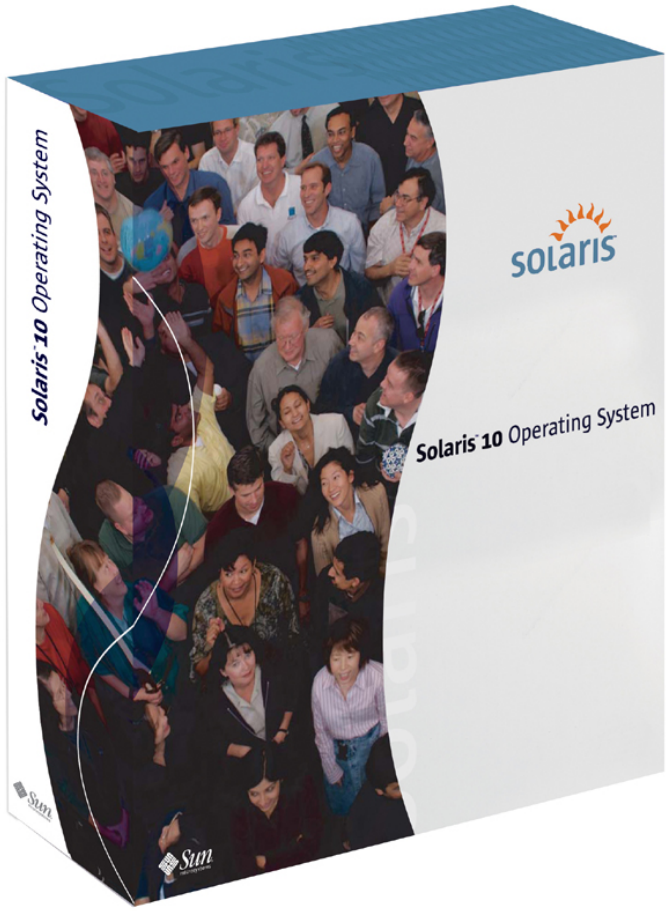
Harnessing the Power of Solaris 10 for IDS v11

Session Time:
Mon, 28 Apr, 3:30 PM
Location: Quail Creek I & II

Ravi Kota, Software Engineer
Sun Microsystems, Inc

Solaris 10 (SPARC and IA-32/64)



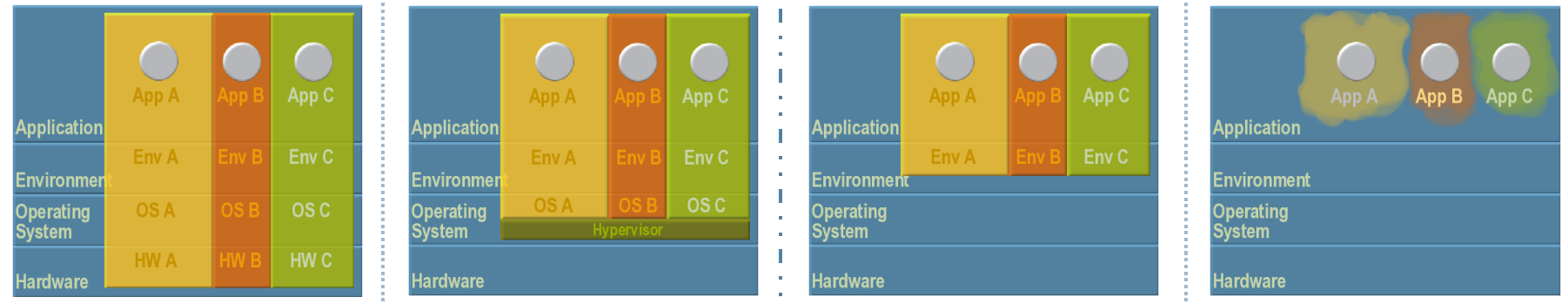
Key Features

- Solaris Containers
- Secure Execution
- Dynamic Tracing (DTrace)
- Predictive Self-Healing
- OpenSolaris.org
  - > Open source/community
- ZFS
- Runs on Sun, IBM, Dell, HP

Virtualization capabilities with no additional cost

Sun's Virtualization Technologies

Hard Partitions **Virtual Machines** **OS Virtualization** **Resource Managem**



Multiple OS's **Single OS**

Trend to flexibility **Trend to isolation**

Dynamic System Domains

Logical Domains
Sun xVM
VirtualBox
VMware

Solaris Containers

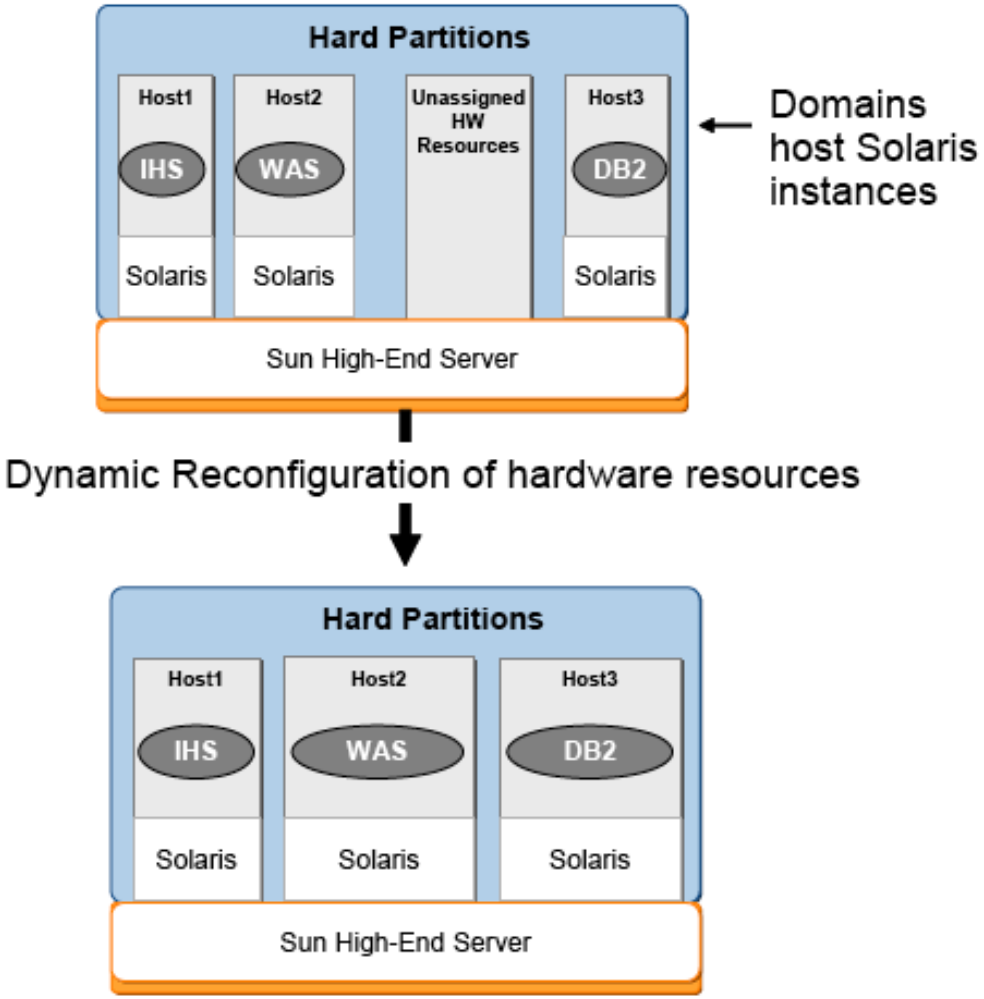
Solaris Containers for Linux Applications

Solaris Trusted Extensions

Solaris Resource Manager (SRM)

Dynamic System Domains

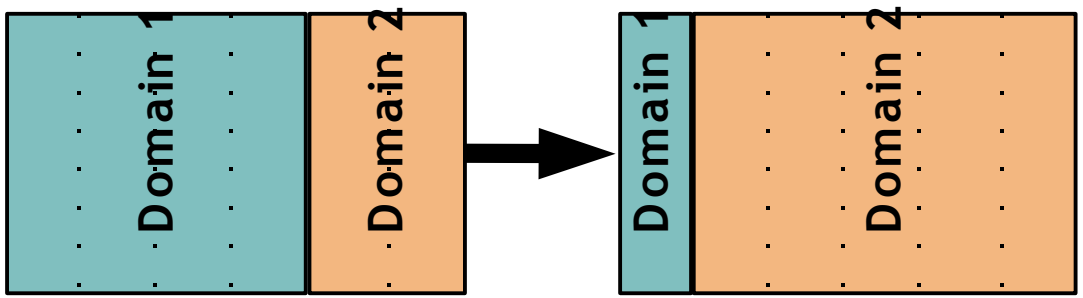
- Mainframe capability
- Open technologies

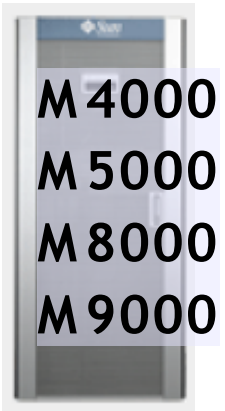


Dynamic System Domains

Hard Partition Technology

- Mid-range and high-end Sun servers support DSD
- First introduced on Sun Enterprise 10000 in 1997
- Ability to separate a pool of hardware resources into sections, electrically isolated from each other
  - Extreme high fault isolation
  - Flexibility to “move” hardware from one domain to another by simply entering a command





Dynamic System Domains

Achieving Consolidation Goals

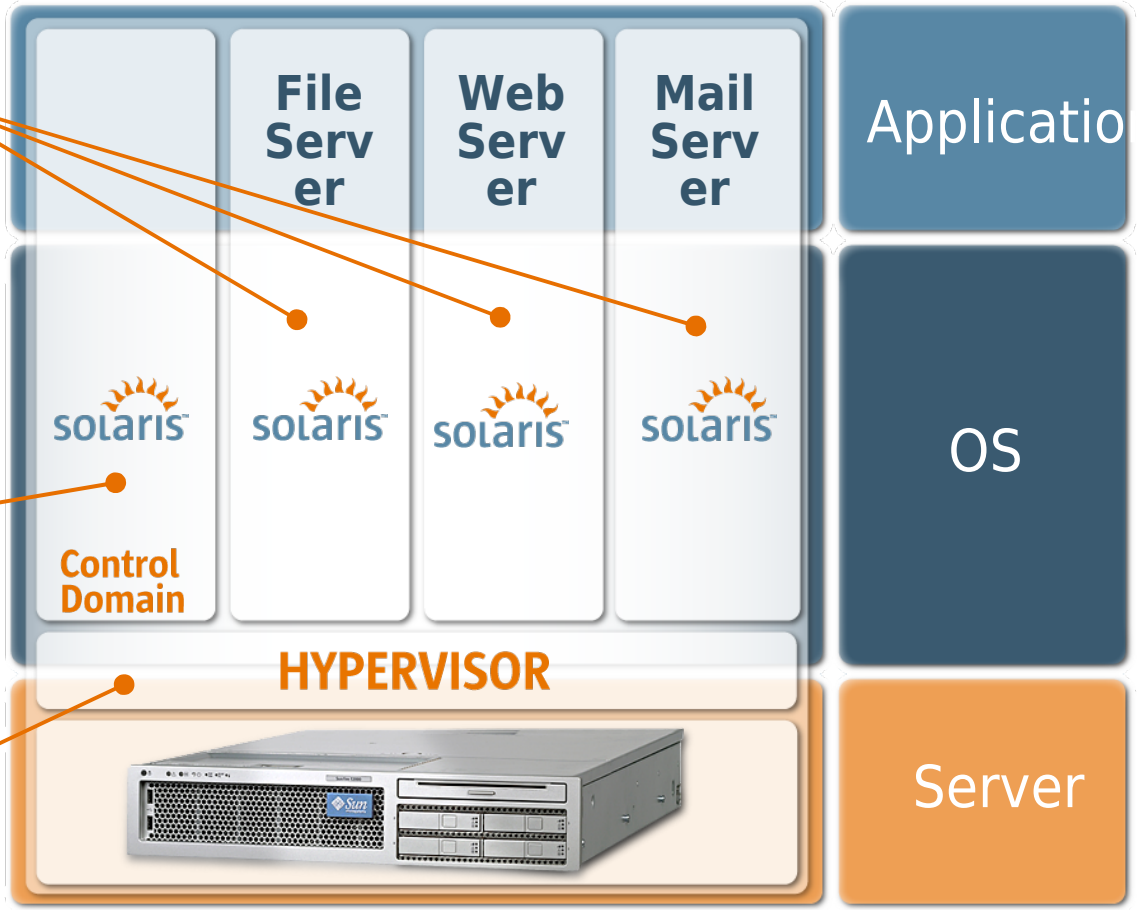
- Reduce Hardware
  - Two smaller systems can be combined into a larger, flexibly-sized domain, reducing hardware, OS, and mgmt costs
  - A system with a first-shift load can be combined with a second-shift system to reduce total resources needed, including floor space and reduce hardware support costs
- Maintain Service Levels
  - Service levels will increase through improved hardware redundancy
- Maintain Architectural Flexibility
  - Flexibility will increase by using the dynamic features of DSD

Logical Domains (LDoms)

Guest Domains

Control Domain

Ultra lightweight Hypervisor in the firmware



Logical Domains (LDoms)

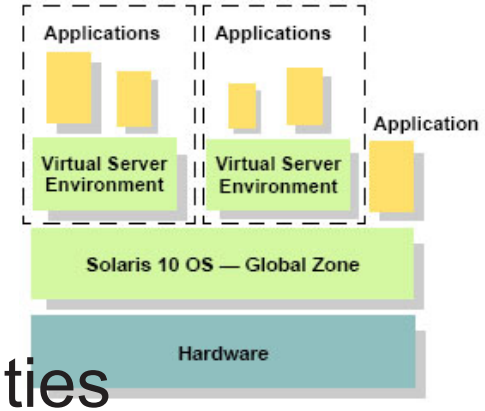
- Server Virtualization: Supported on UltraSPARC T1, T2 based systems and future CMT systems
  - Sun Fire T5220, T5210, T1000, T2000, etc.
- Up to 64 LDoms per server today
  - CPU thread individually assignable to different Ldoms
  - Partitioning of a physical system into multiple virtual machines
- Each virtual machine should appear as an entirely independent machine
  - own OS instance, patches, tuning parameters
  - own user accounts, administrators
  - own disks, network interfaces, MAC & IP addresses

Logical Domains (LDoms)

- Control/Service Domain and Guest Domains
  - Control Domain hardening
- Each guest domain can be configured, started and stopped independently
  - Without requiring a power-cycle of machine
- Ability to dynamically add and remove vCPUs while OS is running
- Many benefits
  - Greatly increased utilization
  - More flexibility in services deployment in data center

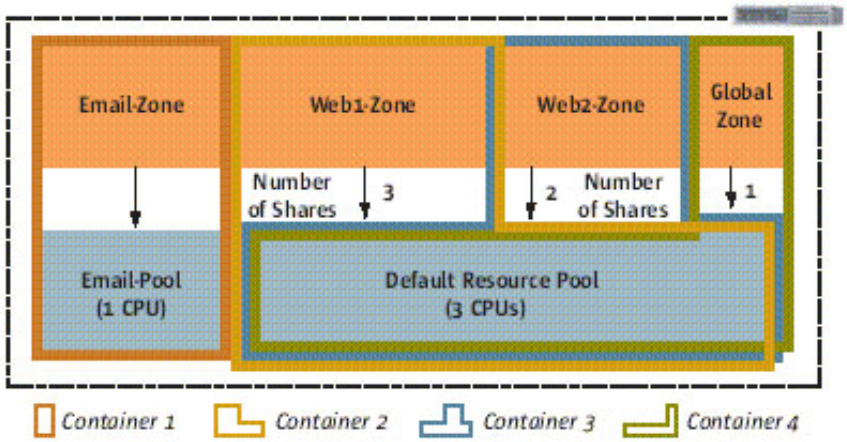
Solaris Containers

- Solaris Containers provide isolated and virtualized software application deployment (e.g. IDS v11 Application Environment)
  - > Available on all systems that run Solaris 10
- Solaris Containers consists of
  - > Zones
  - > Resource Management
  - > Dynamic Resource Pools
- Zones provide the virtualization capabilities
- Resource management allows partitioning of resource utilization

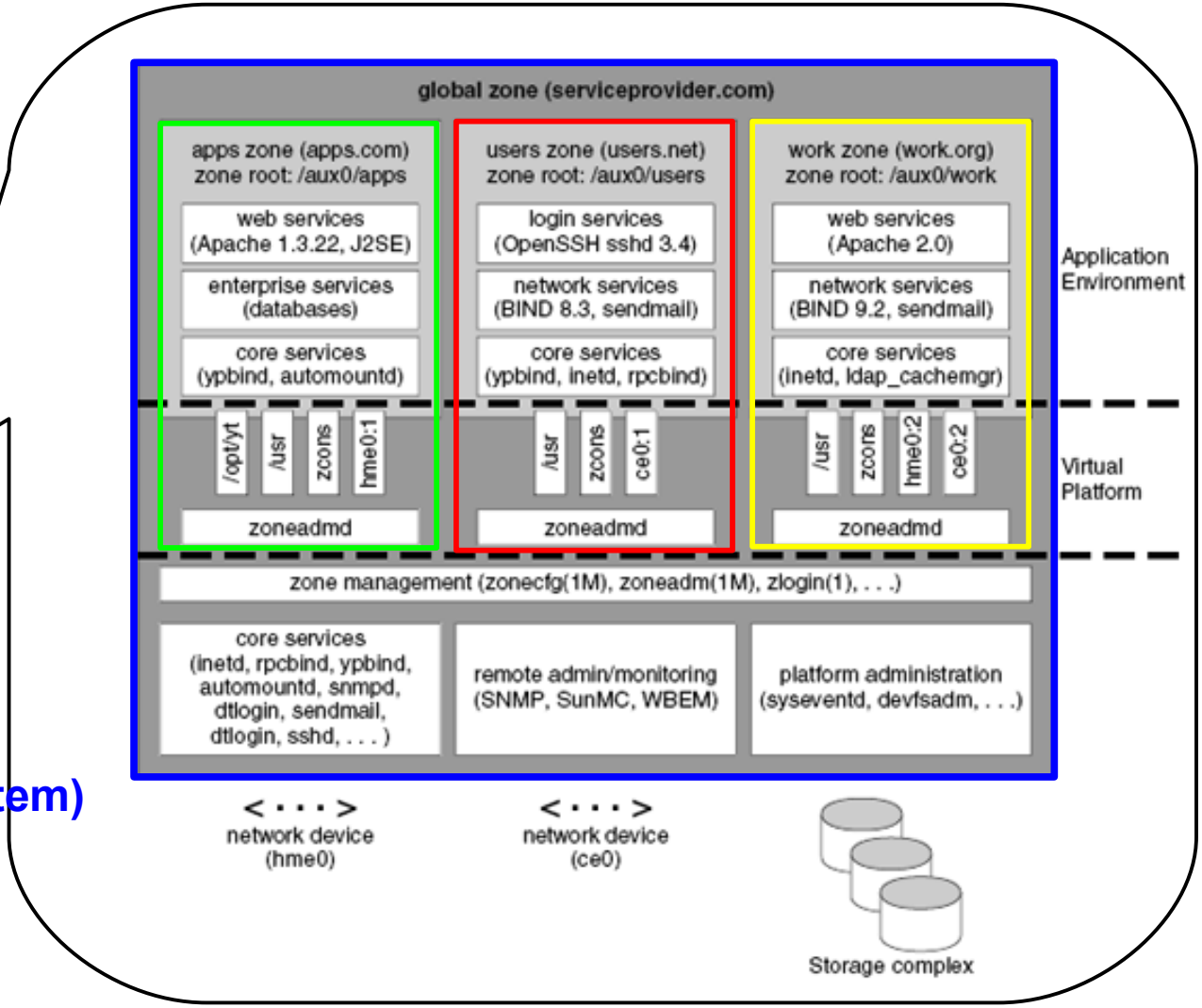


Solaris Containers

- Resource pools can be defined to manage workload among zones allowing several isolated IDS v11 environments on the same server
- Solaris 10 8/07 adds new capabilities to cap memory (physical/virtual), dedicate CPU's, choose FSS for zone configuration
- Solaris 8 Migration Assistant to host Solaris 8 Containers on Solaris 10



Zones



Legend

-- Global Zone (the system)

-- Zone 1

-- Zone 2

-- Zone 3

Zones

- Provides virtualized OS environments, each looking like a Solaris instance
  - Implemented via a lightweight layer in the OS
  - Details of physical resources are hidden
  - Separate host name, IP address, IP port space
  - Processes cannot see or affect processes in other containers
  - Each zone can be administered independently
  - No porting as the ABI/API is the same

Zone's granularity

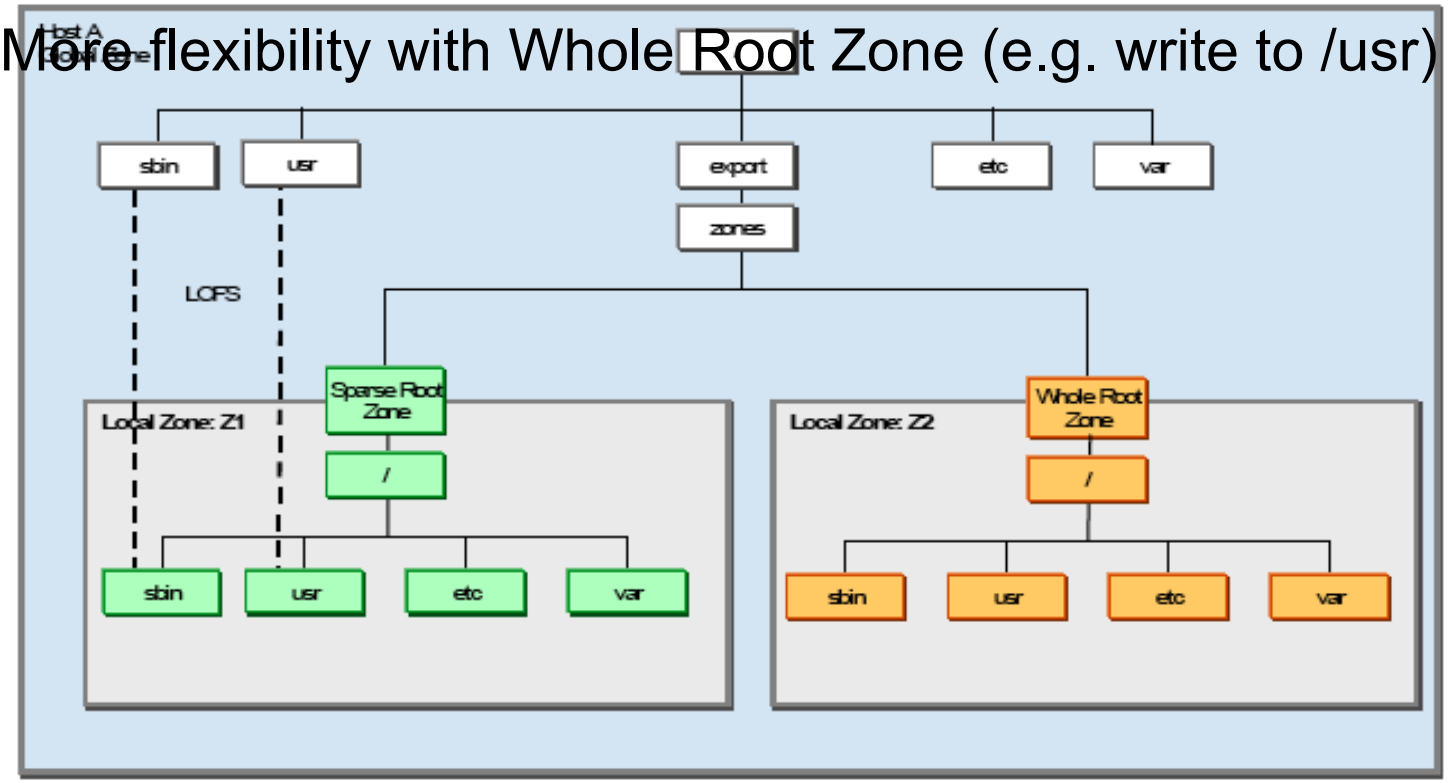
- 8,000+ zones per OS instance
  - 140,000+ zones on an SF25K
- Does not require dedicated CPU's, memory, physical devices, etc.
  - Just the disk space for unique root filesystem
- Existing hardware resources can be:
  - multiplexed across containers, or
  - allocated per container using resource pools

Zone's security

- Security boundary around each zone
- Restricted subset of privileges
  - A compromised container is unable to escalate its own privileges
- Important name spaces are isolated
- Processes running in a container are unable to affect activity in other containers or the global zone

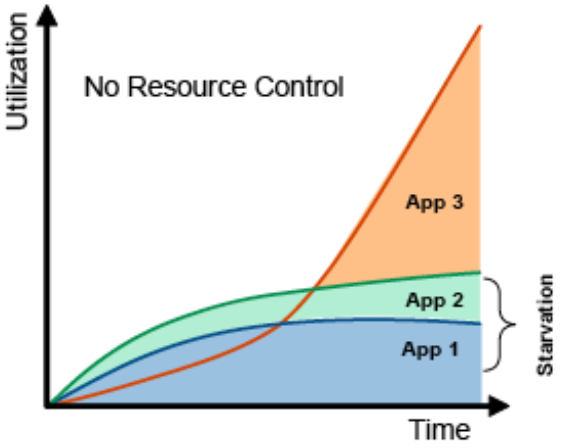
Two types of Zones

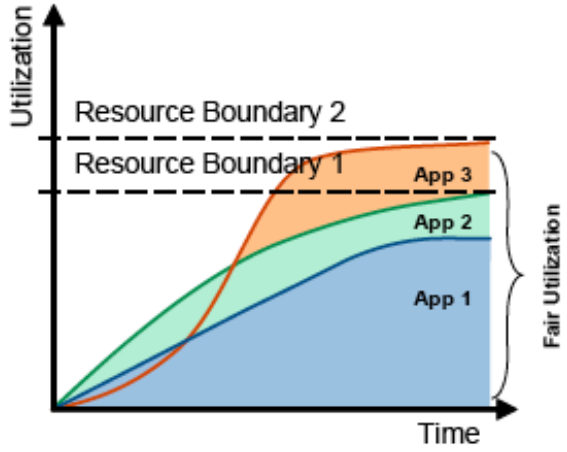
- Sparse Root Zone and Whole Root Zone
  - > Loopback File System (LOFS) are pointers to system directories
  - > Less space, quicker maintenance time with Sparse Root Zone
  - > More flexibility with Whole Root Zone (e.g. write to /usr)



Resource Management

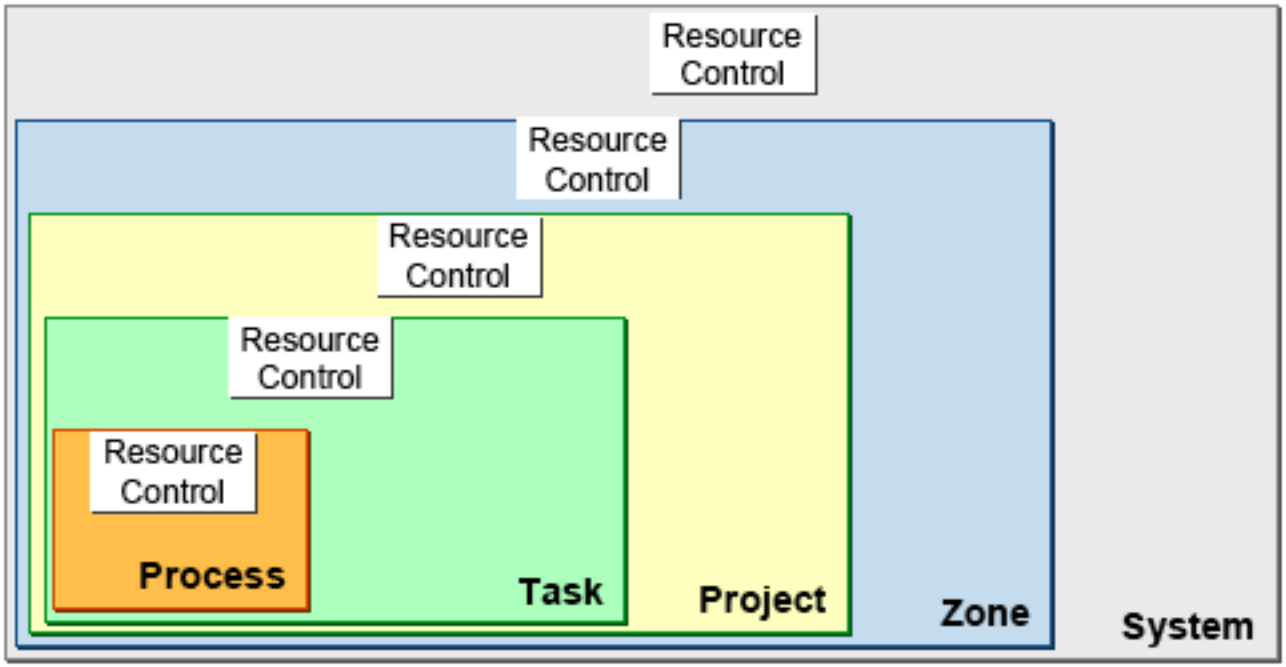
- Without Resource Management
  - A misbehaving application can hog the system
  - Causing resource starvation in other applications
- With Resource Management
  - Different resource management policies can be put in place
  - System utilization is more predictable (meeting SLA)
  - Available for various scopes





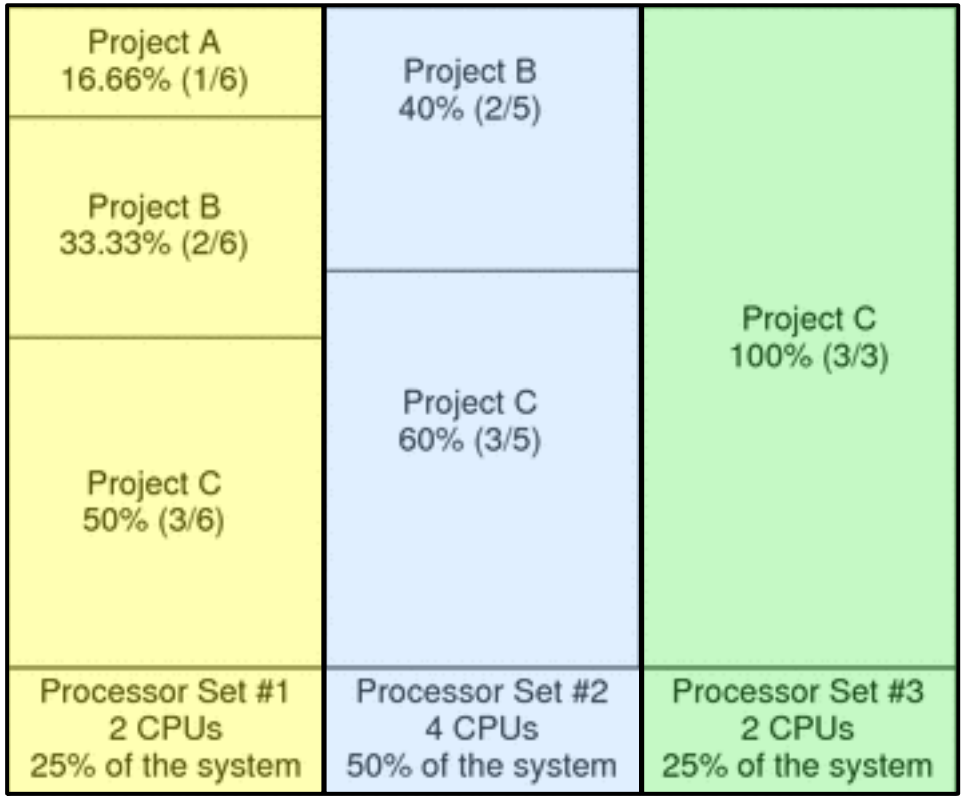
Resource Management

- Scope of Resource Management in Solaris
  - > control resources upon the desired level



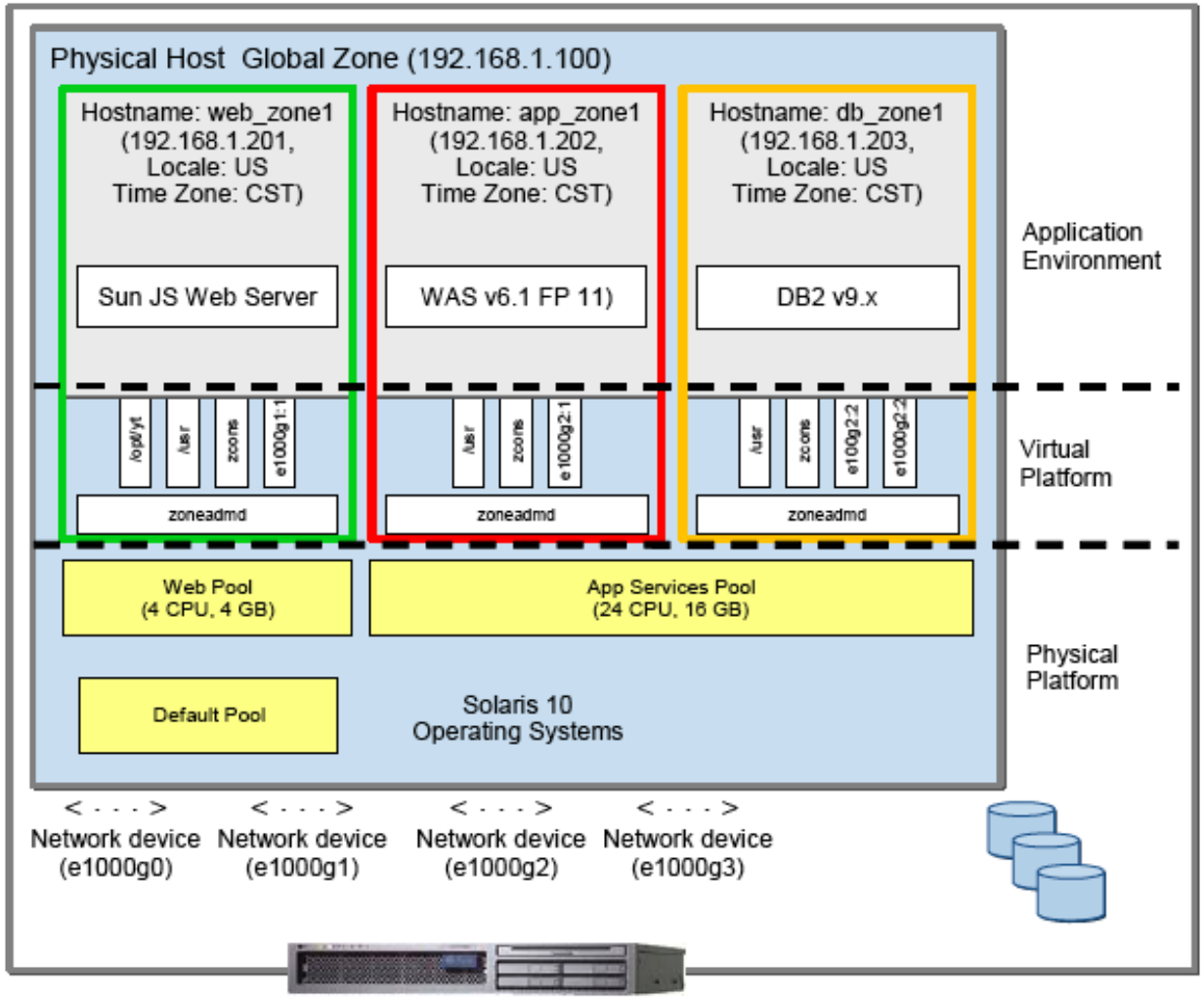
Processor Sets and Pools

- A resource partitioning mechanism in Solaris
- Assign a dedicated set of processors for a specific use
- Can dynamically reconfigure the processors from one set to another
- Solaris 10 8/07 has new zone configuration feature for dedicated processors



Solaris Containers with Resource Pools

- Containers can have different resource pools



Example: A Container (Zone) with Resources

- An example to create a Solaris Container with a pool comprising a processor set with 2 “processors”

```
-- Enable resource pools
# pooladm -e

# pooladm -s

--
-- Create separate resource pools for each zone and save the configuration
--

# poolcfg -c 'create pset pset_app_zone1 (uint pset.min = 2; uint pset.max = 2)'
# poolcfg -c 'create pool pool_app_zone1'
# poolcfg -c 'associate pool pool_app_zone1 (pset pset_app_zone1)'


# zonecfg -z app-zone1
app-zone1: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:app-zone1> create
zonecfg:app-zone1> set zonepath=/export/app-zone1
zonecfg:app-zone1> set autoboot=true
zonecfg:app-zone1> set pool=pool_app_zone1
zonecfg:app-zone1> add net
zonecfg:app-zone1:net> set physical=e1000g1
zonecfg:app-zone1:net> set address=129.110.14.21
zonecfg:app-zone1:net> end
......
zonecfg:app-zone1> verify
zonecfg:app-zone1> commit
zonecfg:app-zone1> exit
```

Example#2: Container (Zone+Resources)

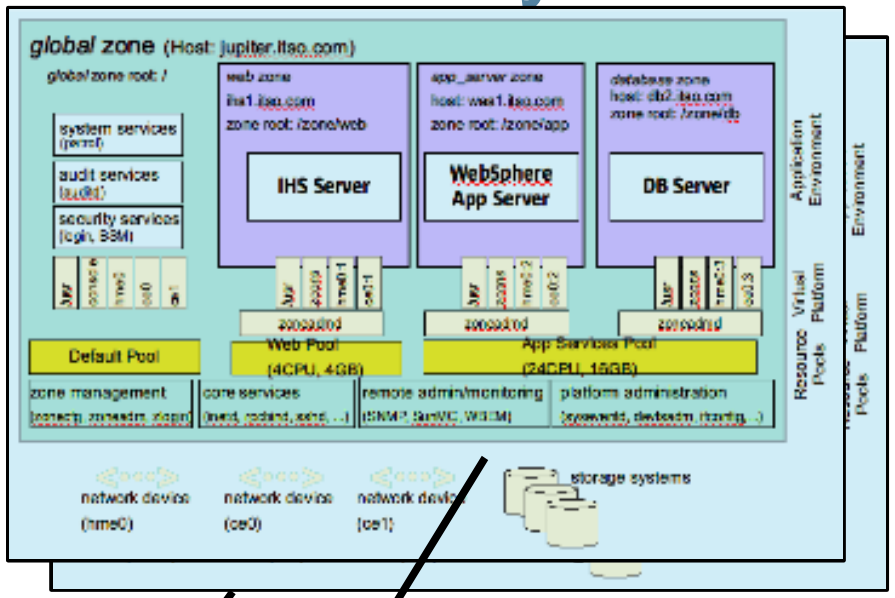
- Another example to create a Solaris Container with a processor set with 2 “processors” as well as capped memory to 1GB physical and swap space using the Solaris 10 8/07 zone features

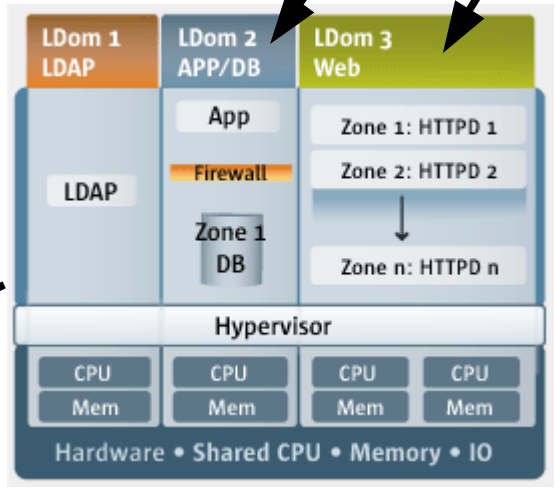
```
# zonecfg -z app-zone1
app-zone1: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:app-zone1> create
zonecfg:app-zone1> set zonepath=/export/app-zone1
zonecfg:app-zone1> set autoboot=true

zonecfg:app-zone1>add dedicated-cpu
zonecfg:app-zone1:dedicated-cpu>set ncpus=2
zonecfg:app-zone1:dedicated-cpu>set importance=2
zonecfg:app-zone1:dedicated-cpu>end

zonecfg:yourzone> add capped-memory
zonecfg:yourzone:capped-memory> set physical=1g
zonecfg:yourzone:capped-memory> set swap=1g
zonecfg:yourzone:capped-memory> end

zonecfg:app-zone1> add net
zonecfg:app-zone1:net> set physical=e1000g1
zonecfg:app-zone1:net> set address=129.110.14.21
zonecfg:app-zone1:net> end
......
zonecfg:app-zone1> verify
zonecfg:app-zone1> commit
zonecfg:app-zone1> exit
```

Selecting the right Virtualization for your business needs

Keeping Competitive edge

- Improve efficiency
- Better Manageability
- Reduce operating costs
- Increase ROI
- Faster time to market
- Enable aggressive growth





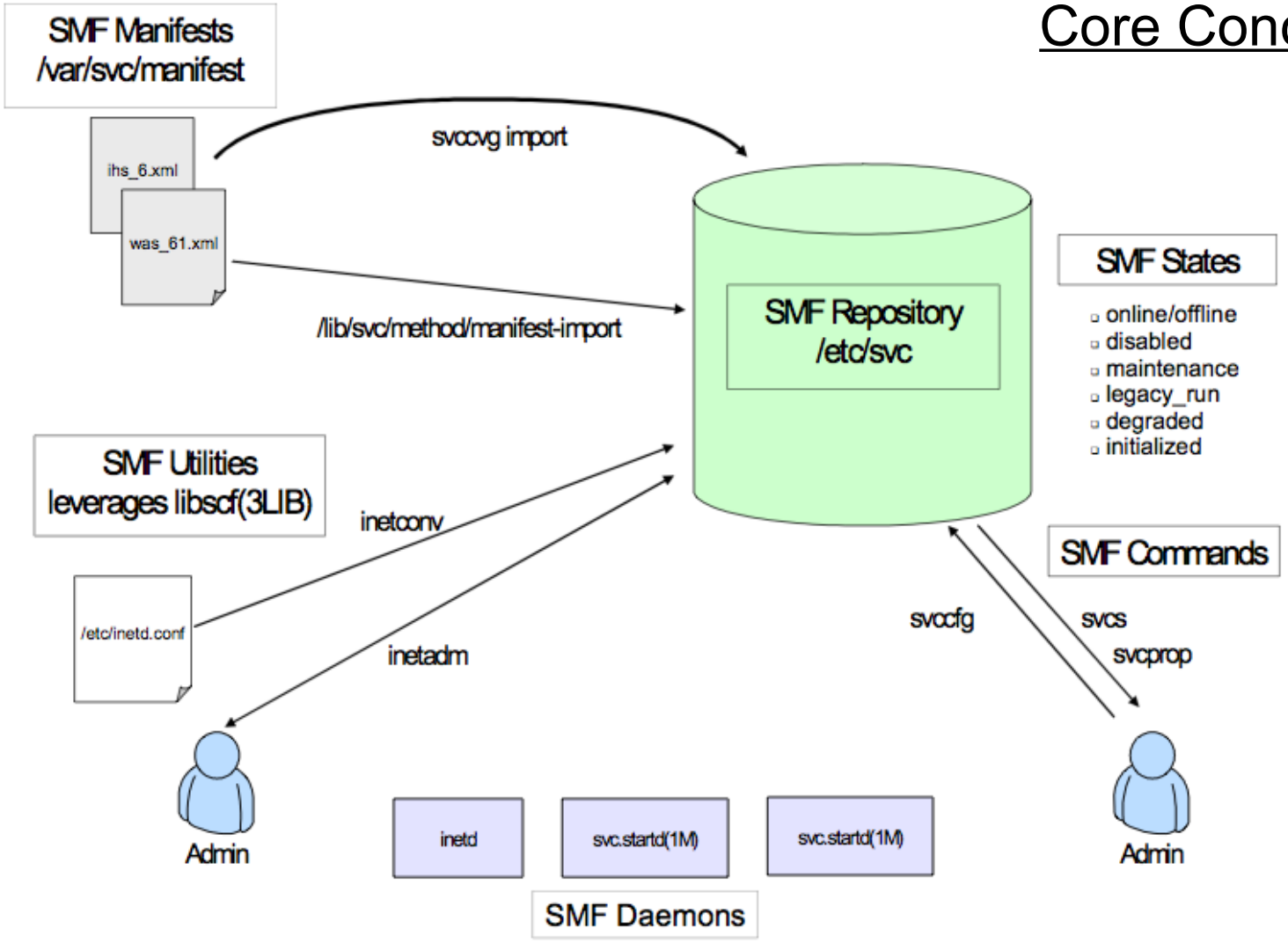
Selecting the right Virtualization for your business needs

Require	Technology	Platform
► Vertical scalability ► Large physical memory (>64GB RAM) ► Complete fault isolation ► Dynamic reconfiguration ► Reliability, Availability, Serviceablility (RAS) ► Multiple Solaris OS	Dynamic System Domains (Hardware Partitioning)	Sun Fire mid-rage SPARC64-VI to high-end servers
► High throughput ► Best price per performance ► Energy efficienciency ► Multiple Solaris OS	Logical Domains (Virtualization with Hypervisor)	Sun CoolThread servers (T5220, T5120, Sun Blade T6320, T2000, T1000, etc.)
► Virtual Systems ► Virtually no overhead ► Secure and isolated process environment even while sharing a single Solaris OS	Solaris Containers (Virtualization at the OS-Level)	Any SPARC or x86/64 (from a single CPU system to Sun Fire 25K) that runs Solaris 10 or newer
► Solaris to host multiple guest operating systems including Solaris, Linux and Microsoft Windows	Sun xVM (Virualization with a cross-platform, open source hypervisor)	(See http://www.sun.com/xvm) (*not Generally Available at this writing)

Built-in commo feature in Solaris 10

Service Management Facility (SMF)

Core Concepts



Sun microsystems

Q & A

Thank You!