets:
         CMIOU_12
         CMIOU_13
         CMIOU_14
         CMIOU_15
      Properties:
         health = Service Required
         health_details = CMIOU13 (CPU Memory IO Unit 13) is faulty. Type 'show
         /System/Open_Problems' for details.
         power_state = Off
         cpu_summary = Four Oracle SPARC M7
         memory_summary = 1536 GB
         location = DCU3 (Domain Configuration Unit 3)
         host_assigned = /HOST0
         fan_list = FM8/F0 (Fan Module 8), FM8/Fl (Fan Module 8), FM9/F0 (Fan Module 9), FM9/Fl (Fan Module 9),
                    FM10/F0 (Fan Module 10), FM10/Fl (Fan Module 10), FM11/F0 (Fan Module 11),
                    FM11/Fl (Fan Module 11), FM12/F0 (Fan Module 12), FM12/Fl (Fan Module 12),
                    FM13/F0 (Fan Module 13), FM13/Fl (Fan Module 13), FM14/F0 (Fan Module 14),
                    FM14/Fl (Fan Module 14), FM15/F0 (Fan Module 15), FM15/Fl (Fan Module 15)
          sp_name = /SYS/SPP2/SPM1
          Initiate_sp_failover = (none)
<cont>
```

Viewing CMIOU Status

- To view the health status and properties for a specific CMIOU:

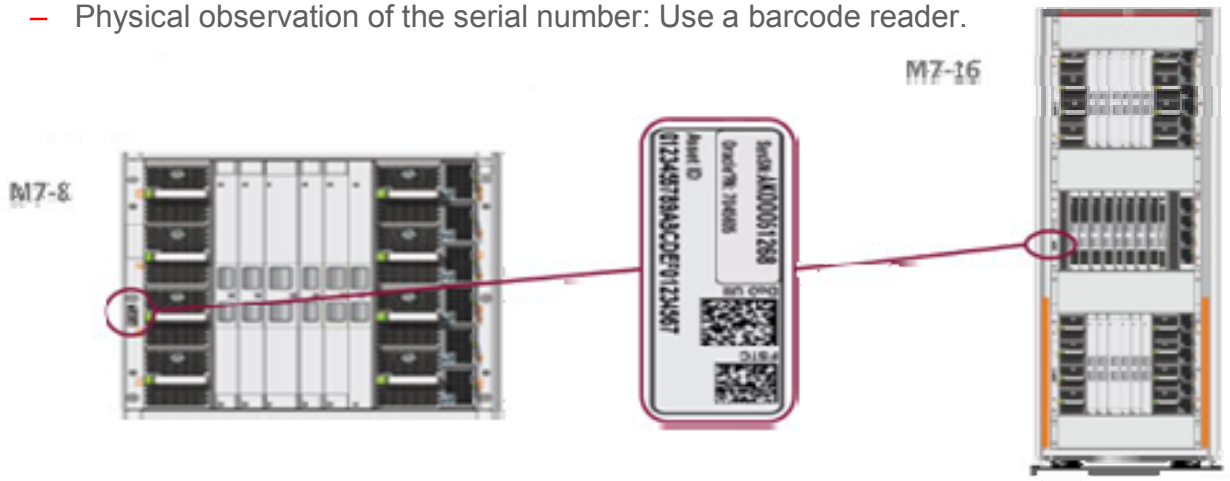
```
-> show /System/DCUs/DCU_0/CMIOU_0 health
  /System/DCUs/DCU_0/CMIOU_0
      Targets:
      Properties:
          health = OK
          health details = -
          requested_state = Enabled
          power_state = Off
          model = ASSY CMIOU
          location = CMIOU0 (CPU Memory IO Unit 0)
          part_number = 7084599
          serial number = 465769T+l4056COONR
          action = (none)
      Commands:
          cd
          set
          show
```

Viewing CMIOU Status with a Fault

```
-> show /System/DCUs/DCU_3/CMIOU_l2 health
  /System/DCUs/DCU_3/CMI0U_l2
      Targets:
      Properties:
          health = Service Required
          health details = During poweron testing, a lane failover has occurred
          on an interconnect between a CPU chip and a switch chip. Type
          'show /System/Open_Problems' for details.
          requested_state = Enabled
          power_state = On
          model = -
          location = CMIOU12 (Processor Board 12)
          part_number = 07042349
          serial number = 465769T+1235VY01CU
          action = (none)
```

Obtaining Server Serial Number

- Use one of the following methods to obtain the server identification and serial numbers:
  - Administrative tools: You can obtain the server serial number by using the ILOM `show /System serial_number` command.
  - Physical observation of the serial number: Use a barcode reader.



The system serial number is located on the front of the rack near the top-right corner.

Updating Firmware Steps

To update the system firmware, perform the following steps:

1. Open an SSH session to connect to the active SP:
   ```
   ssh root@xxx.xxx.xxx.xxx
   ```
2. Set the keyswitch state to normal for each PDom:
   ```
   -> set /Servers/PDomains/PDomain_x/HOST keyswitch_state=normal
   ```
3. Upgrade the firmware:
   ```
   -> load -source URI /SP/firmware
   ```
4. Verify the version of the firmware that is running:
   ```
   -> show /System/Firmware
     Properties:
         system_fw_version = Sun System Firmware xxx
   ```


There is no progress meter during an upgrade, which will take about 45 minutes. A firmware upgrade will cause the SP to be reset; therefore, if there is a connection to the active SP, it will need to be re-established. It is recommended that a clean shutdown of the server be performed before the upgrade procedure.

To ensure that all the hosts are updated at the time the SPs are updated, the hosts must be powered off. The firmware can be updated without impacting the running hosts. If any of the hosts are running, the firmware is automatically updated when the hosts are reset.

The SP will enter a special mode to load new firmware. No other tasks can be performed on the SP until the firmware upgrade is complete and the SP is reset.

The following protocols are supported: FTP, TFTP, SFTP, SCP, HTTP, and HTTPS

Summary

In this lesson, you should have learned how to:

- Compare and contrast the differences between ILOM and the previous server's service processor
- Review the functions of the SPs, SPPs, and SPMs
- List ILOM's key functions, features, and highlights
- Compare the available physical network configurations
- Review the changes in the ILOM CLI within the Simplified Data Mode (SDM)
- Review the changes in the ILOM BUI within the Simplified Data Mode (SDM)
- Explain dual SP behavior

Summary

In this lesson, you should have learned how to:

- Identify the chassis by using ILOM
- Configure the SP Shell (SPSH)
- Configure the ILOM network addresses
- Locate the server physically and by using ILOM
- Display server information by using ILOM
- View server and component information by using ILOM
- Update ILOM firmware

Appendix A

Managing User Accounts

Managing User Accounts

- The Oracle ILOM software that is pre-installed on the server is preconfigured with one user account (root).
- This user can then create additional users accounts as needed.
- The M7 series servers support up to 60 user accounts for logging in to the SP. However, these servers support only 25 simultaneous login sessions.
- These servers introduce the concept of separately managed physical domains (PDoms). Each PDom can be separately controlled.
- Each user account can be optionally configured with specific roles for an individual PDom.

The SP is an appliance. In an appliance model, users or management agents can access the SP and its components only through authorized user interfaces. Users and agents cannot access any of the underlying operating system interfaces, and users cannot install individual software components on the SP.

A user account is a record of an individual user that can be verified through a username and password. Each user account is assigned specific roles that allow a user to execute a subset of ILOM commands and perform select actions on a specific set of components. These components can be physical components, domains, or physical components within a domain. By specifying roles for each user, you can control which operations each user is allowed to perform.

When you assign user roles to a user account for a specific component, the capabilities granted mirror those of the user.

Managing User Accounts

The user roles are additive. For example, a user account is permitted to do reset operations on a PDom in either of the following cases:

- The user has the “r” role for the platform (in /SP/users/username/role).
- The user has the “r” role for the specific PDom (in /SP/users/username/host_roles/hostX_role).

Roles and Capabilities

Role	Capabilities
none	User has no user roles; this prevents the user's roles from being looked up in LDAP.
User (u)	• Create, delete, disable, and enable user accounts. • Change a user's password and password properties. • Change a user's roles. • View all platform states.
administrator (a)	• Perform all service processor configuration tasks other than the useradm and auditadm tasks. • Assign hardware to or unassign hardware from domains. • Perform service processor power operations. • Perform service processor failover operations on systems with more than one service processor. • Perform all operations on the domain hardware. • View all platform and physical domain states.
console (c)	• Access the Oracle ILOM Remote console and the SP console. • View and change the state of the Oracle ILOM console configuration variables.
reset (r)	• View all states of the hardware assigned to the domains on which this role is held. • View all states of the domains on which this role is held. • Operate the system, which includes performing power operations, resetting the system, hot-plugging devices, enabling and disabling components, and fault management for the specified domain.

read-only operator (o)	• View all platform states.
	• Change the password and the session timeout setting for your own user account.
	• When this role is defined at the domain level, users can:
	  • View all states of the hardware assigned to the domains on which this role is held
	  • View all states of the domains on which this role is held

field engineering (s)	Perform all operations reserved for field engineers.

Required User Roles

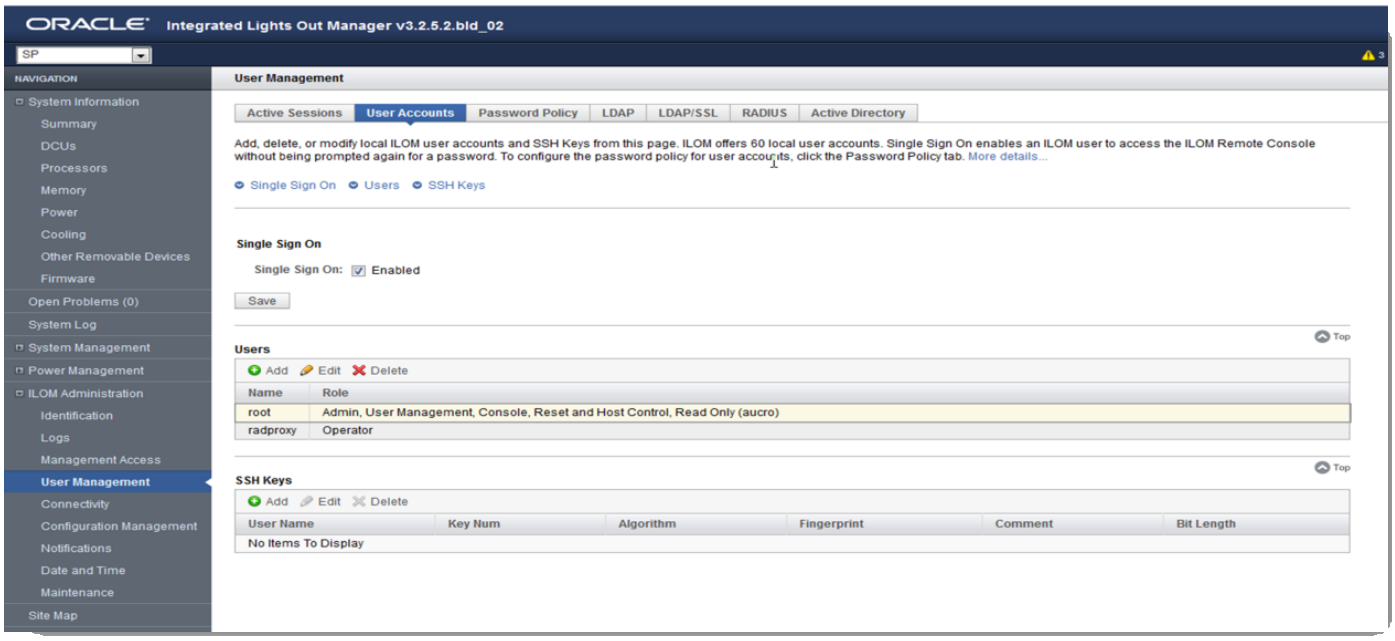
Task	Required User Role
Task Assign and unassign DCUs to a PDomain	admin (a)
Manage ILOM passwords	user (u)
Connect to a PDomain	console (c)
Perform power operations (start, stop, and reset)	reset (r)
Configure user accounts	user (u)
Configure host groups	user (u)

A specific role is required for certain tasks as shown in the table in the slide.

Configuring User Accounts

- User accounts can be created through the BUI, the command line, or through SNMP.
- You must have the user (u) role to configure local user accounts.

Configuring User Accounts (Web Interface)



To configure user accounts through the BUI, perform the following steps:

- Log in to the ILOM web interface.
- Navigate to the ILOM Administration > User Management page. The Active Sessions page is displayed.
- Click the User Accounts tab.
- In the Users table, click Add. Enter the username, password, and password confirmation. Select a CLI mode and the appropriate roles for this user. Select the appropriate roles for the domains that you want this user to have.
- Click Save.

Configuring User Accounts (CLI)

- Create a user account:

```
-> create /SP/users/barry
     Creating user ...
          Enter new password: *********
          Enter new password again: *********
          Created /SP/users/barry
```

- Set a user password:

```
-> set /SP/users/username password
     Enter new password: ********
     Enter new password again: ********
```

- Assign user roles for each active PDomain:

```
-> set /SP/users/barry/host_roles host1 role=acr
     Set 'host1 role' to 'acr'
```


To configure user accounts through CLI, log in to ILOM. Create a user account and assign a password, as well as its roles. The password length must be between 8 and 16 characters.

Viewing User Accounts

- To view all user accounts:

```
-> show /SP/users
     /SP/users
          Targets:
               root
               barry
          Properties:
```

- To view an individual user account:

```
-> show /SP/users/barry
     /SP/users/barry
          Targets:
               ssh
               host roles
          Properties:
               role = 0
               password *****
```


You must have the `user (u)` permissions to view the properties of existing user accounts.

Viewing and Deleting User Accounts

- To view an individual user account's roles:

```
-> show /SP/users/barry/host_roles
     /SP/users/harry/host_roles
          Targets:
          Properties:
               host0 role = (none)
               host1 role = acr
               host2 role = (none)
               host3 role = (none)
```

- To delete a user account:

```
-> delete /SP/users/barry
    Are you sure you want to delete /SP/users/barry (y/n)? Y
    Deleted /SP/users/barry
```


You must have `user (u)` permissions to view the properties of and delete existing user accounts.

Managing User Authentication

You can use Active Directory, LDAP/SSL, or RADIUS to configure host groups for remote user authentication.

- Active Directory provides both authentication of user credentials and authorization of user access levels to networked resources.
- LDAP/SSL offers enhanced security to LDAP users.
- RADIUS provides many servers shared access to user data in a central database, providing good security and easy administration.

ILOM can authenticate user accounts through local accounts that you configure or against a remote user database, such as Active Directory, LDAP/SSL, or RADIUS. With remote authentication, you can use a centralized user database rather than configuring local accounts on each ILOM instance.

User access can be remotely authenticated and authorized based on a user's membership in a host group. A user can belong to more than one host group and, on this server, you can configure up to 10 host groups. The tasks involved in configuring host groups include managing certificates (LDAP/SSL), administrator groups, operator groups, custom groups, and user domains.

Active Directory is the distributed directory service included with Microsoft Windows Server operating systems. Like an LDAP directory service implementation, Active Directory is used to authenticate user credentials.

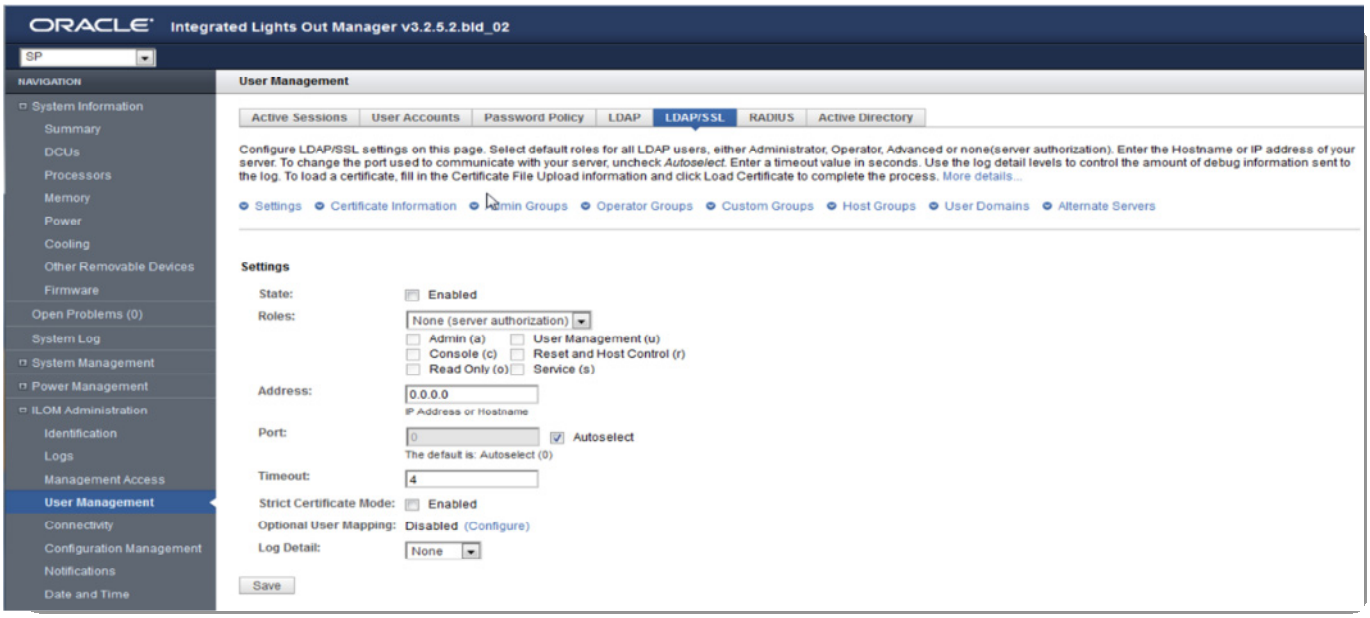
LDAP/SSL offers enhanced security to LDAP users by way of SSL technology. To configure LDAP/SSL in an SP, you enter basic data (such as primary server, port number, and certificate mode) and optional data (such as alternate server, event, or severity levels). You can enter this data by using the LDAP/SSL configuration page of the ILOM web interface, the CLI, or SNMP.

Host Group Properties

- Manage host group properties in the CLI from the following locations:
  - `/SP/clients/activedirectory/hostgroups/<id>`
  - `/SP/clients/ldapssl/hostgroups/<id>`
- Use the following properties:

Property	Description
name	Read/write property that represents the Active Directory or LDAP/SSL group name for the specified host group
hosts	Read/write property that lists the PDomain for which this host group assigns roles
roles	Read/write property that specifies the domain-specific privilege levels for the host group. This property supports any of the individual host role ID combinations of “a,” “c,” and “r.”

Configuring Host Groups for Active Directory or LDAP/SSL (Web Interface)



To configure host groups through the BUI, perform the following steps:

- Access the ILOM web interface.
- Navigate to the ILOM Administration > User Management page. The Active Sessions page is displayed.
- Click the Active Directory or LDAP/SSL tab.
- At the top of the page, click the link to access the Host Groups category. Enable the radio button of the individual table, and then click Edit.
- Enter the name of the host group and select the hosts that you want to be members of the specified host group. Select the appropriate roles for this host group.
- Click Save.

Configuring Host Groups for Active Directory or LDAP/SSL (CLI)

- Enter the name of the host group based on your service.

```
-> set /SP/clients/activedirectory/hostgroups/id property=value
```

Where ***property*** can be: ***hosts***, ***name,*** or ***roles***

```
-> set /SP/clients/ldapssl/hostgroups/id property=value
```

Where ***property*** can be: ***hosts***, ***name***, or ***roles***

- Specify which hosts you want to be members of the specified host group based on your service.

```
-> set /SP/clients/activedirectory/hostgroups/id hosts="/HOSTx /HOSTy"
-> set /SP/clients/ldapssl/hostgroups/id hosts="/HOSTx /HOSTy"
```

Configuring Host Groups for Active Directory or LDAP/SSL (CLI)

- Specify the appropriate roles for this host group:
  - `->set /SP/clients/activedirectory/hostgroups/id roles=`*`value`*
  - `->set /SP/clients/ldapssl/hostgroups/id roles=`*`value`*

You must have `user (u)` permissions to configure host groups.

Configuring Host Groups (SNMP)

- Host groups are managed by using SNMP from the following tables in the `SUN-ILOM-CONTROL-MIB` file:
  - `ilomCtrlActiveDirHostGroupsTable`
  - `ilomCtrlLdapSslHostGroupsTable`

You must have `user (u)` permissions to configure host groups.

Configuring Host Groups for Active Directory (SNMP)

- To view the current name of a specific Active Directory host group:

```
% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
    ilomCtrlActiveDirHostGroupRoles.2
    SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirHostGroupName .2 = STRING: platadm
```

- To specify a new name for a specific Active Directory host group:

```
% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
    ilomCtrlActiveDirHostGroupRoles.2 s "platops"
    SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirHostGroupName.2 = STRING: platops
```

- To specify the hosts for which a specific host group assigns roles:

```
% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
          ilomCtrlActiveDirHostGroupHosts.2
          SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirHostGroupHosts.2 = STRING: /HOST2

% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
          ilomCtrlActiveDirHostGroupHosts.2 "HOST1"
          SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirHostGroupHosts.2 = STRING: HOST1
```


The example in the slide uses the following values:

- Host group ID number: 2
- Existing host group name: platadm
- New host group name: platops
- Host that is assigned roles by this host group: HOST2
- New host that is assigned roles by this host group: HOST1
- Existing host group roles: a, r
- New host group roles: a, c, r

Configuring Host Groups for Active Directory (SNMP)

- To view the roles of a specific Active Directory host group:

```
% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
   ilomCtrlActiveDirHostGroupRoles.2
   SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirHostGroupRoles .2 = STRING: "acr"
```

- To specify new roles for a specific Active Directory host group:

```
% snmpset -vl -cprivate -mALL SNMP_agent_ipaddress
   ilomCtrlActiveDirHostGroupRoles.2 s "acr"
   SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirHostGroupRoles.2 = STRING: "acr"
```

- To verify the new roles:

```
% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
   ilomCtrlActiveDirHostGroupRole.2
   SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirHostGroupRoles.2 = STRING: "acr"
```

Configuring Host Groups for LDAP/SSL (SNMP)

- To view the current name of a specific LDAP/SSL host group:

```
% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
        ilomCtrlLdapSslHostGroupName.3
        SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslHostGroupName.3 = STRING:
        CN=SpSuperCust,OU=Groups,DC=john,DC=sun,DC=com
```

- To specify a new name for a specific LDAP/SSL host group and to verify that the change has been made:

```
% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
        ilomCtrlLdapSslHostGroupName.3 s
        CN=SpSuperCust,OU=Groups,DC=bills,DC=sun,DC=com
        SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslHostGroupName.3 = STRING:
        CN=SpSuperCust,OU=Groups,DC=bills,DC=sun,DC=com

% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
        ilomCtrlLdapSslHostGroupName.3
        SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslHostGroupName.3 = STRING:
        CN=SpSuperCust,OU=Groups,DC=bills,DC=sun,DC=com
```

The example in the slide uses the following values:

- Host group ID number: 3
- Existing host group name: platadm
- New host group name: platops
- Host that is assigned roles by this host group: /PDomain_1, PDomain_2
- New host that is assigned roles by this host group: /PDomain_1, PDomain_3
- Existing host group roles: a, r
- New host group roles: a, c, r

Configuring Host Groups for LDAP/SSL (SNMP)

- To specify the hosts for which a specific host group assigns roles:

```
% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
   ilomCtrlLdapSslHostGroupHosts.3
   SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslHostGroupHosts.3 =
   STRING:/PDomain_1/PDomain_2

% snmpset -vl -cprivate -mALL SNMP_agent_ipaddress
   ilomCtrlLdapSslHostGroupHosts.3 s "/PDomain_1/Pdomain_3"
   SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslHostGroupHosts.3 =
   STRING:/PDomain_1/PDomain_2
```

- To view the roles of a specific LDAP/SSL host group:

```
% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
   ilomCtrlLdapSslHostGroupRoles.3
   SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslHostGroupRoles.3 = STRING: "acr"
```

Configuring Host Groups for LDAP/SSL (SNMP)

- To specify new roles for a specific LDAP/SSL host group and to verify the new roles:

```
% snmpset -vl -cprivate -mALL SNMP_agent_ipaddress
   ilomCtrlLdapSslHostGroupRoles.2 s "acr"
   SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslHostGroupRoles.3 = STRING:"acr"
% snmpget -vl -cprivate -mALL SNMP_agent_ipaddress
   ilomCtrlLdapSslHostGroupRole.2
   SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslHostGroupRoles.3 = STRING:"acr"
```

Setting the Server Altitude

- To set the server's altitude

  ```
  -> set /SP system_altitude=altitude
  ```

  Where altitude can be replaced with the altitude of the data center in meters. The possible values are 0 to 3,000 meters.

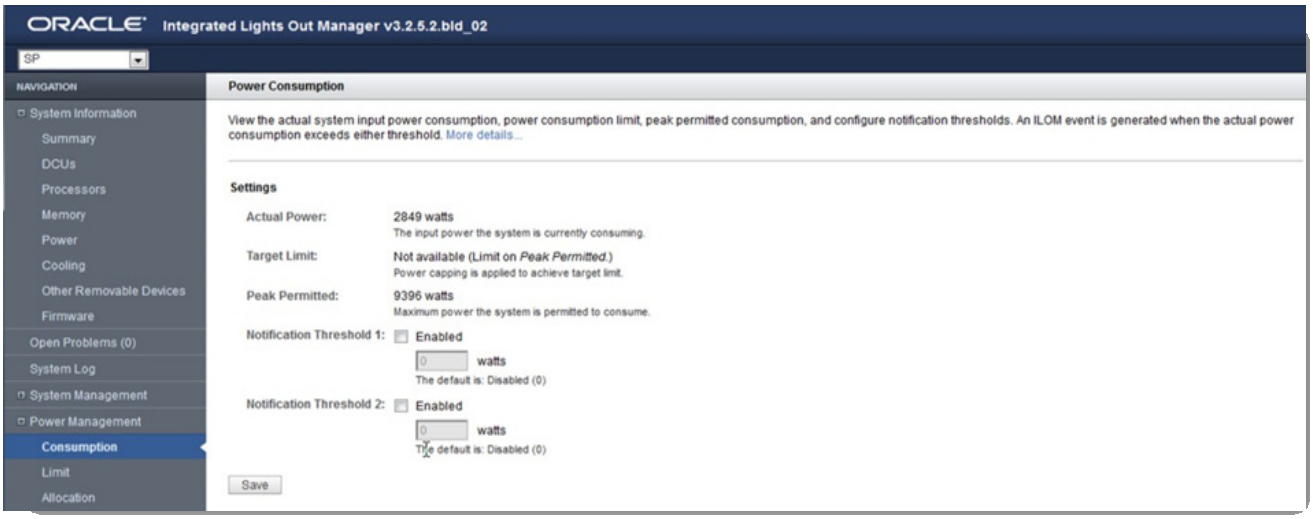
You must set the server altitude so that the server can adjust its fan speeds and monitor the surrounding environmental conditions required for its elevation. Set the server altitude by using the `SP system_altitude` property. This property is set to 200 meters by default.

Setting the `system_altitude` property causes the server to adjust temperature thresholds so that it can more accurately detect any abnormality in the air intake temperature. However, even if you do not set the system altitude, the server still detects and responds to any abnormality in the air temperature, such as the CMP temperature.

Appendix B

Managing Power

Viewing Power Consumption (Web Interface)



To view power consumption information by using the BUI, log in to the ILOM web interface. View the Power Management > Consumption page. The server's power consumption wattage value is displayed for the Actual Power and Peak Permitted Power properties. The consumption metric identifies the input power wattage that the server is currently consuming. The peak permitted power consumption metric identifies the maximum power wattage the server can consume.

You can view the power allocation requirements shown for the components. The power usage statistics can be seen from the Power Management > Statistics page. These statistics are displayed in 15, 30, and 60 second intervals. The per-component power map provides wattage allocations for each server component. Power history can be viewed from the Power Management > History page. The power history for the minimum, average, and maximum power usage is displayed.

Viewing Power State and Status (CLI)

To view the power state and status for all PDomains:

```
-> show /Servers/PDomains/ -level 2 -t power_state status
Target                                          I  Property     I Value
------------------------------------------------+---------------+---------------
/Servers/PDomains/PDomain_0/HOST                | power_state   | Off
/Servers/PDomains/PDomain_0/HOST                | status        | powered Off
/Servers/PDomains/PDomain_0/System              | power_state   | Off
/Servers/PDomains/PDomain_0/System/DCUs/DCU_0   | power_state   | Off
/Servers/PDomains/DCUs/DCU_0/CMIOU_0             | power_state   | Off
/Servers/PDomains/PDomain_0/System/DCUs/DCU_1   | power_state   | Off
/Servers/PDomains/PDomain_0/System/             | power_state   | Off
DCUs/DCU_1/CMIOU_4
<cont>
```

Viewing Power Consumption for the System (CLI)

- To view the power consumption and maximum power allowed:

```
-> show /System/Power
  /System/Power
  Targets:
    Power_Supplies

  Properties:
    health = OK
    health details
    actual_power_consumption = 6485 watts
    max_permitted_power = 16744 watts
    installed_power_supplies = 12
    max_power_supplies = 12
```

3

Domain Configuration

Objectives (1 of 2)

After completing this lesson, you should be able to:

- Explain the characteristics of a physical domain
- Locate the physical domains and Domain Configuration Units in the M7-8 and M7-16 systems
- Allocate DCUs to a domain
- Power on a domain
- Explain the structure of the Open Boot PROM (OBP) device tree

ORACLE

Objectives (2 of 2)

After completing this lesson, you should be able to:

- Access domain information by using standard Oracle Solaris OS utilities
- Monitor overall domain status
- List the three boot methods available on the M7 Systems
- Configure the boot disk

Additional Resources

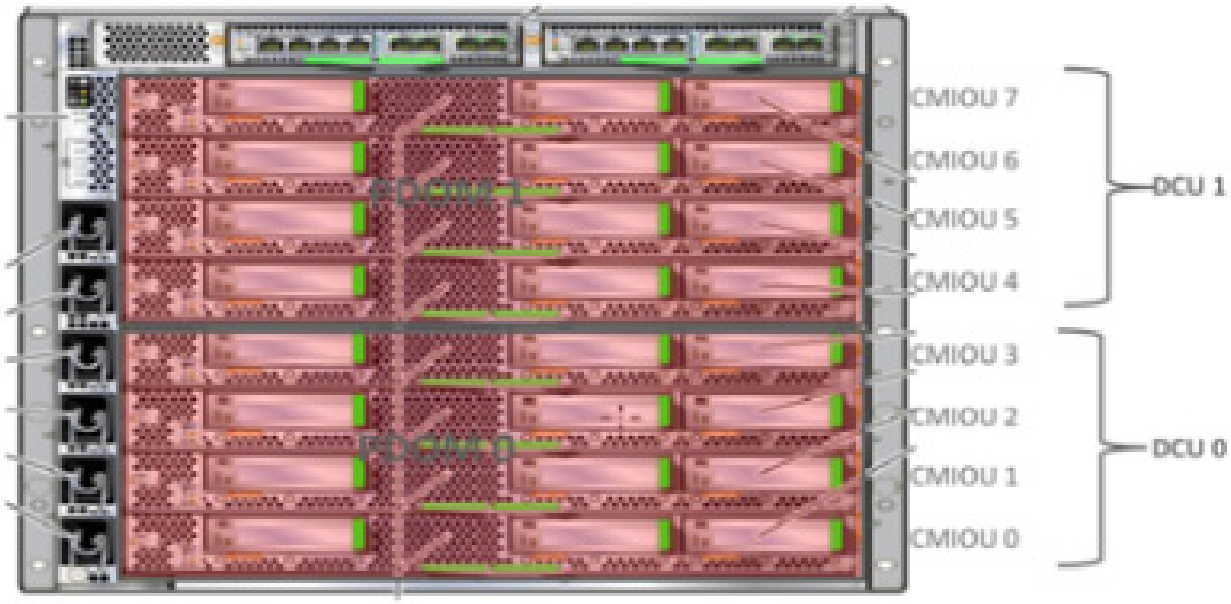
The following references provide additional information about the topics described in this lesson:

- M7-8 and M7-16 Technology Portal on OTN:
  https://www.oracle.com/servers/sparc/m7-8/index.html
  https://www.oracle.com/servers/sparc/m7-16/index.html
- M7-8 and M7-16 white papers:
  https://www.oracle.com/servers/sparc/resources.html

Terminology and Configuration

- A physical domain (PDomain or PDom) and a domain configurable unit (DCU) are logical units, not physical components, that you can configure depending on how you want to isolate applications and data to satisfy particular business requirements.
- On the M7 Systems, a PDom can have from two to four DCUs.
- On the M7 Systems, a DCU can have four to eight CMIOU slots.
- A CMIOU is a board that contains one CPU, 16 DIMM slots, and three PCIe generation 3 slots.

PDomains and DCUs on the Oracle SPARC M7-8 with Two PDOMs



Oracle SPARC M7-8 with two PDOMs uses a 2x4 socket configuration with two PDoms, as labeled in the diagram in the slide. Each of the two PDoms contains one DCU that consists of up to four CMIOUs in each DCU.

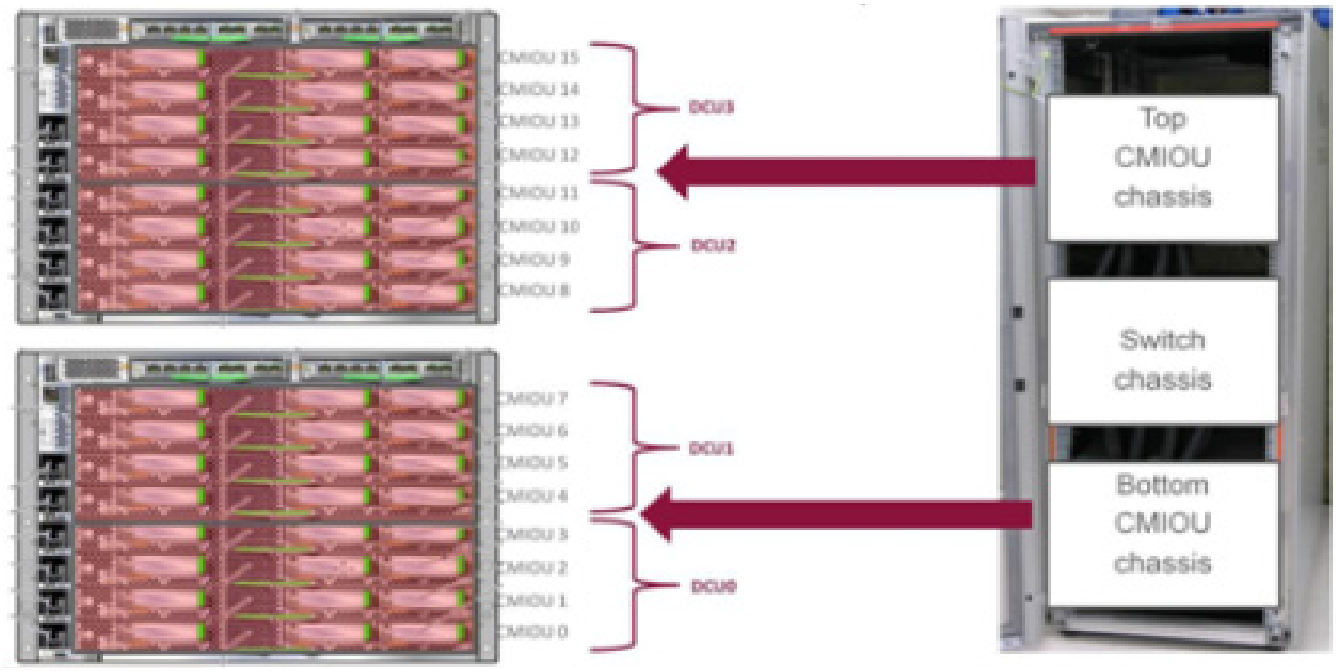
Note: For the population rules, refer to the *Oracle SPARC M7 Series Service Manual*.

PDomain and DCU on the Oracle M7-8 System

Oracle SPARC M7-8 uses a 1x8 socket configuration with one PDom that has one DCU, as labeled in the diagram in the slide. So the PDom is a single domain that contains from one to eight CMIOUs.

Note: For the population rules, refer to the *Oracle SPARC M7 Series Service Manual*.

PDomains and DCUs on the M7-16 System



Oracle SPARC M7-16 consists of four DCUs with up to four CMIOUs in each DCU.

Each PDom is represented as `/Servers/PDomains/PDomain_x/HOST` in Oracle ILOM, where x ranges from 0 to 3 (`PDomain_0`, `PDomain_1`, `PDomain_2`, `PDomain_3`), or simply `/HOSTx`.

Note: For the population rules, refer to the *Oracle SPARC M7 Series Service Manual*.

M7-16 Domain Configuration Unit (DCU)

DCU Managed	CMIOUs Assigned	Assigned SPM Pairs
DCU0	CMIOU0 - CMIOU3	/SYS/SPP0/SPM0 /SYS/SPP1/SPM0
DDCU1	CMIOU4 - CMIOU7	/SYS/SPP0/SPM1 /SYS/SPP1/SPM1
DCU2	CMIOU8 – CMIOU11	/SYS/SPP2/SPM0 /SYS/SPP3/SPM0
DCU3	CMIOU12 – CMIOU15	/SYS/SPP2/SPM1 /SYS/SPP3/SPM1

A SPARC M7-16 server is divided into four configurable units called DCUs. These DCUs have four CMIOUs each. Two of the DCUs are in the top CMIOU chassis, and two are in the bottom CMIOU chassis. You can configure DCUs into one to four PDomains.

Each CMIOU chassis also has two service processor proxies. SPP2 and SPP3 are in the top chassis, and SPP0 and SPP1 are in the bottom chassis. These SPPs have two service processor modules each. To achieve redundancy, each SPM (SPM0 and SPM1) on an SPP is assigned to a different DCU. SPM0 on each SPP manages one DCU in the CMIOU chassis, whereas SPM1 on each SPP manages the other DCU in the chassis.

Oracle ILOM identifies one of these SPMs as a DCU-SPP to manage DCU activity. The other SPM runs Oracle ILOM, but remains inactive unless the DCU-SPP can no longer manage the hardware. In this case, the inactive SPM assumes the role of DCU-SPP.

Oracle ILOM also identifies one of the DCU-SPPs from the pool of DCU-SPPs on the same PDomain as the PDomain-SPP to manage activity on that host.

Building Blocks on Other Systems

E2SK	M9000-32/64	MS/MG-32	M7
1 Uniboard + 11/0 Board	1 CPU+ 8 DIMMs+ 2 PCIe slots	4 CMU + 1 IOU	M7-8 with 2: PDOMs: 4 CMIOUs
	1 CMU + 1 IOU		M7-8 with 1 PDOM: 8 CMIOUs
			M7-16: 4 CMIOUs

The table in the slide shows comparisons on what building block equivalents are on Oracle's other high-end servers.

Assigning DCUs to a Physical Domain

- Check for available DCUs:

```
-> show /Servers/PDomains/PDomain_1/HOST deus_available
   /HOST1
      Properties:
         dcus_available = /SYS/DCU2 SYS/DCU3
```

- Make DCU assignable:

```
-> set /Servers/PDomains/PDomain_1/HOST deus_assignable="/SYS/DCU2"
   Set deus_assignable to /SYS/DCU2
```

- Assign the DCU:

```
-> set /Servers/PDomains/PDomain_1/HOST deus_assigned="/SYS/DCU2"
   Set deus_assigned to /SYS/DCU2
```

- Verify the DCU assignment:

```
-> show /Servers/PDomains/PDomain_1/HOST deus_assigned
   /HOST1
      Properties:
         deus_assigned = /SYS/DCU2
```

The examples in the slide show the syntax for assigning a DCU to a PDom. They require the user role `admin (a)` to be executed.

The `dcus_assignable` property enables you to control which DCUs can be assigned to a PDomain. Setting these variables overwrites what they are currently set to. If you want to revert to the current settings, you must include the current settings in the “set” syntax along with the new values.

Note that `/Servers/PDomains/PDomain_1/HOST` is the equivalent of `/HOST1`.

Unassigning DCUs from a Domain

- Check DCU host assignment:

```
-> show /System/DCUs/DCU_2 host_assigned
   /System/DCUs/DCU_2
      Properties:
         host_assigned = /HOST2
```

- Unassign the DCU:

```
-> set /Servers/PDomains/PDomain_1/HOST/ dcus_assigned=""
```

- Verify that the DCU was unassigned:

```
-> show /Servers/PDomains/PDomain_1/HOST dcus_assigned
   /Servers/PDomains/PDomain_1/HOST
      Properties:
         dcus_assigned = (none)
```


The examples in the slide show the syntax for unassigning a DCU from a physical domain or PDom. It requires the user role `admin (a)` to be executed.

Determining Current DCU Assignment

- Determine the PDom to which the DCU is assigned:

```
-> show /System/DCUs/DCUx host_assigned
   /System/DCUs/DCUx
      Properties:
         host_assigned = /HOSTy
```

- Determine which DCUs are assigned to a specific PDom:

```
-> show /Servers/PDomains/PDomain_x/HOST deus_assigned
   /HOSTx
      Properties:
         dcus_assigned = /SYS/DCUx
```


Each DCU is represented as `/System/DCUs/DCU_x` in Oracle ILOM where *x* ranges from 0 to 3 (`DCU_0`, `DCU_1`, `DCU_2`, `DCU_3`).

Listing the Resources That Are Part of a DCU

To list the resources that are part of a DCU:

```
-> show /System/DCUs/DCU_0
   /System/DCUs/DCU_0
      Targets:
         CMIOU_0
         CMIOU_1
         CMIOU_2
         CMIOU_3
      Properties:
         health = OK
         health details
         power state = On
         cpu summary = Four Oracle SPARC M7
         memory summary = 512 GB
         location = DCUO (Domain Configuration Unit 0)
         host assigned = /HOST0
         Fan list FMO/F0 (Fan Module 0), FMO/Fl (Fan Module 0), FMl/F0 (Fan Module 1), FMl/Fl (Fan Module l),
                  FM2/F0 (Fan Module 2), FM2/Fl (Fan Module 2), FM3/F0 (Fan Module 3), FM3/Fl (Fan Module 3),
                  FM4/F0 (Fan Module 4), FM4/Fl (Fan Module 4), FMS/F0 (Fan Module 5), FM5/Fl (Fan Module 5),
                  FM6/F0 (Fan Module 6), FM6/Fl (Fan Module 6), FM7/F0 (Fan Module 7), FM7/Fl (Fan Module 7),
   sp name = /SYS/SPl/SPM0
   initiate_sp_failover = (none)
<cont>
```


Each DCU is represented as `/System/DCUs/DCU_x` in Oracle ILOM where *x* ranges from 0 to 3 (`DCU_0`, `DCU_1`, `DCU_2`, `DCU_3`).

Assigning All DCUs to a Single Host (1 of 2)

- Assign all DCUs to a single host:

```
-> cd /Servers/PDomains/PDomain_0/HOST
-> set deus_assignable="/SYS/DCU0 /SYS/DCUl /SYS/DCU2 /SYS/DCU3”
-> set dcus_assigned="/SYS/DCU0 /SYS/DCUl /SYS/DCU2 /SYS/DCU3”
```

- If you receive a message that a DCU reconfiguration is in progress, continue to run this command until you get the following output:

```
-> show /Servers/PDomains/Pdomain_0/HOST operation_in_progress
   /Servers/PDomains/PDomain_0/HOST
      Properties:
         operation_in_progress = none
```


When the firmware is updated and the VLAN setup is complete, the DCU assignment can be done. This example shows the assignment of all DCUs to PDom 0.

Assigning All DCUs to a Single Host (2 of 2)

```
-> show /HOST0 deus_assignable
   deus_assignable = /SYS/DCU0 /SYS/DCU1 /SYS/DCU2 /SYS/DCU3
-> show /HOST0 deus_assigned
   deus_assigned = /SYS/DCU0,/SYS/DCU1
-> show /HOST0 deus_assignable deus_assigned deus_available
   /HOSTO
      Properties:
         deus_assignable = /SYS/DCU0 /SYS/DCU1 /SYS/DCU2 /SYS/DCU3
         deus_assigned = /SYS/DCU0 /SYS/DCU1
         deus_available = (none)
```

Assigning One DCU Per Host

```
-> cd /Servers/PDomains/PDomain_0/HOST
-> set deus_assignable="/SYS/DCU0 /SYS/DCUl /SYS/DCU2 /SYS/DCU3”
-> set deus_assigned="/SYS/DCU0"
-> cd /Servers/PDomains/PDomain_1/HOST
-> set deus_assignable="/SYS/DCU0 /SYS/DCUl /SYS/DCU2 /SYS/DCU3”
-> set deus_assigned="/SYS/DCUl"
-> cd /Servers/PDomains/PDomain_2/HOST
-> set deus_assignable="/SYS/DCU0 /SYS/DCUl /SYS/DCU2 /SYS/DCU3”
-> set deus=assigned="/SYS/DCU2"
-> cd /Servers/PDomains/PDomain_3/HOST
-> set deus_assignable="/SYS/DCU0 /SYS/DCUl /SYS/DCU2 /SYS/DCU3”
-> set deus_assigned="/SYS/DCU3"
```


In this example, you can see that one DCU is being assigned to each PDom.

Checking if the PDomain SPP Is Elected

- Verify the PDomain SPPs:

```
-> show /Servers/PDomains/PDomain_0/HOST sp_name
   /HOST0
      Properties:
         sp_name = /SYS/SPP0/SPM0

-> show /Servers/PDomains/PDomain_1/HOST sp_name
   /HOST1
      Properties:
         p_name = /SYS/SPP0/SPM1
```


One of the SPPs on each PDomain is identified as a PDomain-SPP, which manages tasks for its PDomain. The PDomain-SPP also hosts the rKVMS service for the PDomain. If a PDomain has more than one DCU, only the rKVMS service on the PDomain-SPP is used.

In a multi-DCU domain scenario, the SPP with the lowest number available in the domain will automatically become the PDomain SPP, which will manage the domain for all multiple DCUs. In case the Golden SPP fails, the next higher number SPP takes over.

Note: `/Servers/PDomains/PDomain_0/HOST` is the same as `/HOST0`.

Powering On a Host on an M7-8

- Powering on from the Oracle ILOM Web Interface:
  - Navigate to Summary > Actions panel.
  - Click the Power State turn on button.
- Access the console on an M7-8:

```
 -> start /Servers/PDomains/PDomain x/HOST/console
Are you sure you want to start /Servers/PDomains/PDomain_x/HOST/console (y/n) ? Y
Connecting /Servers/PDomains/PDomain_x/HOST/console
```

- Powering on an M7-8:

```
-> start /Servers/PDomains/PDomain_x/HOST
Are you sure you want to start all of the configured hosts on the system {y/n) ? Y
hostname starting
```

- On the M7-8 with 2 PDOMs, x = 0 or 1.
- On the M7-8 with 1 PDOM, x = 0.

Power-On States

- **No power applied:** No power is applied to the server. For example, when the power cords are not connected, or the datacenter power breaker is off.
- **Standby:** Power is applied to the server and the SP is running, but main power is not applied to the host. You can access Oracle ILOM on the SP in Standby state.
- **Fully powered on:** The host is powered on, and you can access Oracle ILOM. After the server boots the OS, you can access the operating system.

Note: Starting the console before starting the host is a good practice because it displays any errors it may encounter.

Powering On a PDomain on an M7-16

- Access the console on a PDomain 2 within an M7-16:

```
->start /Servers/PDomains/PDomain_2/HOST/console
Are you sure you want to start /Servers/PDomains/PDomain_2/HOST/console (y/n) ? Y
Connecting /Servers/PDomains/PDomain_2/HOST/console
```

- Power on PDomain 2 within an M7-16:

```
->start /Servers/PDomains/PDomain_2/HOST
Are you sure you want to start /Servers/PDomains/PDomain_2/HOST (y/n) ? Y
Starting /Servers/PDomains/PDomain_2/HOST
```

ORACLE

Note

- Access to the console may take up to 60 seconds.
- Starting the console before starting the host is a good practice because it displays any errors it may encounter.

Powering Off a Host on an M7-8

- Powering off from the Oracle ILOM Web Interface:
  - Shut down Oracle Solaris.
  - Navigate to Summary > Actions panel.
  - Click the Power State turn off button.
- Powering off an M7-8:

```
-> stop /Servers/PDomains/PDomain_x/HOST
Are you sure you want to stop /Servers/PDomains/PDomain_x/HOST (y/n)? Y
Stopping HOSTx
```

- On the M7-8 with 2 PDOMs, x = 0 or 1.
- On the M7-8 with 1PDOM, x = 0.

Powering Off a PDomain on an M7-16

- Change all PDomains to Standby Mode:

```
-> stop /System
Are you sure you want to stop /System (y/n)? Y
Stopping /System
```

- Change PDomain 2 to Standby Mode:

```
-> stop /Servers/PDomains/PDomain_2/HOST
Are you sure you want to stop /Servers/PDomains/PDomain_2/HOST (y/n)? Y
Stopping /Servers/PDomains/PDomain_2/HOST
```


Note: All these commands perform a graceful shut down. To perform an immediate shut down, add `-f` to the command line.

Displaying Console History

To manage the console history log:

```
-> set /Servers/PDomains/PDomain_x/HOST/console/history property=option [...]
-> show /Servers/PDomains/PDomain_x/HOST/console/history
```

Where, property can be:

- **`line count`**: This option accepts a value within the range of 1 through 2048 lines. Specify "" for an unlimited number of lines. The default is all lines.
- **`Pause_count`**: This option accepts a value of 1 to any valid integer or "" for infinite number of lines. The default is not to pause.
- **`start_from`**: The options are:
  - `end`: The last line (most recent) in the buffer (the default)
  - `beginning`: The first line in the buffer

Use the `/HOST/console/history` host server console output buffer to write all types of log information. If you enter the `show /HOST/console/history` command without first setting any arguments with the `set` command, Oracle ILOM displays all lines of the console log, starting from the end.

Note: Timestamps recorded in the console log reflect server time. These timestamps reflect local time, and the Oracle ILOM console log uses UTC (Coordinated Universal Time). The Oracle Solaris OS system time is independent of the Oracle ILOM time.

Monitoring Power-On Progress

/HOSTx/status	/HOSTx/status_details	Step Detail
Standby	Host status updated	Host power state is “Off.”
Starting	Step 1 of 9	All configurable components are being powered on, tested, and initialized.
Starting	Step 2 of 9	The set of configurable components is being passed to Guest Managers.
Starting	Step 3 of 9	Guest Managers are preparing to take CPUs out of reset.
Powered On	Step 4 of 9	Hostconfig and POST are running on the CPUs in each DCU in isolation.
Powered On	Step 5 of 9	Scalability ASICs are being programmed to reflect domain configuration and scalability links are being initialized (glued platforms only).
Powered On	Step 6 of 9	Hostconfig runs domainwide POST and prepares to launch Hypervisor (glued platforms only).
HW Started	Step 7 of 9	Hypervisor has been started.
HW Started	Step 8 of 9	The first guest logical domain has been started.
Host Started	Step 9 of 9	The first logical domain is running.
OpenBoot Initializing	Host status updated	OBP is initializing.
OpenBoot Running	Host status updated	OBP is running.
Solaris Booting	Host status updated	Solaris is booting.
Solaris Running	Host status update	Solaris is running.

The power on and boot progress can be monitored by using the following commands:

```
-> show /Servers/PDomains/PDomain_0/HOST status
```

or

```
-> show /HOST0 status
```

```
-> show /Servers/PDomains/PDomain_0/HOST status_detail
```

or

```
-> show /HOST0 status_detail
```

Additional Power-On Notes

For purposes of mutual exclusion among host commands, a “`start hostX`” command is considered to be complete when it has reached Step 7, HV Started.

Domain Boot Sequence

- M7 starts executing Hostconfig code from flash prom.
- Hostconfig does the following:
  - Initialization and configuration of CPUs, memory, and so on
  - Generates MDs and PRI for Hypervisor and Guest
  - Invokes diagnostics and applies platform policies to configure system around failed components
  - Jumps to Hypervisor on master CPU; others parked
- Hypervisor proceeds to:
  - Copy itself from ROM to RAM
  - Initializes itself based on HVMD
  - Starts the guest (Open Boot is the first guest.)
- OpenBoot probes I/0 devices based on MD and sets up the device tree for Solaris. It then starts the Solaris boot.

- Physical Resource Inventory (PRI)
- Hypervisor Machine Descriptions (HVMD)

POST: Knobs

- **`Default verbosity`**: Used to control the verbosity level of the output from POST diagnostics, if diagnostics are enabled
  - `none, min, `**`normal`**`, max and debug`
- **`Default_ Level`**: Used to control the level of diagnostic testing to be executed when diagnostics are enabled
  - **`off`**`, min, max`

Note: The bold setting is factory default.

Open Boot: Overview

From the OBP, you can:

- Boot the host
- Configure OBP parameters
- Run diagnostics (not recommended)
- Reset and power off the host (not recommended)

The OpenBoot firmware starts the OS and validates the installed hardware, and it can be used for other server administration tasks below the OS level. For more information about OpenBoot commands, see the *OpenBoot 4.x Command Reference Manual* in the Oracle Solaris documentation.

When the host is powered on, but the OS is not booted, you communicate with the OBP firmware. The OBP firmware displays `ok` as its prompt.

Getting to the OBP for a Physical Domain (1 of 2)

- To specify that the boot sequence should stop at the `ok` prompt:

```
-> set /Servers/PDomains/PDomain_x/HOST/bootmode script:"setenv auto-boot? False"
Set 'script' to 'setenv auto-boot? False'

-> show /Servers/PDomains/PDomain_x/HOST/bootmode
   /HOST2/bootmode
      Targets:

      Properties :
         config  = factory-default
         expires = (none)
         script = setenv auto-boot? false
         state = normal
```

- To start the host:

```
-> start /Servers/PDomains/PDomain_x/HOST
```

Getting to the OBP for a Physical Domain (2 of 2)

- To start the PDomain host console:

```
-> start /Servers/PDomains/PDomain_x/HOST/console
Are you sure you want to start /HOSTx/console (y/n)? y
Serial console started. To stop, type #.
```

- The server takes several minutes to complete POST, and then displays the `ok` prompt.

Setting OBP Variables

- To change the default boot device:
  `ok setenv boot-device boot_device`
- To verify the change:
  `ok printenv boot-device`
- To reset the host:
  `ok reset-all`
- To disable automatic booting:
  `ok setenv auto-boot? false`
- To view all of the OBP parameters and their settings:
  `ok printenv`
- To manually boot the operating system:
  `ok boot`

Time Synchronization

- To check if NTP is currently configured:

```
-> show /SP/clients/ntp/server/1
   /SP/clients/ntp/server/1
      Targets:
      Properties:
         address = ip_address
```

- To configure the system to synchronize with an NTP server:

```
-> set /SP/clients/ntp/server/1 address = aa.bb.cc.dd
-> set/SP/clock usentpserver=enabled
```

- You will also need to configure NTP within Solaris running in your domains.

When PDomains are powered on, their clocks synchronize to the NTP server when the system is configured to listen to NTP multicast (the default for the current Oracle Solaris OS). If the PDomains and SPs use the same NTP server, events logged in the Oracle Solaris OS and on the SP can be correlated based on their timestamps. If the PDomains and SPs use different NTP servers, their times might drift, and correlating log files could become difficult. If you connect a domain to an NTP server other than the one used by the SP, then ensure that both are low-stratum NTP servers that provide the same degree of accuracy.

Timestamps in the console log reflect server time. By default, the Oracle ILOM console log uses UTC/GMT, but you can use the `/Servers/PDomains/PDomain_x/SP/clock` time zone command to set the SP clock to use other time zones. The Oracle Solaris OS system time is independent of the Oracle ILOM time.

Getting Domain Information from Solaris

```
I prtdiag -v
System Configuration:   Oracle Corporation   sun4v SPARC M7-8 with 2 POOMs
Memory size: 521728 Megabytes
-----------Virtual CPUs-----------------------------------------------------
CPU ID  Frequency  Implementation  Status
----------------------------------------------
1024    4133 MHz   SPARC-M7        on-line
1025    4133 MHz   SPARC-M7        on-line
1026    4133 MHz   SPARC-M7        on-line
1027    4133 MHz   SPARC-M7        on-line
<output omitted>
2047     4133 MHz   SPARC-M7         on-line
----------------------Physical Memory Configuration-----------------------
Segment Table:
--------------------------------------------------------------------------
Base      Segment   Interleave    Bank    Contains
Address   Size      Factor        Size    Modules
--------------------------------------------------------------------------
OxO       128 GB                  128GB   /SYS/CMl0U4/CM/CMP/BOB00/CH0/DIMM
                                          /SYS/CM10U4/CM/CMP/BOB0l/CH0/DIMM
                                          /SYS/CMIOU4/CM/CMP/BOB10/CH0/DIMM
                                          /SYS/CM10U4/CM/CMP/BOBll/CH0/DIMM
                                          /SVS/CM10U4/CM/CMP/BOB20/CH0/DIMM
                                          /SYS/CMIOU4/CM/CMP/BOB21/CH0/DIMM
                                          /SYS/CM10U4/CM/CMP/BOB30/CH0/DIMM
                                          /SYS/CMIOU4/CM/CMP/BOB3l/CH0/DIMM
```

Gathering Domain Information from Solaris

```
0x400000000000    128 GB  0    128 GB    /SYS/CMIOU5/CM/CMP/BOB00/CH0/DIMM
                                         /SYS/CMIOU5/CM/CMP/BOB0l/CH0/DIMM
                                         /SYS/CMIOU5/CM/CMP/BOBl0/CH0/DIMM
                                         /SYS/CMIOU5/CM/CMP/BOBll/CH0/DIMM
                                         /SYS/CMIOU5/CM/CMP/BOB20/CH0/DIMM
                                         /SYS/CMIOU5/CM/CMP/BOB21/CH0/DIMM
                                         /SYS/CMIOU5/CM/CMP/BOB30/CH0/DIMM
                                         /SYS/CMIOU5/CM/CMP/BOB31/CH0/DIMM
0x800000000000    128 GB  0    128 GB    /SYS/CMIOU6/CM/CMP/BOB00/CH0/DIMM
                                         /SYS/CMIOU6/CM/CMP/BOB01/CH0/DIMM
                                         /SYS/CMIOU6/CM/CMP/BOB10/CH0/DIMM
                                         /SYS/CMIOU6/CM/CMP/BOB11/CH0/DIMM
                                         /SYS/CMIOU6/CM/CMP/BOB20/CH0/DIMM
                                         /SYS/CMIOU6/CM/CMP/BOB21/CH0/DIMM
                                         /SYS/CMIOU6/CM/CMP/BOB30/CH0/DIMM
                                         /SYS/CMIOU6/CM/CMP/BOB31/CH0/DIMM
0xc00000000000    128 GB  0    128 GB    /SYS/CMIOU7/CM/CMP/BOB00/CH0/DIMM
                                         /SYS/CMIOU7/CM/CMP/BOB0l/CH0/01MM
                                         /SYS/CMIOU7/CM/CMP/BOBl0/CH0/DIMM
                                         /SYS/CMIOU7/CM/CMP/BOBll/CH0/01MM
                                         /SYS/CMIOU7/CM/CMP/80820/CH0/DIMM
                                         /SYS/CMIOU7/CM/CMP/BOB21/CH0/DIMM
                                         /SYS/CMIOU7/CM/CMP/BOB30/CH0/DIMM
                                         /SYS/CMIOU7/CM/CMP/BOB31/CH0/DIMM
```

Gathering Domain Information from Solaris

```
========================= 10 Devices ===================================
Slot +                     Bus  Name+ Model                  MaxSpeed  CurSpeedStatus    Type Path/Width/Width
/SYS/CMIOU4/IOH/IOS4/RP0 PCIE usb-pciexctass,Oc0330           5.0GT/xl  5.0GT/xl  /pci@318/pci@l/pci@0/pci@8/usb@0
/SYS/CMIOU4/IOH/IOS4/RP0 PCIE display-pciex102b,522           2.5GT/x1  2.5GT/x1  /pci@318/pci@l/pci@0/pci@c/d1splay@0
/SYS/CMIOU4/USB_CTRL     PCIE usb-pciexclass,Oc0330           5.0GT/x1  5.0GT/x1  /pci@318/pci@2/usb@0
/SYS/CMIOUS/IOH/10S1/RP0 PCIE network-pciex8086,105e          2.5GT/x4  2.5GT/x4  /pci@31a/pci@1/network@0
                            SUNW,pcie-northstar
/SYS/CMIOU5/IOH/10S1/RP0 PCIE network-pciex8086,105e          2.5GT/x4  2.5GT/x4  /pci@31a/pci@1/network@0,l
                            SUNW,pcie-northstar
/SYS/CMIOU5/IOH/10S3/RP0 PCIE LSl,sas-pciexl000,72            5.0GT/K8  5.0GT/x8  /pci@3lc/pci@l/LSl,sas@0
                            LSl,2008
/SYS/CMIOU5/IOH/IOS4/RP0 PCIE usb-pciexclass,Oc0330           5.0GT/x1  5.0GT/x8  /pci@31d/pci@l/pci@0/pci@8/usb@0
/SYS/CM10U5/IOH/10S4/RP0 PCIE di5play-pciex102b,522           5.0GT/xl  5.0GT/xl  /pci@31d/pci@l/pci@0/pci@c/disptay@0
/SYS/CMIOU5/USB_CTRL     PCIE usb-pciexclass,Oc0330           2.5GT/xl  2.5GT/xl  /pci@31d/pci@2/usb@0
/SYS/CM10U6/IOH/10S3/RP0 PCIE SUNW,qle-pciexl077,8030 7023303 8.0GT/x8  8.0GT/x8  /pci@321/pci@l/SUNW,qle@0
/SYS/CMIOU6/IOH/IOS3/RP0 PCIE SUNW,qle-pciex1077,8030 7023303 8.0GT/x8  8.0GT/x8  /pci@321/pci@l/SUNW,qle@0,l
/SYS/CMIOU6/IOH/IOS3/RP0 PCIE SUNW,qlc-pciexl077,8031 QLE8362 8.0GT/x8  8.0GT/x8  /pci@321/pci@l/SUNW,qlc@0,2
/SYS/CM10U6/I0H/10S3/RP0 PCIE SUNW,qlc-pciexl077,8031 QLE8362 8.0GT/x8  8.0GT/x8  /pci@321/pci@l/SUNW,qlc@0,3
/SYS/CMIOU6/USB_CTRL     PCIE usb-pciexclass,Oc0330           5.0GT/xl  5.0GT/xl  /pci@322/pci@2/usb@0
/SYS/CMIOU7/USB_CTRL     PCIE usb-pciexclass,Oc0330           5.0GT/xl  5.0GT/xl  /pci@327/pci@2/usb@0
```

Gathering Domain Information from Solaris

```
------ Environmental Status ------
Fan sensors:
--------------------------------
Location      Sensor    Status
--------------------------------
SYS/FM6/F0    TACH      OK
SYS/FM6/Fl    TACH      OK
SYS/FMl/F0    TACH      OK
SYS/FMl/Fl    TACH      OK
SYS/FMS/F0    TACH      OK
SYS/FMS/Fl    TACH      OK
SYS/FM0/F0    TACH      OK
SYS/FM0/Fl    TACH      OK
SYS/PSl/F0    TACH      OK
SYS/PSl/Fl    TACH      OK
SYS/FM4/F0    TACH      OK
SYS/FM4/Fl    TACH      OK
SYS/PS3/F0    TACH      OK
SYS/PS3/Fl    TACH      OK
SYS/PS5/F0    TACH      OK
SYS/PS5/Fl    TACH      OK
SYS/PS0/F0    TACH      OK
SYS/PS0/Fl    TACH      OK
SYS/FM3/F0    TACH      OK
SYS/FM3/Fl    TACH      OK
SYS/PS2/F0    TACH      OK
SYS/PS2/Fl    TACH      OK
<output omitted:>
```

Gathering Domain Information from Solaris

```
Temperature sensors:
----------------------------------------------------
Location Sensor                       Sensor    Status
----------------------------------------------------
SYS/CMIOU4/CM/CMP/BOB00/CH0/DIMM    T_AMB     OK
SYS/CMIOU4/CM/CMP/BOB00             T_BOB     OK
SYS/CMIOU4/CM/CMP/BOB01/CH0/DIMM    T_AMB     OK
SYS/CMIOU4/CM/CMP/BOB01             T_BOB     OK
SYS/CMIOU4/CM/CMP/BOB10/CH0/DIMM    T_AMB     OK
SYS/CMIOU4/CM/CMP/BOB10             T_BOB     OK
SYS/CMIOU4/CM/CMP/BOB11/CH0/DIMM    T_AMB     OK
SYS/CMIOU4/CM/CMP/BOB11             T_BOB     OK
SYS/CMIOU4/CM/CMP/BOB20/CH0/DIMM    T_AMB     OK
SYS/CMIOU4/CM/CMP/BOB20             T_BOB     OK
SYS/CMIOU4/CM/CMP/BOB21/CH0/DIMM    T_AMB     OK
SYS/CMIOU4/CM/CMP/BOB21             T_BOB     OK
SYS/CMIOU4/CM/CMP/BOB30/CH0/DIMM    T_AMB     OK
SYS/CMIOU4/CM/CMP/BOB30             T_BOB     OK
SYS/CMIOU4/CM/CMP/BOB31/CH0/DIMM    T_AMB     OK
SYS/CMIOU4/CM/CMP/BOB31             T_BOB     OK
SYS/CMIOU4/CM/CMP                   T_INT     OK
SYS/CMIOU4/CM                       T_AMB     OK
```

<output omitted>

Gathering Domain Information from Solaris

```
Current sensors:
-----------------------------------------------
Location                      Sensor   State
-----------------------------------------------
SYS/CMIOU4/CM/VCORE_PMBUSO    I_OUT    OK
SYS/CMIOU4/CM/VCORE_PMBUS1    I_OUT    OK
SYS/CMIOU4/CM/VCORE_PMBUS2    I_OUT    OK
SYS/CMIOU4/CM/VCORE_PMBUS3    I_OUT    OK
SYS/CMIOU4/CM/VDDI0_0BPS      I_OUT    OK
SYS/CMIOU4/CM/VDDSOC_PMBUS    I_OUT    OK
SYS/CMIOU4/CM/VDDT_0BPSO      I_OUT    OK
SYS/CMIOU4/CM/VDDT_0BPS1      I_OUT    OK
SYS/CMIOU4/CM/VDDT_0BPS2      I_OUT    OK
SYS/CMIOU4/CM/VDDT_0BPS3      I_OUT    OK
SYS/CMIOU4/CM/VDDT_0BPS4      I_OUT    OK
SYS/CMIOU4/CM_ECB             I_OUT    OK
SYS/CMIOU4/ECB                I_OUT    OK

<output omitted>
```

Powering Off a Domain

- To power off a specific PDomain (PDomain_2):

```
-> stop /Servers/PDomains/PDomain_2/HOST
Are you sure you want to stop H0ST2(y/n) ? Y
Stopping /H0ST2
```

- To force an immediate shutdown:

```
-> stop -f /Servers/PDomains/PDomain_2/HOST
Stopping /H0ST2
```

- To verify that the PDom is powered off:

```
-> show /Servers/PDomains/PDomain_2/HOST status
   /HOST2
      Properties:
         status = Powered Off
```


Note: To perform an immediate shut down, use the `-force` option from the `stop` command. Ensure that all data is saved before entering this command. To perform power operations on the server or a specific domain, user accounts on each must be assigned `reset (r)` user roles.

Performing Power Operations for All Domains

- To perform power operations for all domains:

```
-> start/System
   Are you sure you want to start /System (y/n)? Y
   /HOST0: Starting
   /HOSTl: Starting
   /HOST2: Starting
   /H0ST3: Starting
-> stop/System
   Are you sure you want to stop /System (y/n)? Y
   Stopping /HOST0
   Stopping /HOSTl
   Stopping /HOST2
   Stopping /HOST3
-> reset/System
```

You can start, stop, or reset the whole system or an individual PDomain. To perform these tasks, user accounts for the components that you want to `start`, `stop`, or `reset` must be assigned `reset (r)` user roles.

Boot Options on an M7-8 or M7-16

Three boot options are available:

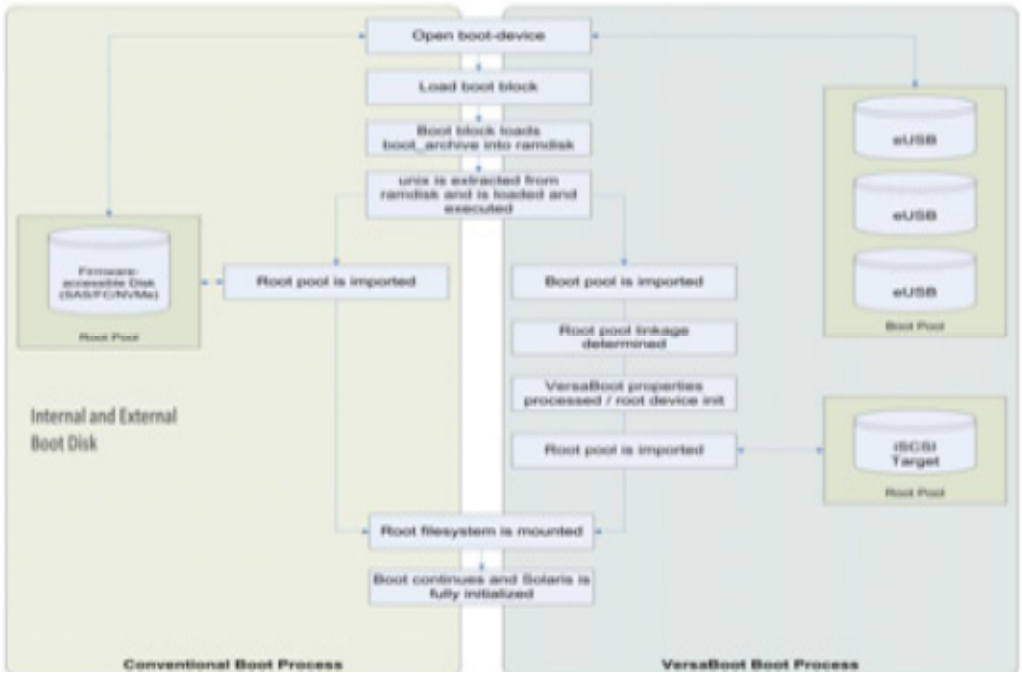
- **Internal Boot Disk:** Install an Aura3 card in one of the three PCIe generation 3 slots on the CMIOU boards and use it as a boot disk. The Aura3 is a Non-volatile Memory Express (NVMe) controller and a Solid-State Disk (SSD).
- **Conventional External Boot Disk:** Install an adapter to one of the three PCIe generation 3 slots on the CMIOU boards and use to connect to external storage.
  - 16 Gb FC to FC SAN attached Storage using Ganymede-Q or Ganymede-E
  - 8 Gb FC to FC SAN attached Storage using Pallene-Q or Pallene-E
  - iSCSI to iSCSI attached Storage using Powerville-UTP, Powerville-MMF, Twinville or Niantic

- **VersaBoot and eUSB Disk:** A VersaBoot image is stored on the embedded USB device that is on every CMIOU board.

Note

- The `VersaBoot/eUSB` option is new; therefore, refer to the Product Notes to determine its status as the product matures.
- VersaBoot uses iSCSI-over-InfiniBand as well as IP-over-InfiniBand (ISCSI/IPoIB).

Conventional Versus VersaBoot



The VersaBoot boot process supports a boot pool consisting of embedded USB or eUSB flash-based disk device that is located in each CMIOU.

Configuring the Flash Accelerator

- List the network devices:

```
ok show-nets
a) /pci@306/pci@1/network@0,3
b) /pci@306/pci@1/network@0,2
c) /pci@306/pci@1/network@0,1
d) /pci@306/pci@1/network@0
q) NO SELECTION
Enter Selection, q to quit: q
```

- Set the network device:

```
ok nvalias net /pci@306/pci@1/network@0
ok nvstore
```

- List the installed flash accelerator cards:

```
ok probe-nvme-all
/pci@301/pci@1/nvme@0
   NVME Controller VID: 8086 SSVID: 108e SN: CVMD4264006Y1P6K MN: INTEL SSDPEDME016T4S
FR:8DV1RA07
   NN: 1 Namespace ID:1 Size: 1.600 TB
```

- Configure the Flash Accelerator as a boot device:

```
ok setenv boot-device /pci@301/pci@1/nvme@0/disk@1:a
```

- Boot the system from the network using the Oracle Solaris OS install service:

```
ok boot net:dhcp - install
```


Note: This configuration is covered in the *SPARC M7 Series Servers Installation Guide*.

Summary (1 of 2)

In this lesson, you should have learned how to:

- Explain the characteristics of a physical domain
- Locate the physical domains and Domain Configuration Units in the M7-8 and M7-16 systems
- Allocate DCUs to a domain
- Power on a domain
- Explain the structure of the Open Boot PROM (OBP) device tree

Summary (2 of 2)

In this lesson, you should have learned how to:

- Access domain information by using standard Oracle Solaris OS utilities
- Monitor overall domain status
- List the three boot methods available on the M7 Systems
- Configure the boot disk

4

Logical Domain Configuration

Objectives

After completing this lesson, you should be able to:

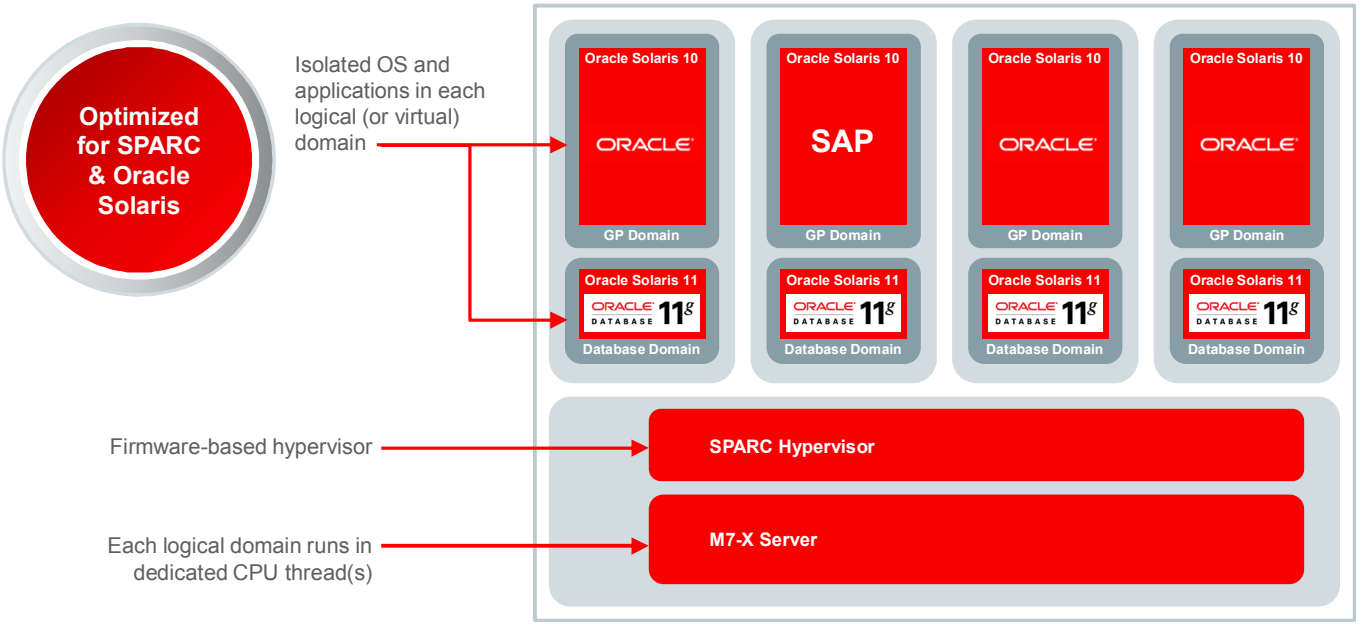
- Define LDoms
- Describe logical domain planning considerations
- Create LDoms
- Add resources to LDoms
- Dynamically change LDoms
- Delete LDoms
- Monitor LDoms

Additional Resources

The following references provide additional information about the topics described in this lesson:

- M7-X Technology Portal on OTN:
  https://www.oracle.com/servers/sparc/m7-8/index.html
  https://www.oracle.com/servers/sparc/m7-16/index.html
- M7-X white papers:
  https://www.oracle.com/servers/sparc/resources.html
- Oracle VM Server for SPARC (LDoms) Resource Center [ID 1367157.1]:
  http://support.oracle.com

Oracle VM Server for SPARC



Oracle VM Server for SPARC, also known as logical domains, provides highly efficient, enterprise-class virtualization capabilities for Oracle's SPARC M-Series servers.

Oracle VM Server for SPARC leverages the built-in Hypervisor to subdivide system resources (CPUs, memory, network, and storage) by creating partitions called logical (or virtual) domains. Each logical domain can run an independent operating system.

Oracle VM Server for SPARC provides the flexibility to deploy multiple Oracle Solaris operating systems simultaneously on a single platform. This virtualization solution has been designed to fully optimize Oracle Solaris and SPARC for enterprise server workloads.

Terminology

- Logical domain (LDom): A set of virtualized resources (CPU, memory, 1/0 devices) that are allowed mutual access, creating a sun4v virtual machine
- Guest software: Any piece of software running in the sun4v virtual machine environment
  - It is opaque to Hypervisor (HV) in the sun4v architecture. By design, HV is agnostic to the software that is executing in a logical domain; the only requirement is that the guest software comply with the sun4v specification.
- Control Domain: A logical domain whose distinguishing feature is access to the HV control channel, allowing it to send reconfiguration requests to the HV
  - There is one control domain per physical domain. The M7-16 can support up to four physical domains and thus up to four control domains.

The primary roles of Hypervisor are:

- To implement the software components of the sun4v virtual machine, providing low overhead hardware abstraction
- To enforce hardware and software resource access restrictions for guest, including inter-LDom communication,
- To provide isolation and security.

It also performs initial triage and correction of hardware errors.

The secondary roles of Hypervisor include implementing dynamic LDom reconfiguration, providing data for performance statistics, and managing the hardware elements of some power management features.

Logical Domains

The roles that logical domains can perform are:

- Control domain
- Service domain
- I/O domain
- Root domain
- Guest domain

- **Control domain:** The LDoms Manager runs in this domain, which enables you to create and manage other logical domains, and to allocate virtual resources to other domains. You can have only one control domain per PDom. The control domain is the first domain that is created when you install the Oracle VM Server for SPARC software. The control domain is named primary.
- **Service domain:** A service domain provides virtual device services to other domains, such as a virtual switch, a virtual console concentrator, and a virtual disk server. You can have more than one service domain, and any domain can be configured as a service domain.
- **I/O domain:** An I/O domain has direct access to a physical I/O device, such as a network card in a PCI EXPRESS (PCIe) controller. An I/O domain owns a PCIe root complex. An I/O domain can share physical I/O devices with other domains in the form of virtual devices when the I/O domain is also used as a service domain.
- **Root domain:** A root domain has a PCIe root complex assigned to it. This domain owns the PCIe fabric, and provides all fabric-related services, such as fabric error handling. A root domain is also an I/O domain, because it owns and has direct access to physical I/O devices. The number of root domains that you can have depends on your platform architecture.
- **Guest domain:** A guest domain is a non-I/O domain that consumes the virtual device services that are provided by one or more service domains. A guest domain does not have any physical I/O devices, but only has virtual I/O devices, such as virtual disks and virtual network interfaces.

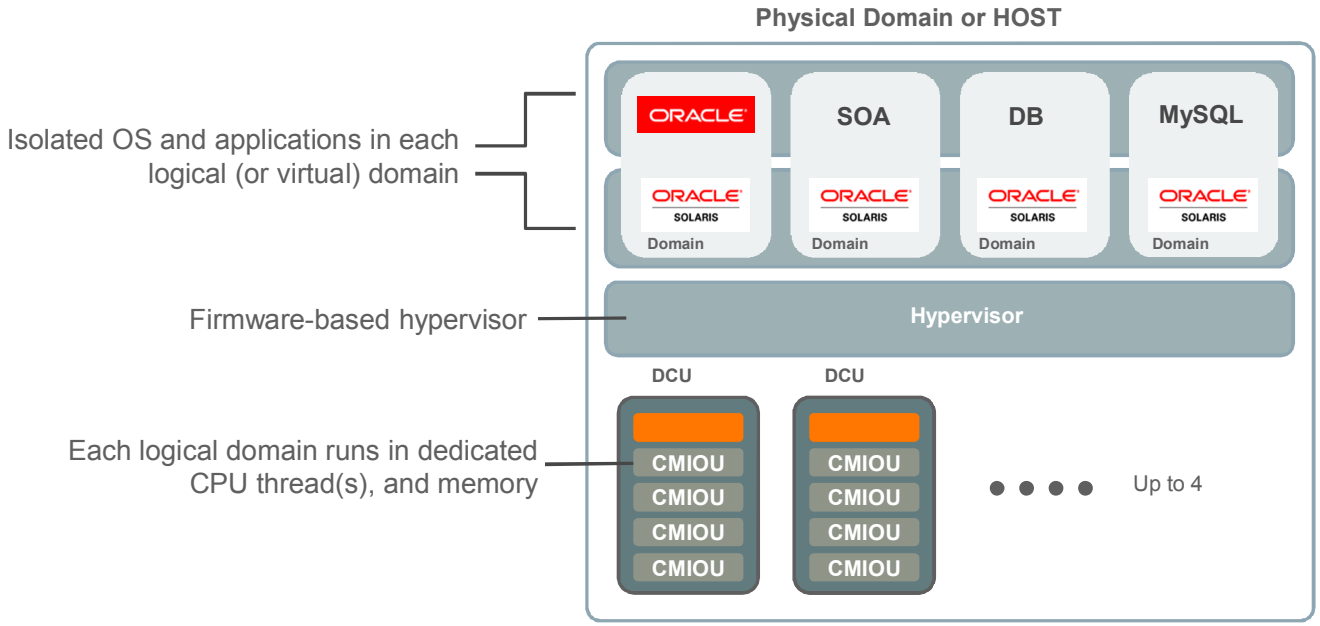
Hypervisor and Logical Domains

Oracle VM for SPARC

- A *logical domain* (LDom) is a virtual machine that comprises a discrete logical grouping of resources.
- Each LDom runs its own instance of Solaris.
- Each LDom can be created, destroyed, reconfigured, and rebooted independently.
- The Hypervisor enforces partitioning of the server's resources, and the OS and applications running in those partitions (that is, LDoms).
- The Hypervisor allocates a subset of the overall CPU, memory, and I/O resources of a server to a given logical domain.
- There can be up to 128 logical domains per Hypervisor.

The Hypervisor software is responsible for maintaining the separation between logical domains. The Hypervisor software also provides logical domain channels (LDCs) that enable logical domains to communicate with each other. LDCs enable domains to provide services to each other, such as networking or disk services.

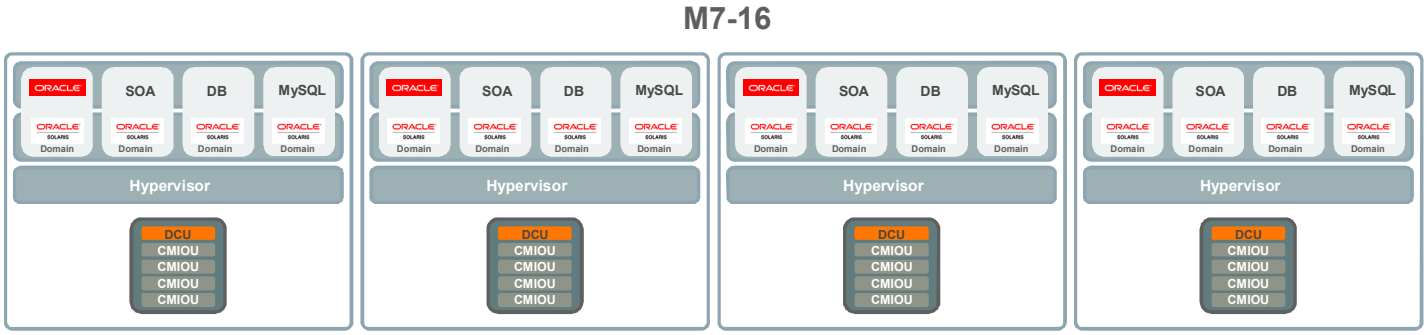
Oracle VM Server for SPARC: M7 Servers



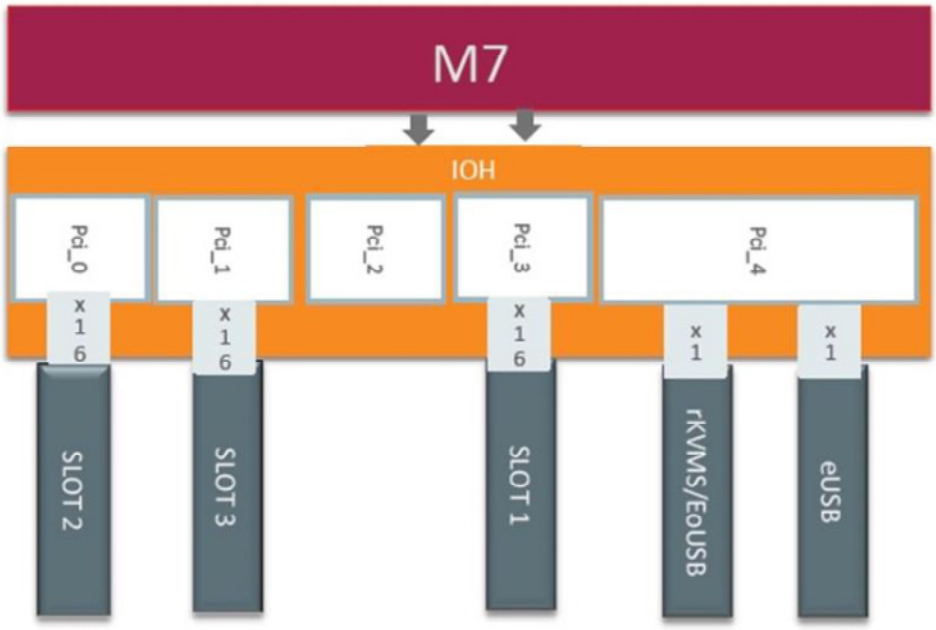
The maximum number of physical domains is four and the minimum number of DCUs per physical domain is one. Each physical domain has its own Hypervisor.

Maximum Number of Physical Domains

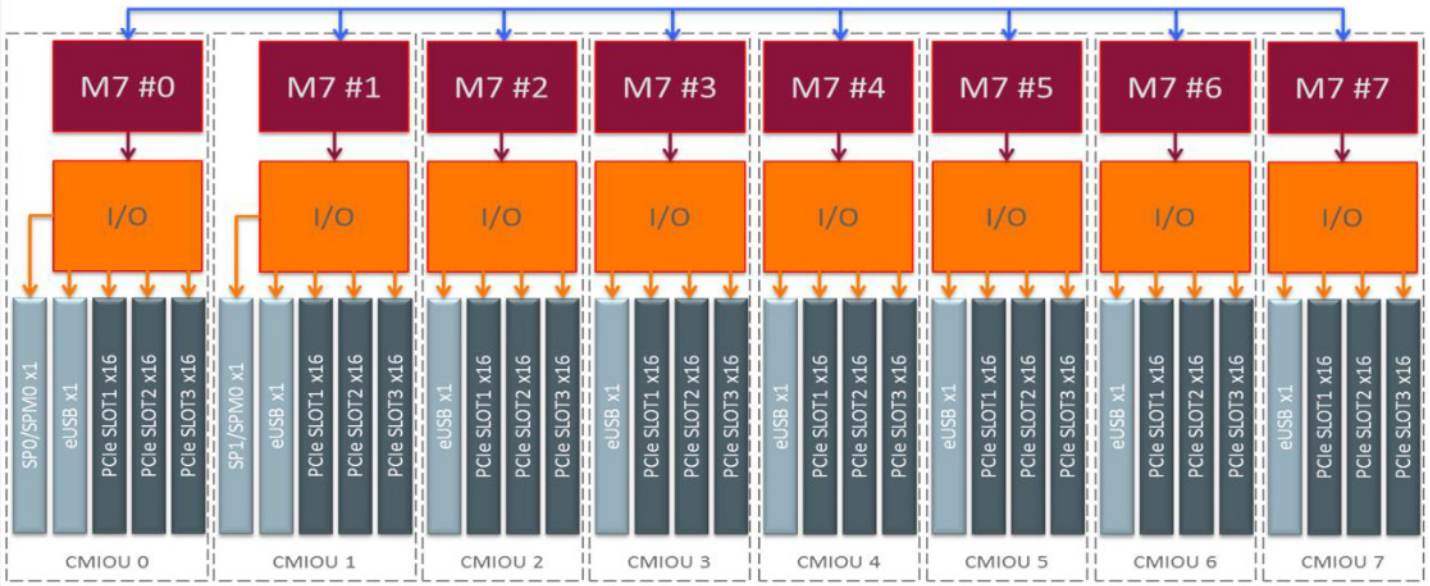
One DCU per physical domain



M7 CMIOU Device Map

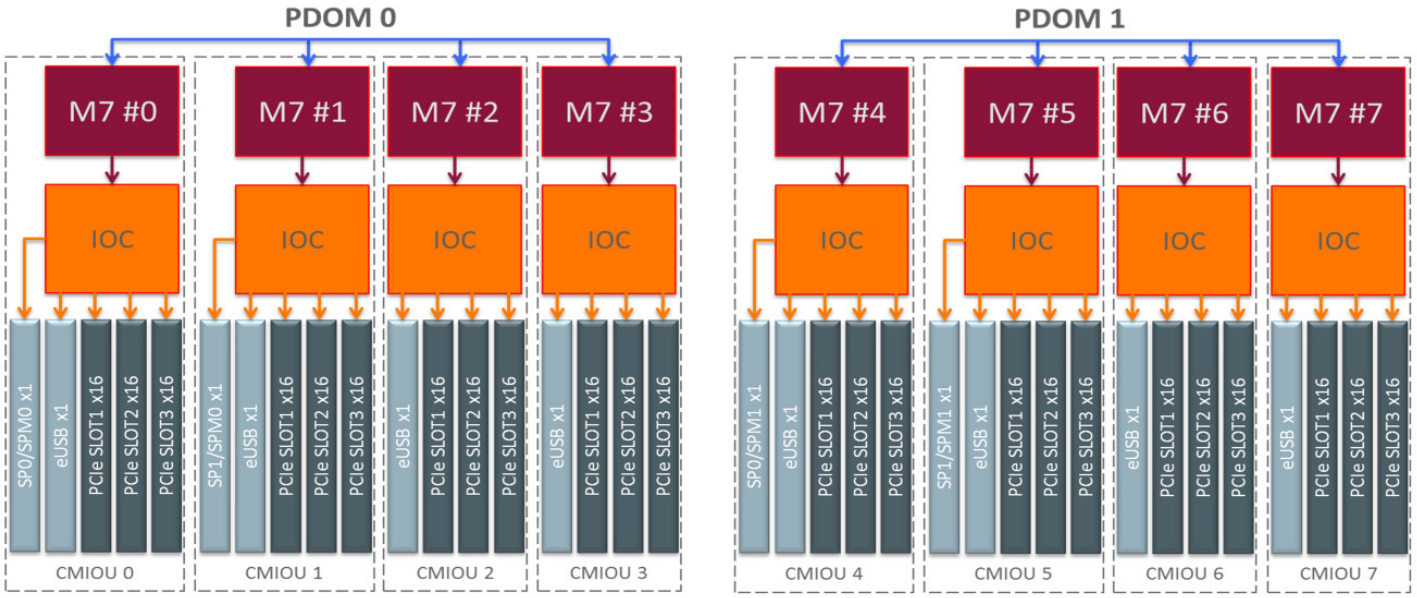


M7-8 Device Map



The slide shows the device map for an M7-8 server with one PDomain.

Device Map of M7-8 with Two PDoms



The slide shows the device map for an M7-8 server with two PDomains.

Device Map of M7-16 with Three PDoms

The slide shows the device map for an M7-16 server with three PDomains.

No “I/O Reconfiguration”

- The previous generations of multisocket M series systems supported a feature called “I/O Reconfiguration.”
  - At boot time, the host firmware could reconfigure the I/O hardware in case a CMT node completely failed.
  - The connections were rerouted to the on-board devices and slots so that they could be reached by the surviving CMT nodes.
- PCIe fabric is now hosted by the M7 I/O Controller.
  - The fabric can remain unchanged even if a CMT node were added or removed.
  - There is no longer a need for the “PCI device paths” to change based on what processors are present, which means that the PCI device paths always remain the same.

The features of the M7 I/O Controller include:

- Four Root Complexes of 16 lanes each, where each group can be individually configured as 1x16, 2x8, or 4x4 Root Ports, that is, bifurcated and quadfurcated
- One Root Complex of 8 lanes that can be configured as 1x8 or 2x4 Root Ports

LDom: Example

The following layers make up the logical domains functionality:

- User/services or applications
- Kernel or operating systems
- Firmware or Hypervisor
- Hardware, including CPU, memory, and I/O

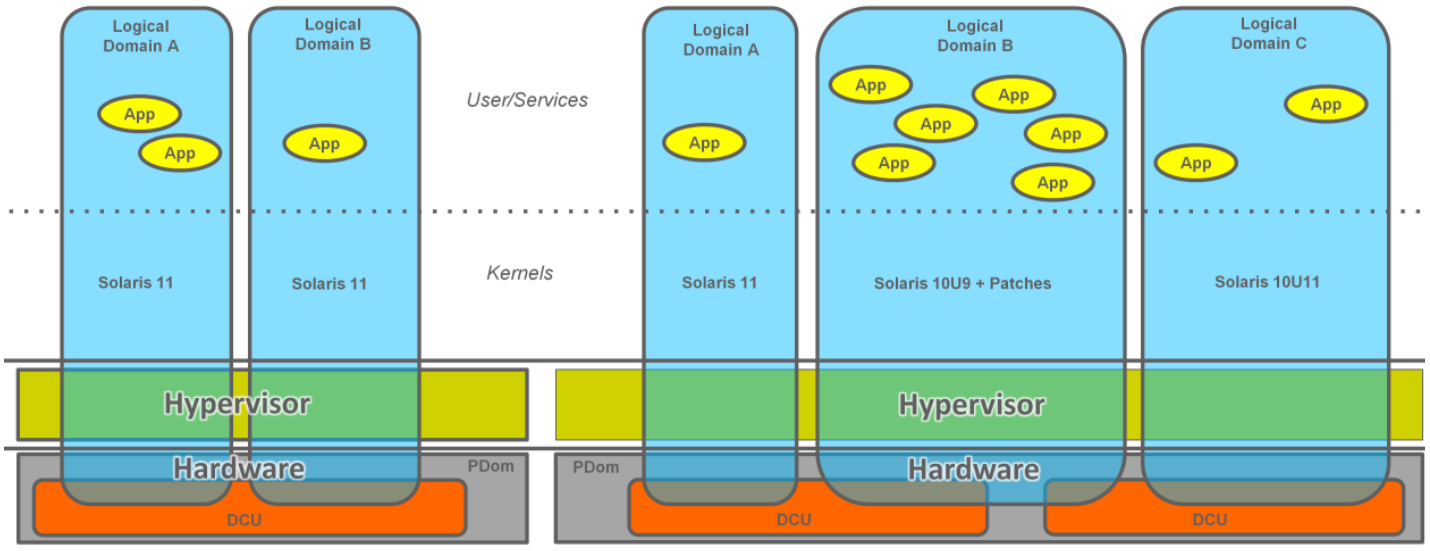
Each logical domain is permitted to observe and interact only with those server resources that are made available to it by the Hypervisor. The LDoms Manager enables you to specify what the Hypervisor should do through the control domain. Thus, the Hypervisor enforces partitioning of the server's resources, and provides limited subsets to multiple operating system environments. This partitioning and provisioning is the fundamental mechanism for creating logical domains.

Minimum Solaris Support on M7 Servers

The M7 requires the following minimum OS system versions:

- Control, Root, 1/0 Domain
  - Solaris 11.3
- Guest Virtual 10 guest domains
  - S11U3
  - S10U9, S10U10, S10U11
    - Patch list: Reference Doc ID 1967047
- LDoms 3.3

LDoms per PDom



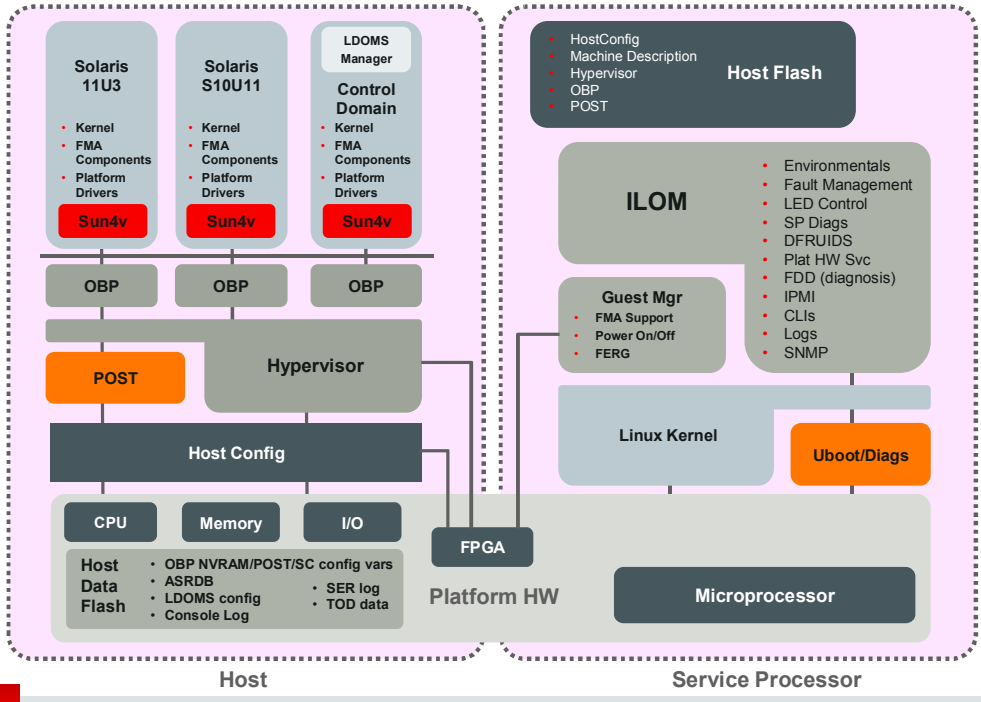
The diagram in the slide shows the Hypervisor supporting two logical domains in the PDom on the left and three logical domains in the PDom on the right.

The number and capabilities of each logical domain that a specific SPARC Hypervisor supports are server-dependent features. The Hypervisor can allocate subsets of the overall CPU, memory, and I/O resources of a server to a given logical domain. This enables support of multiple operating systems simultaneously, each within its own logical domain. Resources can be rearranged between separate logical domains with arbitrary granularity. For example, CPUs are assignable to a logical domain with the granularity of a CPU thread.

Each logical domain can be managed as an entirely independent machine with its own resources, such as Kernel, patches, and tuning parameters; user accounts and administrators; disks, network interfaces, MAC addresses, and IP addresses.

Each logical domain can be stopped, started, and rebooted independently of each other without requiring you to perform a power cycle of the server.

Software/Firmware Block Diagram



There are three types of Machine Descriptions (MDs): Platform MD (aka mini-MD), Guest MD(s), and Hypervisor MD (HVMD).

The Platform MD is generated by Hostconfig during power-on-reset (POR) operations. It contains low-level platform configuration information that is intended for Hypervisor consumption only.

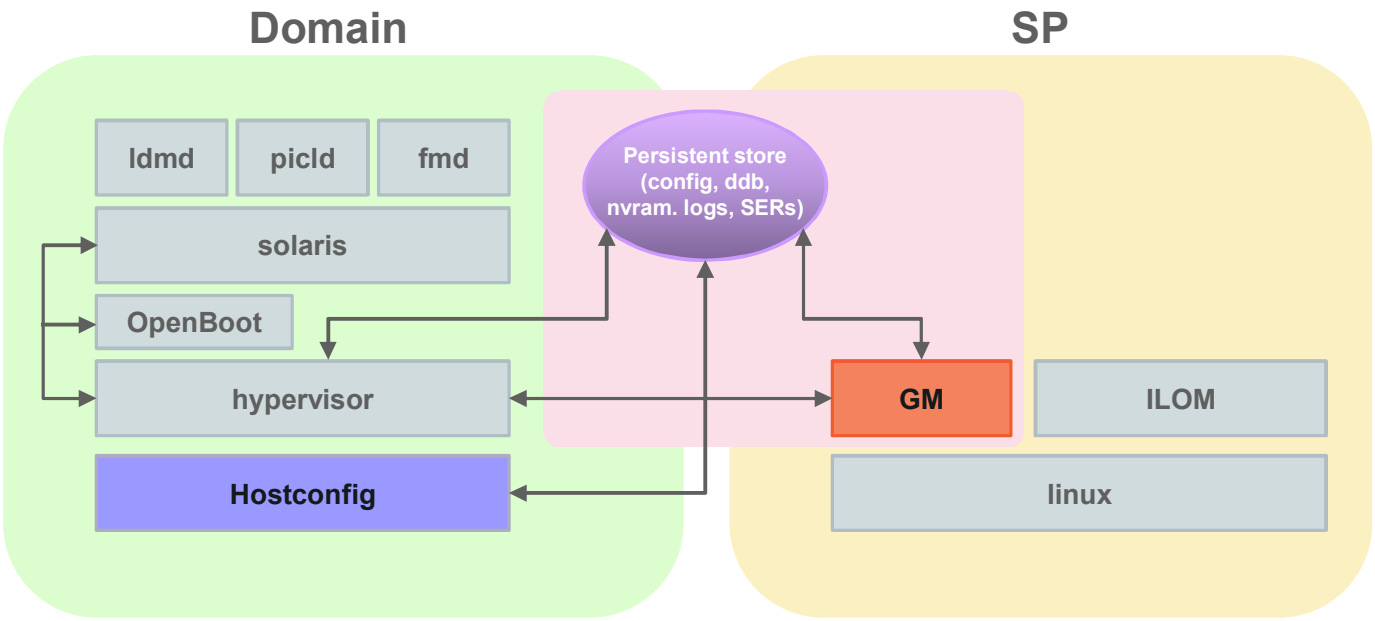
The Hypervisor MD (HVMD) defines a set of logical domains and its resources. It is generated by Hostconfig for initial factory-default configuration only. The LDoms Manager produces new HVMDs and dynamically updates them as the system runs. The LDoms Manager–generated HVMD and guest MDs can be saved on the SP as selectable bootsets.

OBP is resident in the stack only for booting. When Solaris has booted, OBP is no longer part of the picture.

Bootsets

- Bootsets are local copies of the SP config on the control domain **`/var/opt/SUNWldrn/bootsets/<config>.`**
- They contain all MDs and the LDoms constraints database.
- **`ldm add-conf ig <conf ig>`** creates the bootset (and the config on the SP proxy).
- The bootset on the control domain is considered the “master copy” of a config. The copy on the host flash of the SP proxy is a replica.
- The LDoms Manager synchronizes the bootsets with the SP proxy during startup or at the request of the SP proxy.
- Bootsets and SP configs are time-stamped.
- On the SP side, their information is stored in `/Servers/PDomains/PDomain_0/HOST/domain/configs`.

Integrated Software Stack: Simplified View



The Guest Manager (GM) resides on the SP. It provides services to guest domains via Logical Domain Channels (LDCs). It is the communication bridge between the host and ILOM, and is responsible for managing LDom configurations.

OBP is present in the stack only for booting. When Solaris has booted, OBP is no longer part ofthe picture.

The service processor (SP) monitors and runs the physical machine, but it does not manage the logical domains. The LDoms Manager manages the logical domains.

SER in the slide refers to Serious Error Reports.

Hypervisor Runtime Execution

- Virtualization of machine hardware
  - UART Console (for control domain only)
  - Time-of-day (TOD) chip
  - Trusted Platform Module (TPM) and Trusted Boot interfaces
  - Random Number Generator (RNG)
  - Performance Counters
  - Instruction emulation (when necessary)
  - NVRAM functionality through guest MD
- Interrupts
  - Managing their configuration and targeting them to the right CPUs
  - Common framework used for all virtual and physical interrupt sources

Logical Domain Channels (LDC)

- Provide low-level data services between components
- Run point-to-point
  - LDoms Manager to guest
  - LDoms Manager to HV
  - LDoms Manager to SP
  - Control domain Solaris to SP
- Allow passage of large chunks of data more efficiently than mailboxes

Enabling the Logical Domain Manager Daemon

- To enable the Logical Domain Manager daemon if needed:
  `# svcadm enable ldmd`
- To verify that the Logical Domain Manager is running:
  `# /opt/SUNWldm/bin/ldm list`

```
NAME     STATE  FLAGS CONS VCPU MEMORY UTIL UPTIME
primary active ---c-  SP    64   3264M  0.3%  19d 9m
```

Creating Default Services

- To create a virtual console concentrator (vcc) service for use by the virtual network terminal server daemon (vntsd) and as a concentrator for all logical domain consoles:
  `# ldm add-vcc port-range=5000-5100 primary-vcc0 primary`
- To create a virtual disk server (vds) to allow importing virtual disks into a logical domain:
  `# ldm add-vds primary-vds0 primary`
- To create a virtual switch service (vsw) to enable networking between virtual network (vnet) devices in logical domains:
  `# ldm add-vsw net-dev=net0 primary-vsw0 primary`
- To verify that the services have been created:
  `# ldm list-services primary`

The following virtual device services must be created to use the control domain as a service domain and to create virtual devices for other domains:

- vcc: Virtual console concentrator service
- vds: Virtual disk server
- vsw: Virtual switch service

In the first example in the slide, the command would add a virtual console concentrator service (primary-vcc0) with a port range from 5000 through 5100 to the control domain (primary).

In the second example, the command adds a virtual disk server (primary-vds0) to the control domain (primary).

In the third example, the command adds a virtual switch service (primary-vsw0) on the network adapter driver, net0, to the control domain (primary).

Initial Configuration of the Control Domain

- To assign virtual CPUs to the control domain:

  `# ldm set-vcpu 72 primary` OR
  `# ldm set-core 9 primary`
- To initiate a delayed reconfiguration of the control domain:
  `# ldm start-reconf primary`
- To assign memory to the control domain:
  `# ldm set-memory 4G primary`
- To add a logical domain machine configuration to the service processor:
  `# ldm add-config initial`
- To verify that the configuration is ready to be used at the next reboot:
  `# ldm list-config`

Initially, all system resources are allocated to the control domain. To allow the creation of other logical domains, you must release some of these resources.

You must reboot the control domain for the configuration changes to take effect and for the resources to be released for other logical domains. Either a reboot or a power cycle instantiates the new configuration. Only a power cycle actually boots the configuration saved to the service processor (SP), which is then reflected in the `list-config` output.

Enabling the Virtual Network Terminal Server Daemon

- To enable the virtual network terminal server daemon:

```
# svcadm enable vntsd
```

- To verify that the vntsd daemon is enabled:

```
# SVCS vntsd
STATE  STIME  FMRI
online Jan 08 svc:/ldoms/vntsd:default
```


You must enable the virtual network terminal server daemon (vntsd) to provide access to the virtual console of each logical domain.

Setting Up Guest Domains

- To create a logical domain:
  `# ldm add-domain domain-name`
- To add CPUs to the guest domain:
  `# ldm add-vcpu 8 domain-name`
- To add memory to the guest domain:
  `# ldm add-memory 2G domain-name`
- To add a virtual network device to the guest domain:
  `# ldm add-vnet vnet1 primary-vsw0 domain-name`

Where:

- `vnet1` is a unique interface name to the logical domain.
- `primary-vsw0` is the name of an existing network service (virtual switch) to which to connect.

The guest domain must run an operating system that understands both the sun4v platform and the virtual devices presented by the Hypervisor. Currently, this means that you must run at least the Oracle Solaris 10 11/06 OS. Running the Oracle Solaris 10 8/11 OS provides you with all the Oracle VM Server for SPARC 3.0 features.

When you have created the default services and reallocated resources from the control domain, you can create and start a guest domain.

Setting Up Guest Domains

- To specify the device (physical disk) to be exported by the virtual disk server as a virtual disk to the guest domain:

```
# ldm add-vdsdev /dev/dsk/c2tld0s2 voll@prirnary-vdsO
```

Where:

- `/dev/dsk/c2tld0s2` is the path name of the actual physical device.
- `vol 1` is a unique name that you must specify for the device being added to the virtual disk server.
- `primary-vds0` is the name of the virtual disk server to which to add this device.

Setting Up Guest Domains

- To add a virtual disk to the guest domain:

```
# ldm add-vdisk vdisk1 vol1@primary-vds0 domain-name
```

Where:

- `vdisk 1` is the name of the virtual disk.
- `vol 1` is the name of the existing volume to which to connect.
- `primary-vds0` is the name of an existing virtual disk server to which to connect.

The virtual disks are generic block devices that are associated with different types of physical devices, volumes, or files. A virtual disk is not synonymous with an SCSI disk and therefore, excludes the target ID in the disk label. The virtual disks in a logical domain have the following format: c*N*d*N*s*N*, where c*N* is the virtual controller, d*N* is the virtual disk number, and s*N* is the slice.

Setting Up Guest Domains

- To set the `auto-boot?` and `boot-device` variables for the guest domain:

```
# ldm set-var auto-boot\?=true domain-name
# ldm set-var boot-device=vdisk1 domain-name
```

- To bind the resources to the guest domain, and then list the domain to verify that it is bound:

```
# ldm bind-domain domain-name
# ldm list-domain domain-name
 NAME STATE FLAGS CONS VCPU MEMORY UTIL UPTIME
 ldg1 bound ----- 5000    8    2G
```

- To find the console port of the guest domain, look at the output from `list-domain` under the `CONS` heading. In the preceding example, this logical domain has its console output bound to port 5000.

Setting Up Guest Domains

- To connect to the console of the guest domain from the control domain:
  `# telnet localhost port-number`
- To start the guest domain from the control domain:
  `# ldm start-domain domain-name`

Removing Guest Domains

- To stop a logical domain:
  `# ldm stop-domain domain-name`
- To unbind resources from the guest domain:
  `# ldm unbind domain-name`
- To remove the guest domain:
  `# ldm rm-domain domain-name`
- To verify that the guest domain is gone:
  `# ldm list-domain`
- To see the LDom SP configuration:
  `# ldm list-spconfig`
- To remove the LDom SP configuration if needed:
  `# ldm rm-spconfig config-name`

Before removing a guest domain, you must first stop it and unbind the resources from it. After removing the guest domain, you need to clean up the configuration on the SP.

I/O Domains

You may want to configure an I/O domain for the following reasons:

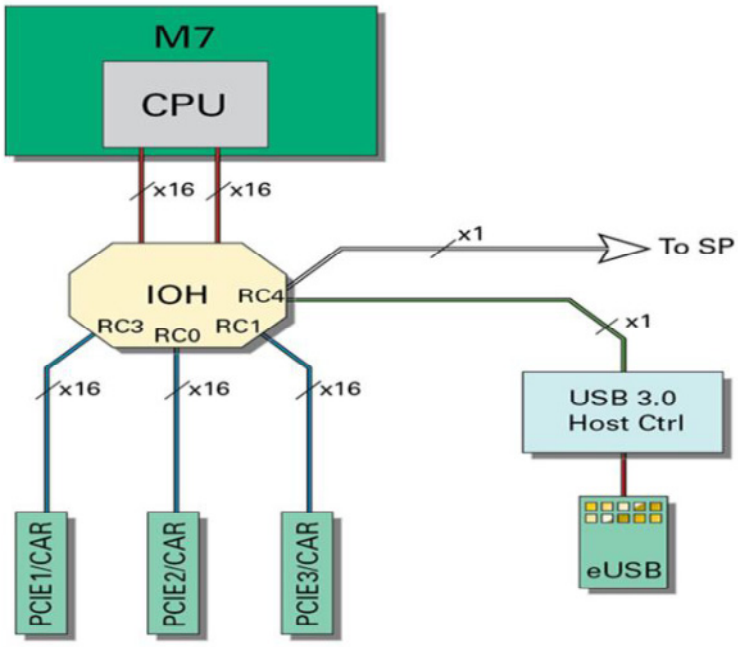
- An I/O domain has direct access to a physical I/O device, which avoids the performance overhead that is associated with virtual I/O. As a result, the I/O performance on an I/O domain more closely matches the I/O performance on a bare-metal system.
- An I/O domain can host virtual I/O services to be used by other guest domains.

An I/O domain has direct ownership of and direct access to physical I/O devices. It can be created by assigning a PCI EXPRESS (PCIe) bus or a PCIe endpoint device to a domain. Use the `ldm add-io` command to assign a bus or device to a domain.

An I/O domain might have direct access to one or more I/O devices, such as PCIe buses, network interface units (NIUs), PCIe endpoint devices, and PCIe single root I/O virtualization (SR-IOV) virtual functions.

This type of direct access to I/O devices means that more I/O bandwidth is available to provide services to the applications in the I/O domain as well as virtual I/O services to guest domains.

Assigning PCIe Buses



You can use the Oracle VM Server for SPARC software to assign an entire PCIe bus (also known as a *root complex*) to a domain. An entire PCIe bus consists of the PCIe bus itself, and all of its PCI switches and devices. The PCIe buses that are present on a server are identified with names such as pci@300. An I/O domain that is configured with an entire PCIe bus is also known as a *root domain*.

The diagram in the slide shows a system that has two PCIe buses (pci_0 and pci_1). Each bus is assigned to a different domain. Thus, the system is configured with two I/O domains.

When you assign a PCIe bus to an I/O domain, all the devices on that bus are owned by that I/O domain. You are not permitted to assign any of the PCIe endpoint devices on that bus to other domains. Only the PCIe endpoint devices on PCIe buses that are assigned to the primary domain can be assigned to other domains.

When a server is initially configured in a logical domains environment or is using the factory-default configuration, the primary domain has access to all the physical device resources. This means that the primary domain is the only I/O domain configured on the system and that it owns all the PCIe buses.

Setting Up I/O Domains with PCIe Buses

- To verify that the primary domain owns more than one PCIe bus:
  `primary# ldm list-io`
- To determine the device path of the boot disk that must be retained:
  `primary# df /`
- To determine the network interface that is used by the system:
  `primary# ifconfig -a`
  `primary# dladm show-phys net0 (to get device)`
  `primary# ls -1 /dev/<device>`

The next few slides will provide an example of how to create a new I/O domain from an initial configuration where several buses are owned by the primary domain. By default, the primary domain owns all the buses present on the system.

First, you must retain the bus that has the primary domain's boot disk. Then remove another bus from the primary domain and assign it to another domain.

Setting Up I/O Domains with PCIe Buses

- To remove a bus that does not contain the boot disk or the network interface from the `primary` **domain:**

```
primary# ldm start-reconf primary
primary# ldm remove-io pci_1 primary
```

- To save this configuration to the service processor:

```
primary# ldm add-config io-domain
```

- To reboot the root domain so that the changes take effect:

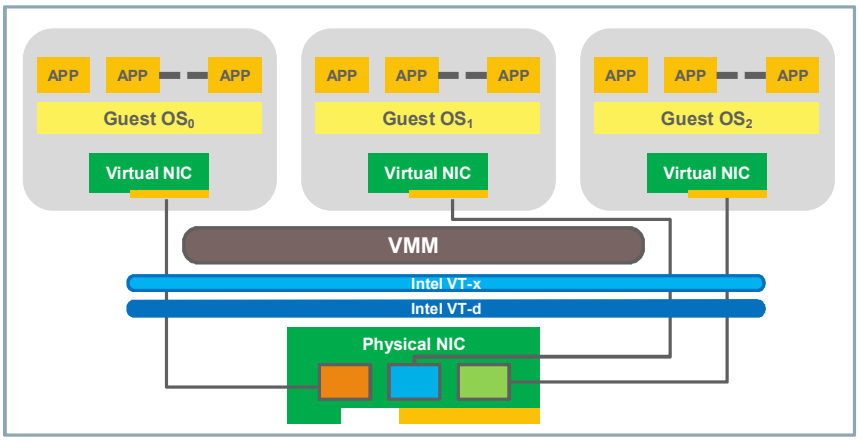
```
primary# shutdown -i6 -g0 -y
```

Setting UP I/O Domains with PCIe Buses

- To stop the domain to which you want to add the PCIe bus:
  `primary# ldm stop <id-name>`
- To add the available bus to the domain that needs direct access:
  `primary# ldm add-io pci_1 <ld-name>`
- To restart the domain so that the changes take effect:
  `primary# ldm start <ld-name>`
- To confirm that the correct bus is still assigned to the primary domain and that the correct bus is assigned to `ld-domain`:
  `primary# ldm list-io`

PCI-SIG Single Root IOV

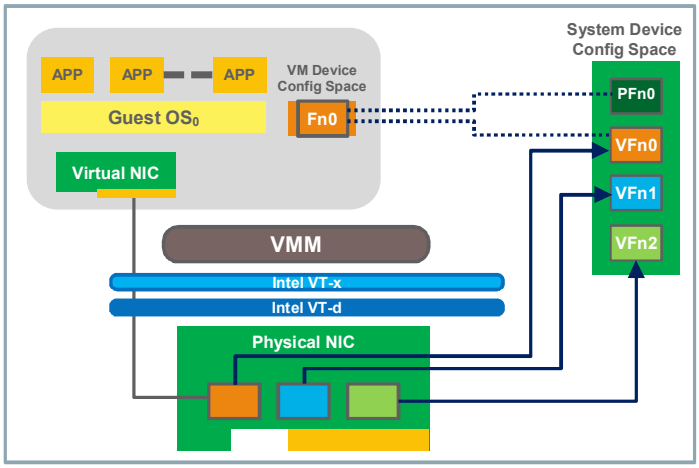
- Standardized way of bypassing the VMM's involvement in data movement by providing independent memory space, interrupts, and OMA Streams for each VM
- Benefits
  - Native I/O Performance
  - Scalability
- Drawbacks: Hard to live migrate VM, somewhat similar to direct assignment



Support for the Peripheral Component Interconnect Express (PCIe) single root I/O virtualization (SR-IOV) feature has been added starting with the Oracle VM Server for SPARC 2.2 release.

Single Root I/O Virtualization (SR-IOV)

- IOV for PCI Express (PCIe) HW
  - An IOV solution that allows direct access to PCI Express devices at Virtual Function (VF) granularity from a Guest Domain
  - Standard for the PCIe fabric with a Single Root-Complex (SR-IOV)
  - Standard for the PCIe fabric with multiple Root-Complexes (MR-IOV)
- Features: Direct access to VF registers, interrupts, and OMA
- Usage model
  - The individual NIC port belongs to different operating systems.
  - Multiple Guests share SR-IOV devices.



The SR-IOV standard enables efficient sharing of PCIe devices among virtual machines, and is implemented in the hardware to achieve I/O performance that is comparable to native performance. The SR-IOV specification defines a new standard wherein new devices that are created enable the virtual machine to be directly connected to the I/O device.

A single I/O resource, which is known as a *physical function*, can be shared by many virtual machines. The shared devices provide dedicated resources and also use shared common resources. Thus, each virtual machine has access to unique resources. Therefore, a PCIe device that is SR-IOV-enabled with the appropriate hardware and OS support, such as an Ethernet port, can appear as multiple, separate physical devices, each with its own PCIe configuration space.

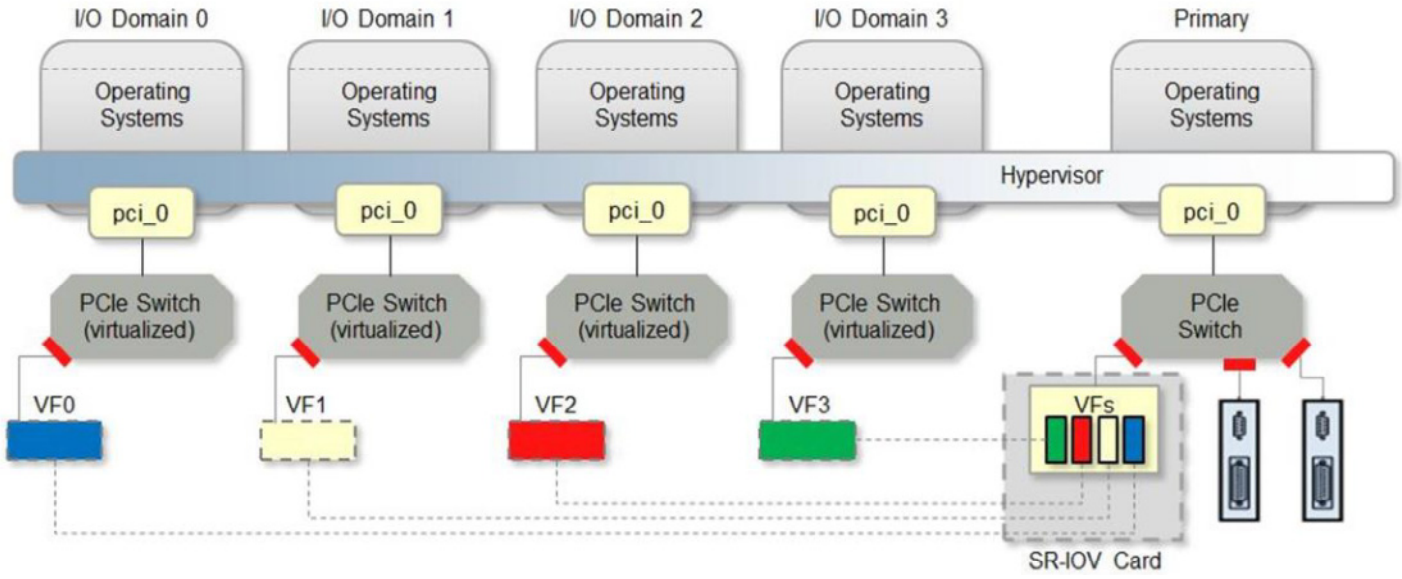
I/O Virtualization (IOV) Benefits

- Performance
  - Full utilization of I/O device resources such as 10*g* NIC bandwidth
  - Low latency
- Cost reduction: Capital and operational expenditure savings from power, reduced adapter count, less cabling, and fewer switch ports
- Disadvantage
  - Migration is disabled after the VFs are assigned to a domain.

Each SR-IOV device can have a physical function and each physical function can have up to 64,000 virtual functions associated with it. This number is dependent on the particular SR-IOV device. The virtual functions are created by the physical function.

After SR-IOV is enabled in the physical function, the PCI configuration space of each virtual function can be accessed by the bus, device, and function number of the physical function. Each virtual function has a PCI memory space, which is used to map its register set. The virtual function device drivers operate on the register set to enable its functionality and the virtual function appears as an actual PCI device. After creation, you can directly assign a virtual function to an I/O domain. This capability enables the virtual function to share the physical device and to perform I/O without CPU and Hypervisor software overhead.

A High-Level View of Virtual Functions



The figure in the slide shows the relationship between virtual functions (VFs) and a physical function in an I/O domain.

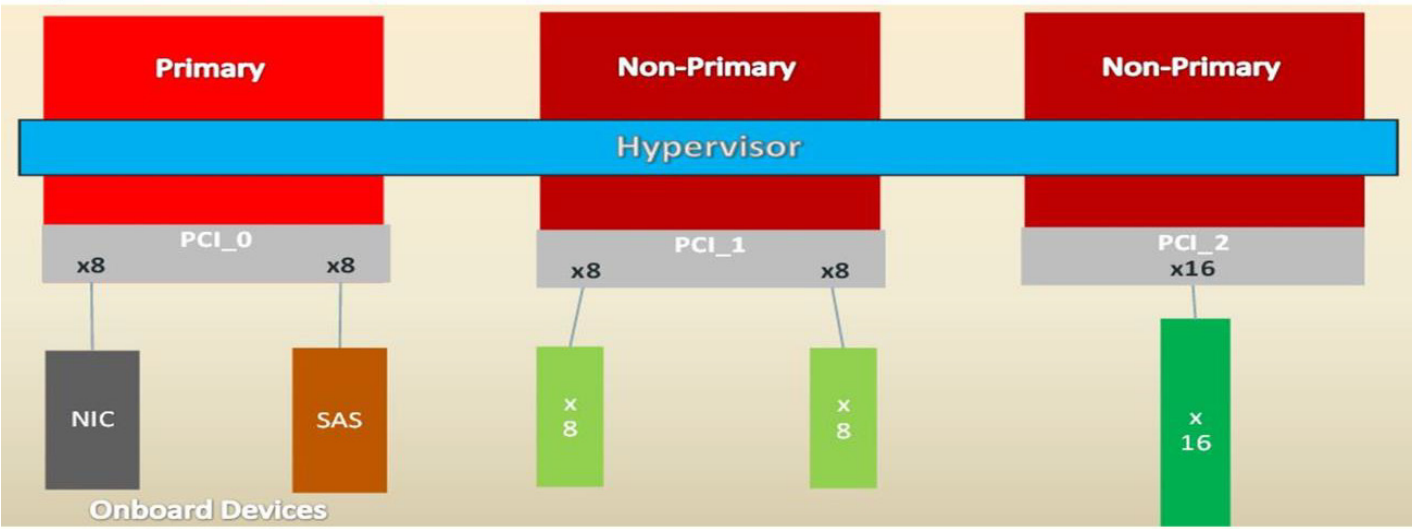
SR-IOV has the following function types:

- **Physical function:** A PCI function that supports SR-IOV capabilities as defined by the SR-IOV specification. A physical function contains the SR-IOV capability structure, and manages the SR-IOV functionality. Physical functions are fully featured PCIe functions that can be discovered, managed, and manipulated like any other PCIe device. Physical functions can be used to configure and control a PCIe device.
- **Virtual function:** A PCI function that is associated with a physical function. A virtual function is a lightweight PCIe function that shares one or more physical resources with the physical function and with virtual functions that are associated with that physical function. Unlike a physical function, a virtual function can configure only its own behavior.

Dynamic Bus (DBUS)

- A Dynamic Bus is where you can add an entire PCIe Bus (also known as a PCIe Root-Complex) along with all PCIe slots behind it to and remove the entire bus from running LDOMs Hypervisor domains without any reboot.
  - Use `ldm add-io / remove-io <bus> <domain>`.
  - Static assignment of PCIe Bus is also called “Split PCI.”
- A I/O domain that is configured with an entire PCIe bus is also known as a Non-Primary Root Domain (NPRD).
  - By default, all the PCIe root complexes are owned by the primary or control domain.

Dynamic Bus: NPRD



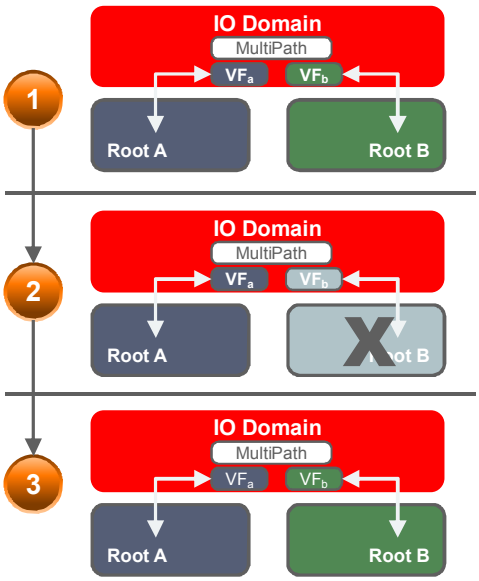
Use Cases

- Creating NPRD domains by assigning PCIe buses that can be set up as alternative service domains
- Live service of CMIOU

I/O Domain Resiliency (IOR)

- Requirements
  - An I/O Domain shall continue to run when the associated Root domain reboots or panics.
  - The impacted devices shall be reinstated and restored when the Root domain recovers.
    - Device driver support required to suspend or restore I/O on the affected devices
  - Possible in LDOMs environment only (on sun4v platforms)
  - Multipath configurations are required in the I/O domain for uninterrupted I/O.
- The I/O Domain is only as resilient as the weakest link in the configuration.
  - Multipathed configurations across Root domains are the best.
  - A vnet device can be an alternative path for an SR-IOV NIC device.
- Current support is limited to SR-IOV devices.

IOR Use Case 1: Multipathed SR-IOV, IOR-Compliant VF Drivers

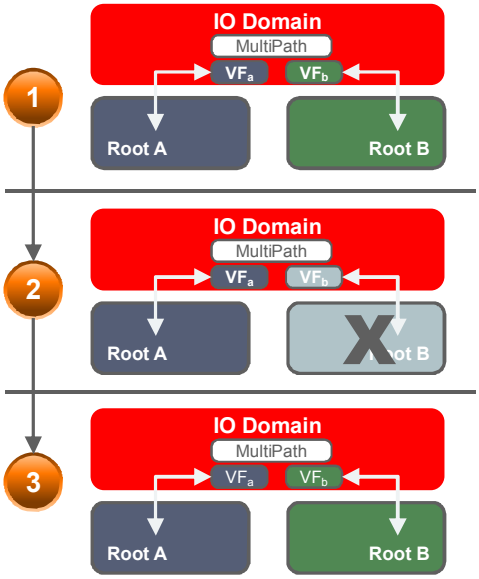


- Two Root domains are configured.
- Each Root provides a VF to the I/O Domain.
- The I/O Domain multi-paths the VFs.
- Applications access the multi-path group for I/O.

- Root B is interrupted (panic or reboot).
- VF-B LSR is suspended in the I/O Domain.
- The multi-pathing software is engaged; all I/O is routed through Root A.

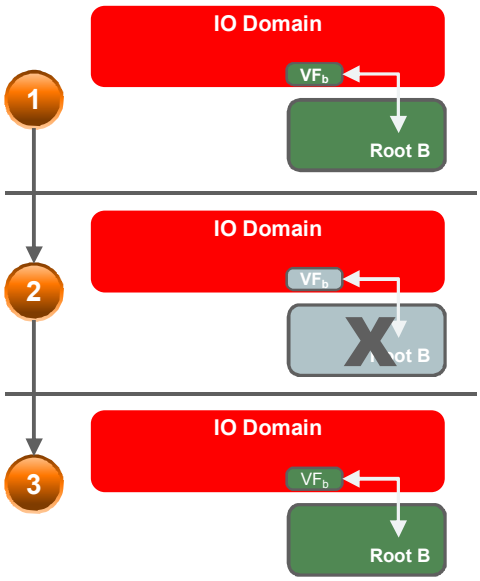
- Root B is restored to service.
- VF-B LSR resumes in the I/O Domain. It is reinstated and returned to service.
- The multi-path group is restored to full RAS.

IOR Use Case 2: Multipathed SR-IOV, Non-IOR-Compliant VF Drivers



- Two Root domains are configured.
- Each Root provides a VF to the I/O Domain.
- The I/O Domain multi-paths the VFs.
- Applications access the multi-path group for I/O.
- Root B is interrupted (panic or reboot).
- VF-B is “fenced” in I/O Domain; all I/O attempts to VF·B fail.
- Multi-pathing is eventually engaged. All I/O is routed through Root A.
- Root B is restored to service. VF-B remains in “fenced” state.
- For manual recovery (requires “force detach,” which is not implemented)
  - Force detach VF-B.
  - Use DIOV to remove VF-B, then use DIOV to add VF-B to the I/O domain.
  - Restore the multi-path group.

IOR Use Case 3: Single SR-IOV, IOR-Compliant VF Driver

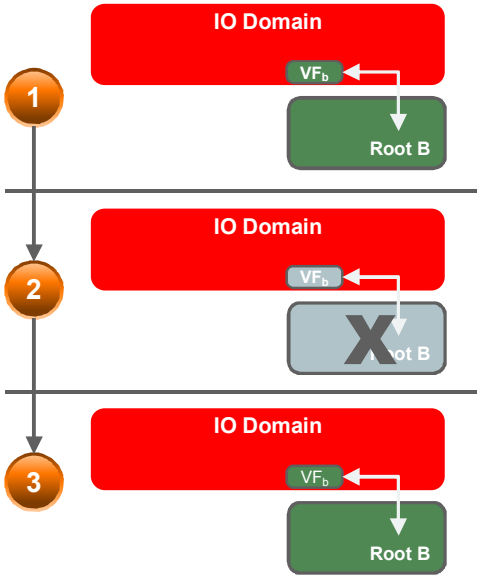


- A Single Root Domain is configured.
- Root provides a VF to the I/O Domain.
- Applications access the VF directly for I/O.

- Root B is interrupted (panic or reboot).
- VF-B LSR is suspended on I/O Domain.
- Applications in the I/O Domain are impacted.

- Root B is restored to service.
- VF-B LSR resumes in the I/O Domain. It is reinstated and returned to service.
- Applications can re-establish connections with VF.
  - May require manual Intervention at the application layer

IOR Use Case 4: Single SRIOV, Non-IOR-Compliant VF Driver

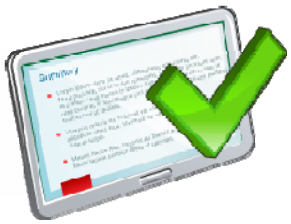


- Single Root Domain is configured.
- Root provides a VF to the I/O Domain.
- Applications access the VF directly for I/O.
- Root B is interrupted (panic or reboot).
- VF-B is “fenced” in the I/O Domain. All I/O attempts to VF-B fail.
- Applications In the I/O Domain are impacted.
- Root B is restored to service. VF-B remains in “silenced” state.
- Manual recovery (which may not work)
  - Use DIOV to remove VF-B, then use DIOV to add VF-B to the I/O domain.
  - Applications can re-establish connections with VF.

Summary

In this lesson, you should have learned how to:

- Define logical domains
- Describe logical domain planning considerations
- Create LDoms
- Add resources to LDoms
- Dynamically change LDoms
- Delete LDoms
- Monitor LDoms



5

Data Collection and Fault Analysis

Objectives

After completing this lesson, you should be able to:

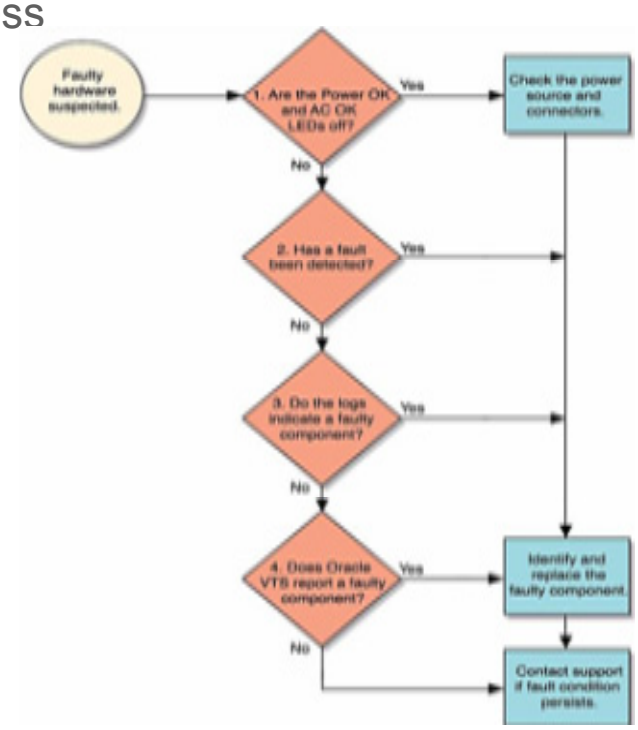
- Discuss the process for diagnosing faults
- Use diagnostic tools to help when troubleshooting
- Check for faults
- View and interpret log files and system messages
- Manage the event and audit logs
- Configure POST
- Interpret system LEDs
- Monitor the server
- Display server information
- Use Oracle Explorer and snapshot scripts to gather domain information

Additional Resources

The following references provide additional information about the topics described in this lesson:

- M7-8 and M7-16 Technology Portal on OTN:
  https://www.oracle.com/servers/sparc/m7-8/index.html
  https://www.oracle.com/servers/sparc/m7-16/index.html
- M7-8 and M7-16 white papers:
  https://www.oracle.com/servers/sparc/resources.html

Diagnostics Process



Diagnostic Actions

Diagnostic Action	Possible Outcome
Check Power OK and AC Present LEDs on the server.	If these LEDs are not on, check the power source and power connections to the server.
Check the log files for fault information.	If system messages indicate a faulty component, replace it.
Enter the `# fmadm faulty` command from Solaris or the `-> show faulty` command from ILOM to check for faults.	If the fault is detected by PSH, refer to the PSH Knowledge Article website for additional information.
Run the Oracle VTS software. The Oracle VTS version required is: 511.3 VTS 7.0ps 19.2 (511.3).	To run Oracle VTS, the server must be running Solaris. If Oracle VTS reports a faulty component, replace it. If it does not report a faulty component, run POST.

Note: By setting an IP address for `syslog` on the Oracle ILOM's Notifications > Syslog BUI page, you can enable remote logging.

The table in the slide provides descriptions of the troubleshooting actions that you should take to identify a faulty component. The diagnostic tools you use, and the order in which you use them, depend on the nature of the problem you are troubleshooting.

Log files can be found both in the domain and on the SP. Domain log information can be obtained with the `dmesg` command to see what is in the system buffer and by viewing the contents of the `/var/adm/messages` file. SP log information can be obtained by viewing the event log, `-> show/SP/logs/event/list`, and the audit log, `-> show /SP/logs/audit/list`.

Diagnostic Tools: Overview

Diagnostic tools include:

- LEDs
- Oracle ILOM
- Power-on self-test (POST)
- Oracle Solaris OS Predictive Self-Healing
- Fault Management Architecture (FMA) on ILOM
- Log files and command-line interface
- Sun VTS

Note: The LEDs, Oracle ILOM, PSH, and many of the log files and console messages are integrated. For example, when the Oracle Solaris software detects a fault, it displays the fault, logs it, and passes the information to Oracle ILOM, where it is logged.

You can use a variety of diagnostic tools, commands, and indicators to monitor and troubleshoot a server. LEDs provide a quick visual notification of the status of the server and some of the replaceable components.

Oracle ILOM firmware runs on the SPs. In addition to providing the interface between the hardware and OS, ILOM also tracks and reports the health of key server components. It also works closely with POST, Solaris Predictive Self-Healing technology, and the Fault Management Architecture on Oracle ILOM to keep the server running even if there is a faulty component. POST performs diagnostics on server components upon server reset to ensure the integrity of those components. POST is configurable and works with Oracle ILOM to take faulty components offline if needed.

Solaris's Predictive Self-Healing (PSH) technology continuously monitors the health of the CPU, memory, and other components, and works with FMA on Oracle ILOM to take a faulty component offline if needed. The PSH technology enables servers to accurately predict component failures and mitigate many serious problems before they occur. The system provides the standard Oracle Solaris OS log files and investigative commands that can be accessed and displayed on the device of your choice. Sun VTS is an application that exercises the server, provides hardware validation, and discloses possible faulty components with recommendations for repair.

Tool Availability

Tool	Oracle ILOM Prompt	Open Boot Prompt	Solaris Prompt
Status LEDs	Yes	Yes	Yes
PSH commands	Yes	No	Yes
Oracle ILOM logs and commands	Yes	No	No
OpenBoot commands	No	Yes	No
Solaris commands and logs	No	No	Yes
Oracle VTS	No	No	Yes (if installed)
Third-party software	No	No	Yes (if installed)

The table in the slide describes what tools are available at the different states in which the server operates.

ILOM Service–Related Tools

Oracle ILOM Command	Description
`help [command)`	Displays a list of all available commands with syntax and descriptions. Specifying a command name as an option displays help for that command.
`set /HOSTx send_break_action=break`	Takes the host server from the OS to the Open Boot prompt. It is not a graceful shutdown. See `help /HOSTO` for more details.
`set /HOSTx send_break_action=dumpcore`	Causes the host server to panic and save a core dump. It is not graceful.
`start /HOSTx/console`	Connects to the host. See `help /HOSTO/console` for an example.
`show /HOSTx/console/history`	Displays the contents of the host's console buffer
`set /HOSTx/bootmode property=value`	Controls the method of booting for the server's firmware. The value of property can be state, config, or script. See `/HOSTO/bootmode` for an example.

The Oracle ILOM shell commands listed in the slide are used most frequently when performing service- related tasks.

Remember, `/HOSTx = /Servers/PDomains/PDomain_x/HOST`.

ILOM Service–Related Tools

Oracle ILOM Command	Description
`stop /System or stop /SYS`	Powers off all of the HOSTs
`start /System or start /SYS`	Powers on all of the HOSTs
`start /HOSTx`	Powers on the specified HOST, x
`stop /HOSTx`	Powers off the specified HOST, x
`reset /System or reset /SYS`	Generates a hardware reset on all of the HOSTs
`reset /SP`	Reboots both service processors and all four service processor proxies (SPPs)

Remember, `/HOSTx = /Servers/PDomains/PDomain_x/HOST`.

Predictive Self-Healing: Overview

A Predictive Self-Healing (PSH) console message provides the following information about each detected fault:

- Type
- Severity
- Description
- Automated response
- Impact
- Suggested action for system administrator

PSH enables the server to diagnose and mitigate problems before they negatively affect operations. PSH uses the Fault Manager daemon, `fmd`, which starts at boot time and runs in the background, to monitor all of the faults that are generated by the components in the server. On the SP, PSH works with Oracle ILOM to manage all of the components on the server. On the host, PSH works with POST and the Oracle Solaris OS to manage the components assigned to the host.

If a component generates a fault, the `fmd` daemon correlates the fault with data from previous faults and other relevant information to diagnose the problem. After diagnosis, the daemon assigns a UUID to the error. This value distinguishes this error across any set of systems.

When possible, the Fault Manager daemon initiates steps to self-heal the failed component and take the component offline. The daemon also logs the fault to the `syslogd` daemon and provides a fault notification with an MSGID. You can use the message ID to get additional information about the problem from the knowledge article database. If PSH detects a faulty component, use the `fmadm faulty` command to display information about the fault.

Checking for Faults (1 of 4)

```
-> start /SP/faultmgmt/shell
Are you sure you want to start /SP/faultmgmt/shell (y/n)? Y

faultmgmtsp> fmadm faulty
--------------------------------------------------------------------------------
Time                 UUID                                   msgid
--------------------------------------------------------------------------------
2015-01-28/06:10:56  04d5aafe-1797-6765-c4d4-b55c90c48787  SPT-8000-95
Problem Status     :  OPEN
Diag Engine        :  FDD 1.0
System
   Manufacturer    : Oracle Corporation
   Name            : SPARC M7-16
   Part_Number     : 32397701+2+1
   Serial_Number   : AK00192121
```


The `fmadm faulty` command displays the list of faults detected by PSH. You can run this command from either the host or through the `fault management` shell under ILOM.

In this example, a fault is displayed, indicating these details:

- Date and time of the fault (`2013-03-12/21:45:36`)
- UUID (`dd779bdd-a73a-c46c-f4c1-cc3dcd071cad`), which is unique for every fault
- Message identifier (`SPSUN4V-8001-6Q`), which can be used as a link to obtain additional fault information
- Faulted field replaceable unit (FRU). The information provided in the example includes the part number of the FRU (Part Number: `7045605`) and the serial number of the FRU (Serial Number: `AK00066329`).

Checking for Faults (2 of 4)

```
---------------------------------------------------------------
Suspect 1 of 1
   Fault class  : fault.chassis.power.overtemperature
   Certainty    : 100%
   Affects      : /SYS/PS15/SUPPLY
   Status       : faulted
   FRU
      Status            : faulty
      Location          : /SYS/PS15
      Manufacture       : Power-One, Inc.
      Name              : A261 POWER SUPPLY
      Part Number       : 7068817
      Revision          : 21
      Serial_Number     : 465776G+1348B20C8D
      Chassis
         Manufacturer   : Oracle Corporation
         Name           : SPARC M7-16
         Part_Number    : 32397701+2+1
         Serial_Number  : AK00192121
```


The FRU field provides the name of the FRU (`/SYS/PS15` for PS15 in this example).

Checking for Faults (3 of 4)

```
Description  : A power supply over-temperature failure has occurred.
Response     : The service-required LED on the affected power supply and
               chassis will be illuminated.
Impact       : Server will be powered down when there are insufficient
               operational power supplies.
Action       : Please refer to the associated reference document at
               http://support.oracle.com/msg/SPT-8000-95 for the latest service
               procedures and policies regarding this diagnosis.
```


The end of the output reviews the description, response, impact, and recommended action for this fault.

Checking for Faults (4 of 4)

```
-> show faulty
Target                    I Property                        I Value
------------------------------------------------------------------------------
/SP/faultmgmt/0  fru      | fru                             | /SYS/PS1
/SP/faultmgmt/0/faults/0  | class                           | fault.chassis.env.power.loss
/SP/faultmgmt/0/faults/0  | sunw-msg-id                     | SPT-8000-5X
/SP/faultmgmt/0/faults/0  | component                       | /SYS/PS1/SUPPLY
/SP/faultmgmt/0/faults/0  | uuid                            | 14e27b8e-5755-62ea-9bbf- bfea22fe0b7f
/SP/faultmgmt/0/faults/0  | timestamp                       | 2015-01-26/22:24:24
/SP/faultmgmt/0/faults/0  | fru_part_number                 | 7068817
/SP/faultmgmt/0/faults/0  | fru_rev_level                   | 21
/SP/faultmgmt/0/faults/0  | fru_serial_number               | 465776G+l348B20CS8
/SP/faultmgmt/0/faults/0  | fru_manufacturer                | Power-One, Inc.
/SP/faultmgmt/0/faults/0  | fru_name                        | A261_POWER_SUPPLY
/SP/faultmgmt/0/faults/0  | system_component_manufacturer   | Oracle Corporation
/SP/faultmgmt/0/faults/0  | system_component_name           | SPARC M7-16
/SP/faultmgmt/0/faults/0  | system_component_part_number    | 32397701+2+1
/SP/faultmgmt/0/faults/0  | system_component_serial_number  | AK00192121
/SP/faultmgmt/0/faults/0  | chassis_manufacturer            | Oracle Corporation
/SP/faultmgmt/0/faults/0  | chassis_name                    | SPARC M7-16
/SP/faultmgmt/0/faults/0  | chassis_part_number             | 32397701+2+1
/SP/faultmgmt/0/faults/0  | chassis_serial_number           | AK00192121
/SP/faultmgmt/0/faults/0  | system _manufacturer            | Oracle Corporation
/SP/faultmgmt/0/faults/0  | system_name                     | SPARC M7-16
/SP/faultmgmt/0/faults/0  | system_part_number              | 32397701+2+1
/SP/faultmgmt/0/faults/0  | system_serial_number            | AK00192121
```


To discover faults using the web interface, perform the following steps: (If problems are found, refer to the service manual for your server for information about detecting faults, for corrective actions, and for information about manually clearing faults.)

1. Log in to the Oracle ILOM web interface.
2. Check the server status on the System Information > Summary page.
3. Check for any open problems by clicking System Information > Open Problems. If open problems exist, details describing the problems appear in the Open Problems table.

   Note: Oracle ILOM automatically clears the messages in the Open Problems table upon detecting the replacement or repair of a server component.
4. When applicable, click the URL link in the message to view further details about the problem and for suggested corrective actions.

Clearing a Fault from Solaris

- To determine whether the replaced FRU still shows a faulty status:
  ```
  # fmadm faulty
  ```


When PSH detects faults, the faults are logged and displayed on the console. In most cases, after the fault is repaired, the corrected state is detected by the server and the fault condition is repaired automatically. However, this repair should be verified. In cases where the fault condition is not automatically cleared, the fault must be cleared manually.

You can use the `fmadm replaced` command to indicate that the suspect FRU has been replaced or removed. If the system automatically discovers that an FRU has been replaced (the serial number has changed), this discovery is treated in the same way as if `fmadm replaced` had been entered on the command line. The `fmadm replaced` command is not allowed if `fmd` can automatically confirm that the FRU has not been replaced (the serial number has not changed). If the system automatically discovers that an FRU has been removed but not replaced, the current behavior is unchanged. The suspect is displayed as not present, but is not considered to be permanently removed until the fault event is 30 days old, at which point it is purged.

Clearing a Fault from Solaris

- If the fault is still reported, clear the fault from all persistent fault records:

```
[(flash}root@mpk25emi80:~]# fmadm
   Usage: fmadm <subcommand>
      where <subcommand> is one of the following:
         faulty [-asv] [-u <uuid>} : display list of faults
         faulty -f [-a]            : display faulty FRUs
         faulty -r [-a}            : display faulty ASRUs
         acquit <FRUN>             : acquit faults on a FRU
         acquit <UUID>             : acquit faults associated with UUID
         acquit <FRU> <UUID>       : acquit faults specified by (FRU, UUID) combination
         replaced <FRU>            : replaced faults on a FRU
         repaired <FRU>            : repaired faults on a FRU
         repair <FRU>              : repair faults on a FRU
         rotate errlog             : rotate error log
         rotate fltlog             : rotate fault log
```


You can use the `fmadm repaired` command when some physical repair has been carried out to resolve the problem, other than replacing an FRU. Examples of such repairs include reseating a card or straightening a bent pin.

Often you use the acquit option when you determine that the resource was not the cause. Acquittal can also happen implicitly when additional error events occur, and the diagnosis gets refined. Replacement takes precedence over repair, and both replacement and repair take precedence over acquittal. Thus, you can acquit a component and then subsequently repair it, but you cannot acquit a component that has already been repaired.

Clearing a Fault from ILOM

- To determine whether the replaced FRU still shows a faulty status:
  # fmadm faulty
- If the fault is still reported, clear the fault from all persistent fault records:

```
faultmgmtsp> fmadm
 Usage: fmadm <subcommand>
   where <subcommand> is one of the following:
     acquit <FRU>          : acquit faults on a FRU
     acquit <UUID>         : acquit faults associated with UUID
     acquit <FRU> <UUID>   : acquit faults specified by (FRU, UUID) combination
     replaced <FRU>        : replaced faults on a FRU
     repaired <FRU>        : repaired faults on a FRU
     repair <FRU>          : repair faults on a FRU
```

Locating the Failed Component

- Use an Oracle ILOM command to display faulty components in a server or a group of servers:
```
-> show /System/Open_Problems
-> show faulty
```
- Prepare the component for removal by illuminating the Ready to Remove LED:
```
-> set /SYS/ILOM_target_name
prepare_to_remove_action=true
```
- Verify that the Ready to Remove LED is lit:
```
-> show /SYS/ILOM_target_name prepare_to_remove_status
prepare_to_remove_status = Ready
```

SPARC M7-8 ILOM NAC Names

Component	ILOM Target Name	X	Y	W	Z
CMIOU	/SYS/CMIOUx	0-7			
CM	/SYS/CMIOUx/CM	0-7			
CMP	/SYS/CMIOUx/CM/CMP	0-7			
DIMM	/SYS/CMIOUx!CM/CMP/BOByw/CHz/DIMM	0-7	0-3	0-1	0-1
SP	/SYS/SPx	0-1			
SPM	/SYS/SPx/SPMy	0-1	0-1		
SPDB	/SYS/SPDB				
PCIE	/SYS/CMIOUx/PCIEy	0-1	13		
FM	/SYS/FMx	0-7			
PS	/SYS/PSx	0-8			
PDECB	/SYS/PDECBx	0-7			

SPARC M7-16 ILOM NAC Names

Component	ILOM Target Name	X	Y	W	Z
CMIOU	/SYS/CMIOUx	0-15			
CM	/SYS/CMIOUx/CM	0-15			
CMP	/SYS/CMIOUx/CM/CMP	0-15			
DIMM	/SYS/CMIOUx!CM/CMP/BOByw/CHz/DIMM	0-15	0-3	0-1	0-1
SP	/SYS/SPx	0-1			
SPP	/SYS/SPPx	0-3			
SPM	/SYS/SPPx/SPMy	0-3	0-1		
SPDB	/SYS/SPDBx	0-2			
PCIE	/SYS/CMIOUx/PCIEy	0-15	0-3		
FM	/SYS/FMx	0-7			
PS	/SYS/PSx	0-15			
SWU	/SYS/SWUx	0-5			
SWU/FM	/SYS/SWUx/FMy	0-5	0-5		

Alternate Commands

Component	Alternate Command
HDD	Use the `cfgadm` command.
I/O Card	Use the `hotplug` command.
PCIe low profile carrier	Use the `hotplug` command.
SSD	Use the `cfgadm` command.

In Solaris, use `cfgadm` to locate and manage I/O cards and hard drives. The `cfgadm` command `un-configures` and `configures` HDDs and SSDs. Use `format` to locate hard drives. The `format` command does not support I/O cards. Use `hotplug` to list I/O card slots and identify card types. The `hotplug` command currently does not support hard drives. The device path is how the Oracle Solaris `hotplug` command identifies a slot location.

Ready to Remove LED

Component	Has Ready to Remove LED	Does not have Ready to Remove LED
CMIOU	x	
SP and SPP	x	
Switch Unit	x	
PCIe Unit		x
Power Supplies		x
PDECB		x
Fan Module		x

Note: The CMIOU has a Ready to Remove LED, but it is not currently a hot swap item. Check Product Notes to see if it becomes a hot swap item in the future.

Interpreting Log Files and System Messages

- To check the system buffer for recent diagnostic messages:
  `# dmesg`
- To view the contents of the Solaris messages file:
  `# more /var/adm/messages`
- To view the ILOM event log files:
  `> show/SP/logs/event/list`
- To view the ILOM audit log:
  `-> show/SP/logs/audit/list`

With the OS running on the server, you have the full complement of Oracle Solaris OS files and commands available for collecting information and for troubleshooting. If PSH does not indicate the source of a fault, check the message buffer and log files for notifications for faults. Drive faults are usually captured by the Oracle Solaris message files.

The error logging daemon, `syslogd`, automatically records various system warnings, errors, and faults in message files. These messages can alert you to system problems, such as a device that is about to fail. The `/var/adm` directory contains several message files. The most recent messages are in the `/var/adm/messages` file.

Displaying Console History

- To manage the console history log:
  `-> set /HOSTx/console/history property=option`

  Where property can be:
  - **`line count`:** This option accepts a value within the range of 1 to 2048 lines. Specify"" for an unlimited number of lines. The default is all lines.
  - **`pause count`:** This option accepts a value of 1 to any valid integer or"" for infinite number of lines. The default is not to pause.
  - **`start from`:** The options are:
    - **`end`:** The last line (most recent) in the buffer. This is the default.
    - **`beginning`:** The first line in the buffer.
- To view the console history log:
  `-> show /HOSTx/console/history`

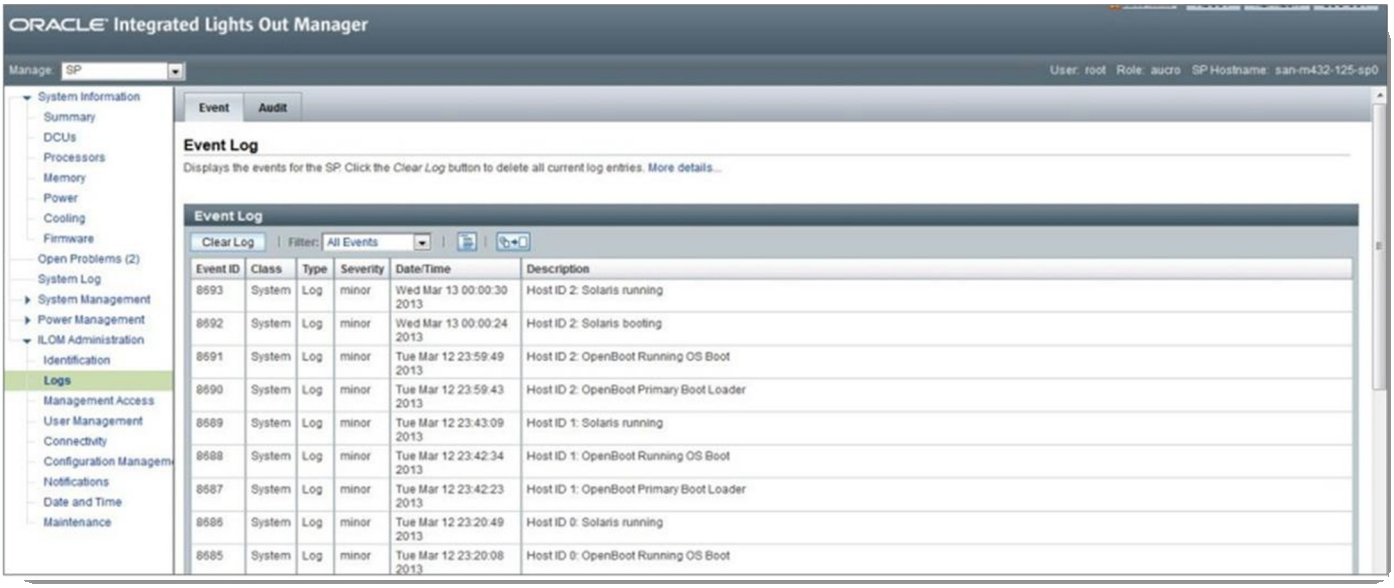
You can use the `/Servers/PDomains/PDomain_x/HOST/console/history` console output buffer to write all types of log information. If you enter `show /HOSTx/console/history` without first setting any arguments with the `set` command, Oracle ILOM will display all lines of the console log, starting from the end.

Note: Timestamps recorded in the console log reflect server time. These timestamps reflect local time, and the Oracle ILOM console log uses Coordinated Universal Time (UTC). The Oracle Solaris OS system time is independent of the Oracle ILOM time.

Event and Audit Logs

- **Event log:** Tracks informational, warning, or error messages about a managed device, such as the addition or removal of a component, or the failure of a component. The properties of the events recorded in the log can include the severity of the event, the event provider (class), and the date and time the event was logged.
- **Audit log:** Tracks all interface-related user actions, such as user logins, logouts, configuration changes, and password changes. The user interfaces monitored for user actions include the Oracle ILOM web interface, the CLI, the fault management shell (captive shell), the restricted shell, and the SNMP and IPMI client interfaces.

Managing the Event and Audit Logs



To manage the event and audit logs:

1. Log in to the Oracle ILOM web interface.
2. View the ILOM Administration > Logs page. The event log is displayed.
3. If needed, filter the event types shown, or control the display properties for rows and pages.
4. Use the controls at the top of the log table.
5. If needed, clear all log entries shown in the table by clicking Clear Log.
6. A confirmation dialog box appears. In the confirmation dialog box, click OK to clear the entries.
7. Click the Audit tab to view the audit log. The audit log is displayed.

POST Variables (1 of 2)

Parameter	Values	Description
/HOSTx keyswitch_state =	normal	The SPARC server can power itself on and start the boot. For more information, enter -> help /HOSTx.
	diag	The SPARC server can power on and use the Oracle ILOM default host diagnostic property values to provide fault coverage. When enabled, this option overrides user-specified Oracle ILOM diagnostic property values.
	standby	The SPARC server is prevented from powering on.
	locked	The SPARC server can power itself on; however, you are prohibited from updating flash devices or modifying the CLI property value set for /HOST send_break_action=break.
/HOST /diag/mode=	off	Prevents the power-on self-test (POST) to run for all enabled triggers
	normal	Runs the power-on self-test (POST) for all enabled triggers

POST is a group of PROM-based tests that run when the server is powered on or when it is reset. POST checks the basic integrity of the critical hardware components in the server. You can set other Oracle ILOM properties to control various other aspects of POST operations. For example, you can specify the events that cause POST to run, the level of testing POST performs, and the amount of diagnostic information POST displays.

If POST detects a faulty component, the component is disabled automatically. If the system is able to run without the disabled component, the system boots when POST completes its tests. For example, if POST detects a faulty processor core, the core is disabled, POST completes its test sequence, and the system boots using the remaining cores. Remember, /HOSTx =/Servers/PDomains/PDomain_x/HOST.

POST Variables (2 of 2)

Parameter	Values	Description
/HOSTx/diag level= For more information -> help /HOSTx/diag	Max	When enabled, the POST will run basic diagnostic tests and extensive processor and memory tests upon routine system power-on
	min	When enabled, the POST will run basic diagnostic tests upon routine system power-on
	off	When enabled, POST will not run upon routine system power-on
/HOSTx/diag verbosity=	normal	When enabled, a moderate amount of debugging output is printed to the system console. Output includes test name and results.
	min	When enabled, a limited amount of debugging output is printed to the system console
	max	When enabled, all the POST step debugging output is printed to the system console
	Debug	When enabled, an extensive amount of debugging output is printed to the system console. Output includes the names of the devices being tested, as well as the results of each test.
	None	When enabled, no debugging output is printed to the system console

Remember, `/HOSTx = /Servers/PDomains/PDomain_x/HOST`.

Configuring POST

- To set the virtual `keyswitch` to `normal`:
  `-> set /HOSTx keyswitch_state=normal`
- To set `diag mode` and `verbosity`:
  `-> set /HOSTx/diag mode=normal`
  `-> set /HOSTx/diag verbosity=max`
- To view the current values for the settings:

```
-> show /HOST1/diag
   /HOST1/Diag
      Target:
      Properties:
         default level = max
         default = verbosity = max
         error_level = max
         error_verbosity = max
         hw_change_level = max
         hw_change verbosity = max
      Commands:
         cd
         set
         show
```


The example in the slide sets the virtual `keyswitch` to `normal`, which will configure POST to run according to other parameter values. Remember, `/HOSTx = /Servers/PDomains/PDomain_x/HOST`.

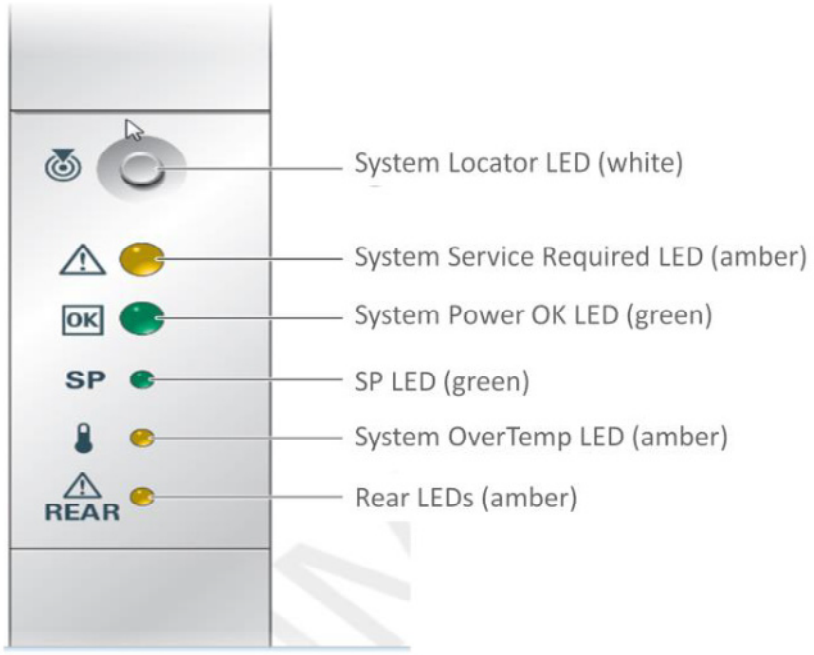
Running POST with Maximum Testing

- To set the virtual `keyswitch` to `diag`:
  `-> set /HOSTx keyswitch_state=diag`
- To reset the host so that POST will run:
  `-> reset /HOSTx`
- To return to the previous diagnostic setting after running POST:

```
-> set /HOSTx keyswitch_state=normal
-> show /HOSTx keyswitch_state
/HOSTx
   Properties:
      keyswitch state normal
```


The procedure in the slide describes how to configure the server to run the maximum level of POST for an individual PDom. The virtual `keyswitch` can be used to run full POST diagnostics without having to modify the diagnostic property settings. Note that POST diagnostics can take a significant amount of time to run at server reset. Remember, `/HOSTx = /Servers/PDomains/PDomain_x/HOST`.

Interpreting from Panel LEDs

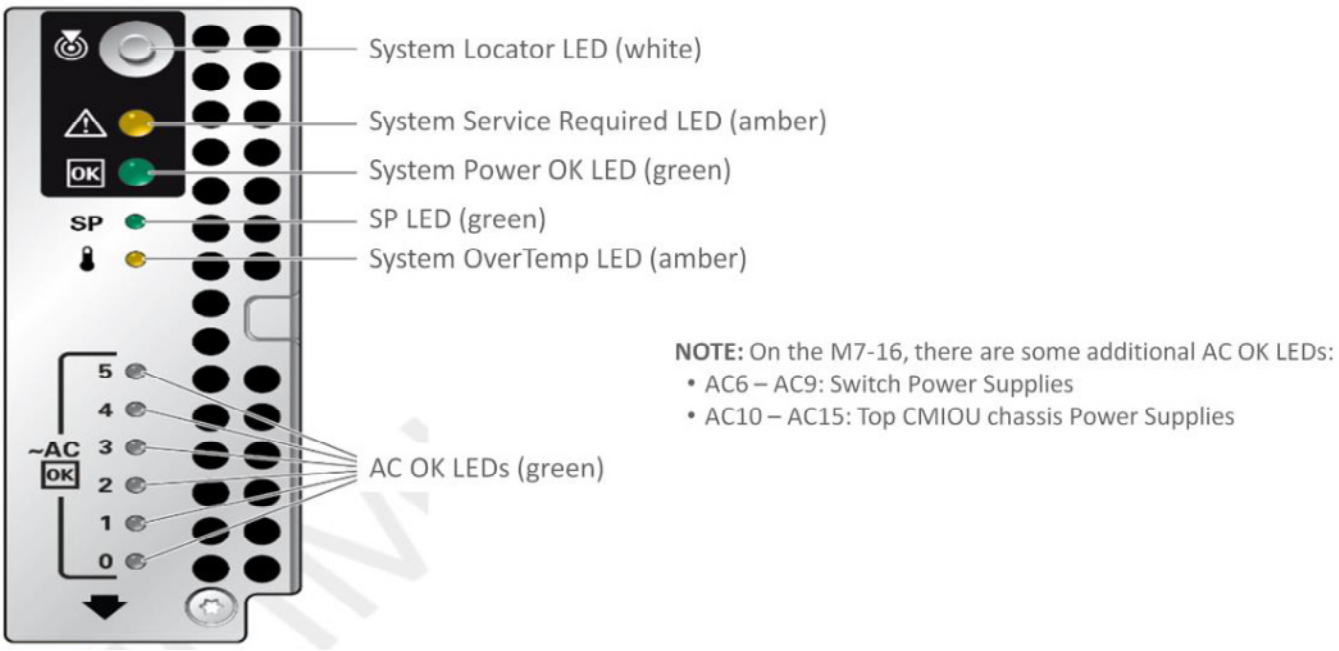


The server has two status panels, one located at the front of the server and the other at the rear. The labeling in the diagram represents:

- **System Locator LED (white):** The Locator LED can be turned on to identify the server. When on, it blinks rapidly. There are two methods for turning on the Locator LED:
  - Issuing the Oracle ILOM command set `/SYS/LOCATE value=Fast_Blink`
  - Pressing the Locator button on the front of the server
- **System Service Required LED (amber):** Indicates that service is required
  - The Oracle ILOM `show faulty` command provides details about any faults that cause this indicator to light.
  - Under some faulty conditions, individual component fault LEDs light up in addition to the Service Required LED.
- **System Power OK LED (green):** Indicates the following conditions:
  - **Off** – System is not running in its normal state. System power might be off. The SPs might still be running.
  - **Steady on** – System is powered on and is running in its normal operating state. No service actions are required.
  - **Fast blink** – System is running in standby mode and can be quickly returned to full function.
  - **Slow blink** – A normal but transitory activity is taking place. Slow blinking might indicate that system diagnostics are running or that the system is booting.

<Definitions continued in the next slide>

Interpreting Rear Panel LEDs



ORACLE

- **SP LED:** Indicates the following conditions:
  - **Off** – Indicates the AC power might have been connected to the power supplies.
  - **Steady on, green** – SPs are running in their normal operating state.
  - **Blink, green** – SPs are initializing the Oracle ILOM firmware.
  - **Steady on, amber** – A SP error has occurred and service is required.
- **System OverTemp LED (amber)** – Indicates that a temperature failure event has been acknowledged and a service action is required.
- **Rear Fault LEDs (amber)** – A rear panel component requires service.
- **AC OK LEDs (green)** – These LEDs light only after the SP LED turns green and indicate these conditions:
  - **Off** – No AC power is applied to the indicated power supplies.
  - **On** – AC power is applied to the indicated power supplies.

Monitoring the Server (1 of 2)

- To view the power state and status of all PDomains on an M7-8:

```
-> show / -level 2 -t power_state status
Target                 I Property     I Value
-----------------------+--------------+----------
/HOST                  | status       | Powered On
/System                | power_state  | On
/SP/powermgmt/budget   | status       | ok
/SP/redundancy         | status       | Active
/SP/services/fips      | status       | disabled
/SP/services/kvms/     | status       | operational
```


To view the information, the user accounts for each component must be assigned read-only `operator (o)` user roles.

Monitoring the Server (2 of 2)

- To view the power state and status of all PDomains on an M7-16:

```
-> show /Servers/PDomains/ -level 2 -t power_state status
Target                                I Property     I Value
--------------------------------------+--------------+-------------
/Servers/PDomains/PDomain_0/HOST      | power_state  | Off
/Servers/PDomains/PDomain_0/HOST      | status       | Powered Off
/Servers/PDomains/PDomain_0/System    | power_state  | Off
/Servers/PDomains/PDomain_0/System    | power_state  | Off
PDomain_0/System/DCUs/DCU_0           |              |
/ServerS/PDomains/PDomain_0/System/   | power_state  | Off
DCUs/DCU_0/CMIOU_0                    |              |
/Servers/PDomains/PDomain_0/System/   | power_state  | Off
DCUs/DCU_1                            |              |
/Servers/PDomains/PDomain_0/System/   | power_state  | Off
DCUs/DCU_1/CMIOU_4                    |              |
/Servers/PDomainS/Pdomain_0/SP/       | status       | ok
powermgmt/budget                      |              |
/Servers/PDomains/PDomain_0/SP/       | status       | operational
services/kvms/host_storage_device     |              |
```


To view the information, the user accounts for each component must be assigned read-only `operator (o)` user roles.

Monitoring the Server

- To view the power consumption and maximum allowed power:

```
-> show /System/Power
   /System/Power
      Targets:
         Power_Supplies
      Properties:
         health = OK
         health details
         actual_power_consumption = xxxx watts
         max_permitted_power = xxxxx watts
         installed_power_supplies = 6
         max_power_supplies = 6
```


To view the information, the user accounts for each component must be assigned read-only `operator (o)` user roles.

Monitoring the Server

- To view information and health status for a specific PDomain:

```
-> show /Servers/PDomains/PDomain_1/HOST
   /Servers/PDomains/PDomain_1/HOST
      Targets:
         VPS
         VPS_CPUS
         VPS_MEMORY
         bootmode
         console
         diag
         domain
         status_history
         tpm
         verified_boot
```

This server provides many ways to identify faulty behavior, including LEDs, Oracle ILOM, and POST. In addition to the systemwide and subcomponent statuses you can access with Oracle ILOM, for this server, you can view the state of individual PDomains or specific components (DCUs, CMIOUs, or CPUs). To view the information, the user accounts for each component must be assigned read-only `operator (o)` user roles.

Monitoring the Server

```
Properties:
    autorestart = reset
    autorunonerror = none
    bootfailracovery = poweroff
    bootrestart = none
    boottimeout = 0
    dcus_assignable = /SYS/DCUO /SYS/DCUl /SYS/DCU2 /SYS/DCU3
    dcus_assigned = (none)
    deus_available = (none)
    dimm_sparing = enabled
    gm_version = (none)
    hostconfig_ version = (none)
    hypervisor_version = (none}
    initiate_sp_failover = (Cannot show property)
    keyswitch_state = Normal
    macaddress = 00:10:eO:24:dO:d5
    maxbootfail = 3
    obp_version = (none)
    operation_in_progress = none
    post_version = (none)
    power_state = Off
    send_break_action = (Cannot show property)
    sp_name = (none)
    state_capture_mode = default
    state_capture_on_error = enabled
    state_capture_status = enabled
    status = Powered Off
    status_detail = 20150130 12:27:59: Host is off
    sysfw_version = (none)
```


This output is continued from

`-> show /Servers/PDomains/PDomain_1/HOST` in the preceding slide.

Monitoring the Server

- To view the health status and properties for a specific DCU:

```
-> show /System/DCUs/DCU_0
   /System/DCUs/DCU_O
      Targets:
         CMIOU_0
      Properties:
         health = OK
         health details = -
         power_state = Off
         cpu_summary = One Oracle SPARC M7
         memory_summary = 128 GB
         location = DCU0 (Domain Configuration Unit 0)
         host_assigned = /HOST0
         fan list FM0/F0 (Fan Module 0), FM0/Fl (Fan Module 0), FMl/F0 (Fan Module 1),
                  FMl/Fl (Fan Module 1), FM2/F0 (Fan Module 2), FM2/Fl (Fan Module 2),
                  FM3/F0 (Fan Module 3), FM3/Fl (Fan Module 3), FM4/F0 (Fan Module 4),
                  FM4/Fl (Fan Module 4), FM5/F0 (Fan Module 5), FM5/Fl (Fan Module 5),
                  FM6/F0 (Fan Module 6), FM6/Fl (Fan Module 6), FM7/F0 (Fan Module 7),
                  FM7/Fl (Fan Module 7)
           sp_name = /SYS/SPPl/SPM0
           initiate_sp_failover = (none)
```

Monitoring the Server

- To view information and health status details for all DCUs:

```
-> show /System/DCUs
   /System/DCUs
      Targets:
         DCU_0
         DCU_1
         DCU_2
         DCU_3
      Properties:
         health = OK
         health details = number:7087408, Serial Number:AK00180409, Reference
            Document: http://support.oracle.com/msg/ILOM-8000-56, Part
            Number:7087408, Serial Number:AK00180409,  Reference Document:
            http://support.oracle.com/msg/ILOM-8000-56 Type 'show
            /System/Open_Problems' for details.
         installed deus = 4
         max deus = 4
```

Monitoring the Server

- To view the health status and properties for a specific CMIOU:

```
-> show /System/DCUs/DCU_0/CMIOU_0
   /System/DCUs/DCU_0/CMIOU_0
         Targets:
         Properties:
         health = OK
         health details = -
         requested_state = Enabled
         power_state = Off
         model = ASSY CMIOU
         location = CMIOUO (CPU Memory IO Unit 0)
         part_nurnber = 7084599
         serial number = 465769T+l4056COONR
         action = (none)
```

Monitoring the Server

- To view the health status and properties for a specific CPU:

```
-> show /System/Processors/CPUs/CPU 0
   /System/Processors/CPUs/CPU_0
      Targets:
      Properties:
         health = OK
         health details = -
         requested_state = Enabled
         part number = Not Available
         serial number = 0000000000000000000b9061208020c8
         location = CMIOU0/CM/CMP (CPU Memory IO Unit 0)
         model = Oracle SPARC M7
         max clock speed = 4.133 GHz
         total cores = 32
         enabled cores = 32
         temperature = Not Available
```

Displaying Current Configuration State

- To display the current configuration state:

```
-> show disabled
Target       I  Property              I  Value
-----------+-------------------+-----------------------
/SYS/SWUO    disable reason     I  Diagnosed faulty

-> show /SYS -t -1 all current config state [•E]* current_config_state disable reason
Target                    I  Property              I  Value
------------------------+-----------------------+------------------------------
  /SYS/SPPO/PCIE SWITCH   current config state      Degraded
  /SYS/SPPO/PCIE-SWITCH   disable-reason            None
  /SYS/SWUO               current config state      Disabled
  /SYS/SWUO               disable-reason            Diagnosed faulty
  /SYS/SWUO/CORE SYNTH0   current config state      Disabled
  /SYS/SWUO/CORE-SYNTH0   disable-reason            Parent resource disabled
  /SYS/SWUO/CORE-SYNTHl   current config state      Disabled
  /SYS/SWUO/CORE=SYNTHl   disable-reason            Parent resource disabled
```

Displaying Server Information: CLI

- To display information about the components installed in the server:

```
-> show components
Target                              I      Property           I Value
------------------------------------+--------------------------+---------
/SYS/CMIOU0                         | current_config_state     | Enabled
/SYS/CMIOU0/CM/CMP                  | current_config_state     | Enabled
/SYS/CMI0U0/CM/CMP/BOB00            | current_config_state     | Enabled
/SYS/CMIOUO/CM/CMP/BOB00/CH0        | current_config_state     | Enabled
/SYS/CMIOUO/CM/CMP/BOB00/CH0/DIMM | current_config_state     | Enabled
/SYS/CMI0U0/CM/CMP/BOB00/CH1        | current_config_state     | Enabled
/SYS/CMI0U0/CM/CMP/BOB01            | current_config_state     | Enabled
/SYS/CMI0U0/CM/CMP/BOB01/CH0        | current_config_state     | Enabled
/SYS/CMI0U0/CM/CMP/BOB0l/CH0/DIMM | current_config_state     | Enabled
```

Resetting a PDOM

- To specify that the boot sequence should stop at the `ok` prompt:

```
-> set /Servers/PDomains/PDomain_x/HOST/bootmode
   script="setenv auto-boot? False"
```

- One way to reset a physical domain:

```
-> reset /Servers/PDomains/PDomain_x/HOST
```

- If a graceful reset is not possible, you can perform a forced hardware reset:

```
-> reset -force /Servers/PDomains/Pdomain_x/HOST
```

Note: Setting the `auto-boot` parameter to `false` is a one-time setting. The next time a PDomain is reset, the `auto-boot` parameter returns to its default setting.

Using Oracle Explorer

- After downloading Oracle Explorer, you can install the Oracle Services Tool Bundle components by starting the installer, `install_stb.sh`.
- To run Oracle Explorer to gather your system's information (`-g` generates the default file):

```
# /opt/SUNWexplo/bin/explorer -g
```

Oracle Explorer is a collection of shell scripts and a few binary executables that gathers information and creates a detailed snapshot of a system's configuration and state. Explorer output enables Oracle's engineers to perform assessments of the system by applying the output against a knowledge-based rules engine. Information related to drivers, patches, recent system event history, and log file entries is obtained from the Explorer output. It can be used by Oracle and Oracle's customers to identify and solve problems both reactively, to expedite problem diagnosis and resolution, and proactively, to prevent future problems.

Using Snapshot: CLI

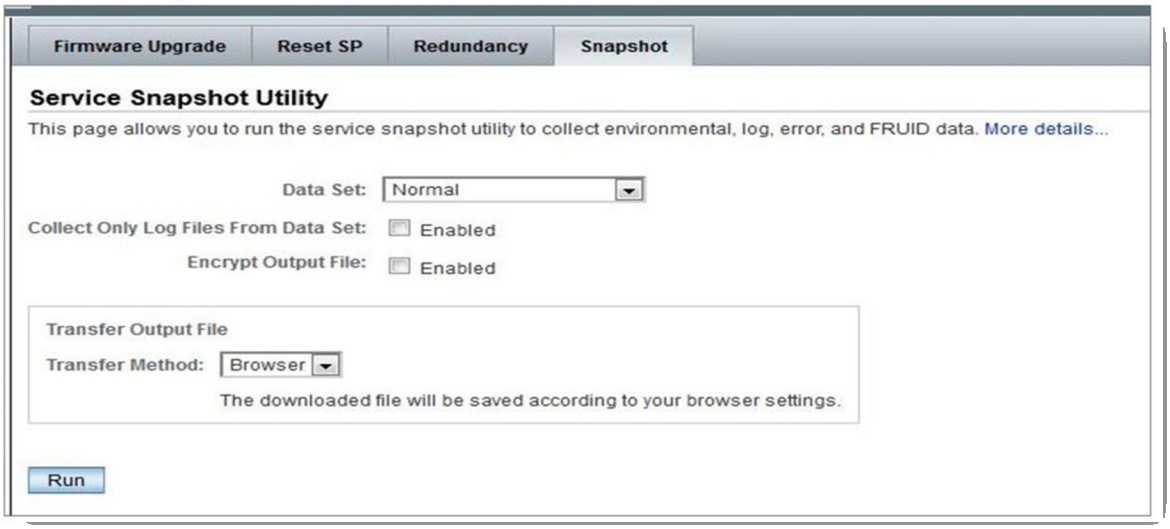
- Snapshot is part of ILOM on the SP:

```
-> help /SP/diag/snapshot
   /SP/diag/snapshot  : Save SP Snapshot for Diagnostic Purposes
      Targets:
      Properties:
         dataset : dataset
         dataset : Possible values = normal, normal-logonly, fruid, fruid-logonly,
                   full, full- logonly
         dataset : User role required for set = a
         dump_uri: initiate snapshot to URI. URI syntax and examples:
            ftp://user[:password]@host//absolute-directory-path/
            ftp://user[:password]@host/relative-directory-path/
            sftp://user[:password]@host/absolute-directory-path/
            sftp://jane@1.2.3.4/tmp/
         dump_uri: Possible values = sftp, ftp
         dump_uri: User role required for set = a
         encrypt_output : encrypt snapshot output file
         encrypt_output : Possible values = true, false
         encrypt output : User role required for set = a
         result : snapshot command result
```


The Snapshot utility provides a single solution to collect SP data for use by Oracle Services personnel to diagnose problems. The utility collects log files, runs various commands and collects their output, and sends the data collection as a zip file to a user-defined location. The resulting file is a zip file.

It is possible to invoke Snapshot in normal mode by using DMTF CLI and BUI. Collecting the snapshot requires the “a” role. Snapshot supports the SFTP (Secure File Transfer Protocol) and FTP (File Transfer Protocol) download protocols, as well as HTTPS when using the browser as the target in the BUI. Snapshot also supports encrypting the entire output file.

Using Snapshot: BUI



The Service Snapshot utility collects data about the current state of the service processor, including environmental data, logs, and FRU information. It can also run host diagnostics and capture the log. The output from Snapshot is saved as a zip or an encrypted zip file. This information is used by Oracle Service for diagnostic purposes.

The `Admin (a)` role is required to run the Snapshot utility. The `Host Control and Reset (r)` role is required for running diagnostics that reset the host. Snapshot is available from the BUI under the ILOM Administration > Maintenance tab. Click the More details… link to gather additional information about the fields in this window.

Summary

In this lesson, you should have learned how to:

- Discuss the process for diagnosing faults
- Use diagnostic tools to help when troubleshooting
- Check for faults
- View and interpret log files and system messages
- Manage the event and audit logs
- Configure POST
- Interpret system LEDs
- Monitor the server
- Display server information
- Use Oracle Explorer and snapshot scripts to gather domain information