
ORACLE®

Safe Habour Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

ORACLE PRODUCT LOGO

ORACLE

Ressource-Management für und mit modernen Rechnerarchitekturen

Franz Haberhauer
Chief Technologist Hardware Presales Northern Europe

Presenting with

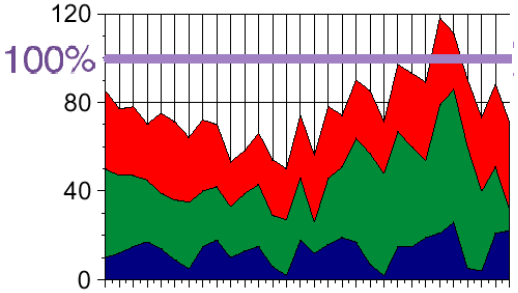
LOGO

Agenda

- Ressource-Management
- Aktuelle CPU-Features und ihre Abstraktion im OS
- Netzwerk-Virtualisierung und Bandbreiten-Management
- Storage Quality of Service

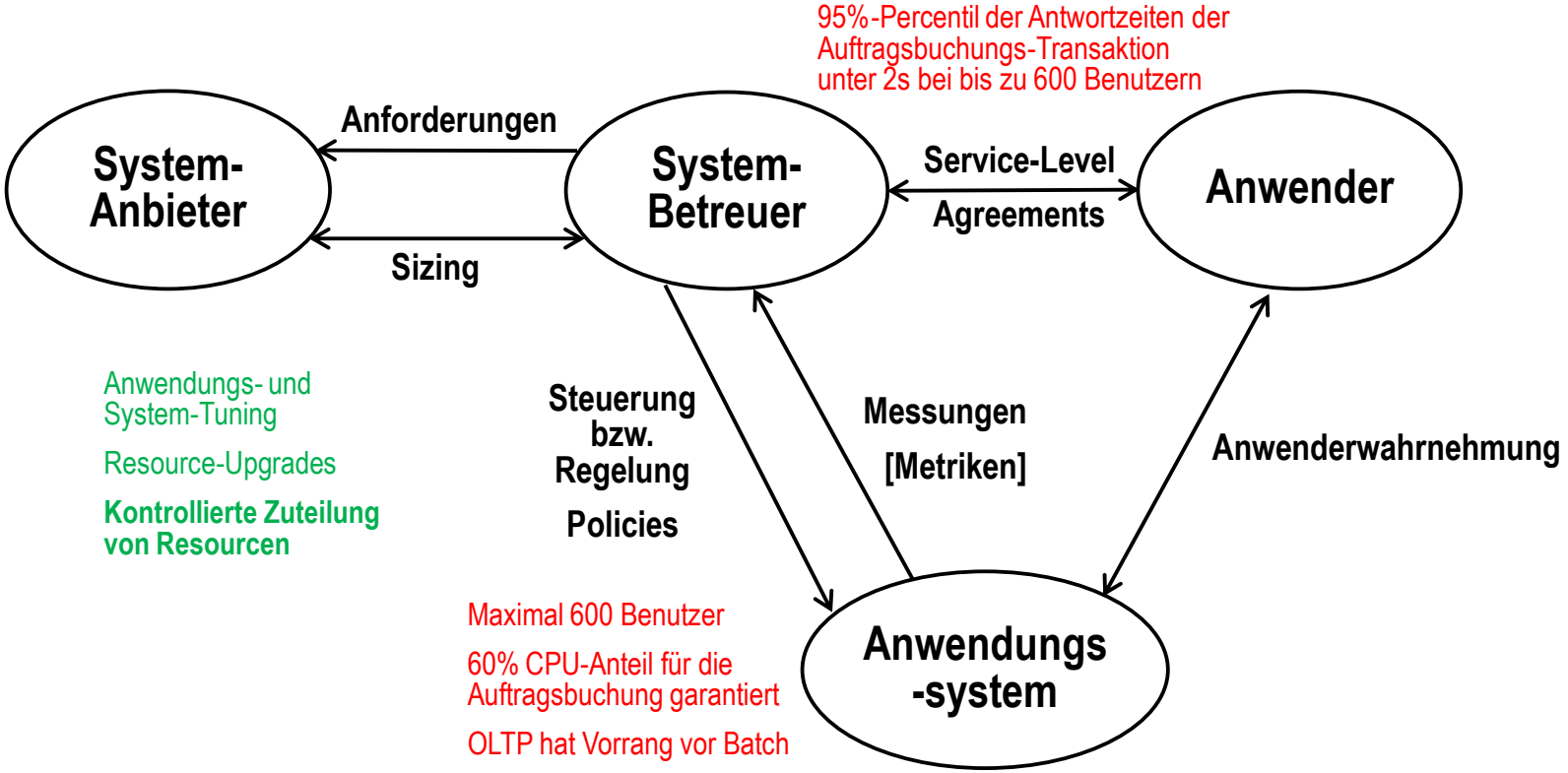
Warum Ressource-Management?

- Ressource−Management
  - Zuteilung von Ressourcen so, daß eine geforderte Dienstqualität erreicht wird.



- Insbesondere nötig bei
  - hoher Auslastung mit Lastspitzen und unterschiedlich priorisierten Lasttypen auf einem System
    - typisch bei Anwendungs− und Server−Konsolidierung
    - Cloud-Computing
- Erfordert Accounting – kann auch primäres Ziel sein

Performance-Management



Kategorien von Ressourcen

- Sich erneuernde Resourcen
  - CPU, Netzwerkbandbreite
  - kontrolliert durch Anteile, Prioritäten
- Fest vergebene Resourcen
  - Plattenplatz, Swapspace
  - kontrolliert durch Grenzwerte
  - Schutz vor Denial−Of−Service−Angriffen

Ansätze für Ressourcen-Verteilung

- Ausgleichen
  - „Fair Share“
  - Verteilung festgelegter Anteile bei 100% Auslastung
    - Unter 100% keine Beschränkung
- Deckeln (Capping)
  - Beschränkung durch festen Grenzwert
- Partitionieren
  - Exklusive Zuordnung und Nutzung

Granularität der Ressourcen-Zuordnung

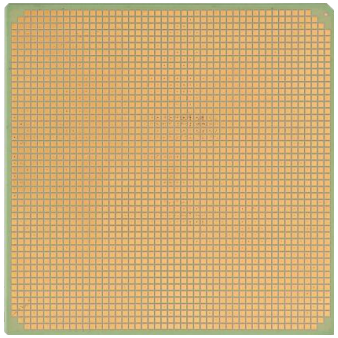
- Ressource-Instanzen
  - NIC
- Laufzeit-Instanz
  - Thread, Prozess, Prozessgruppe, Task, Zone, Virtuelle Maschine
- Accounting-Objekt
  - User, Gruppe, Projekt
- Konfiguration
  - Zone, Virtuelle Maschine

Ressource-Kontrollmechanismen

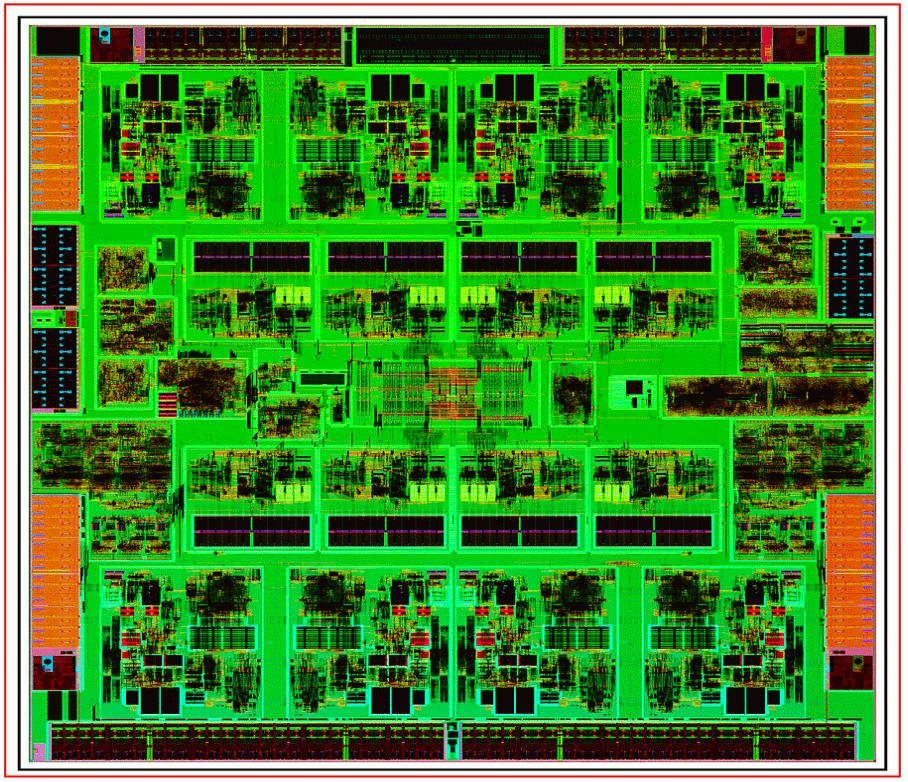
- CPU
  - Scheduler
    - Algorithmus, Prioritäten, Zeitquantum, Shares
      - dispadmin(1m), priocntl(1,2)
  - Partitionierung mit Prozessorsets/-pools
  - Zuordnungs von Interrupts
    - Reduktion von Jitter
- Speicher
  - RAM
    - Residenz via `mlock(2)`, ISM
    - Memory Placement Optimization (MPO) – NUMA
    - RSS-Limitierung über `rcapd`
  - Virtuelles Memory
- Netzwerk
  - Integrated Services (IntServ)
    - RSVP, End-to-End
  - Differentiated Services (DiffServ)
    - Type-of-Service (ToS) im IP-Header
  - Am NIC
    - Bandbreite
- Block-IO
  - Komplex
  - Bandbreite ist nicht alles
    - Sequentiell vs. wahlfreie Zugriffe
    - komplexe Speichersysteme
      - - multihosted

Was ist eine CPU?

SPARC T4 CPU



- 2117 Pins
- 403 mm^2 die size
- 8 cores
  - 15.4 mm^2 core size
- 40 nm process geometry
- ~ 855 mio. transistors



SPARC T4 Servers

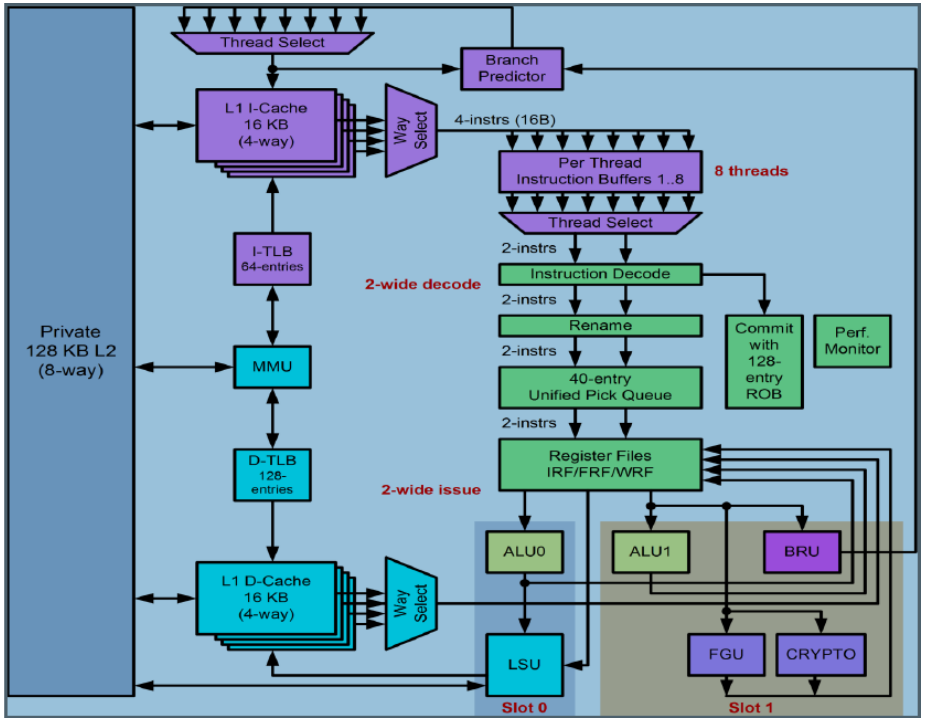
Product Line Overview




	SPARC T4-1B	SPARC T4-1	SPARC T4-2	SPARC T4-4
Processor	SPARC T4 2.85GHz	SPARC T4 2.85GHz	SPARC T4 2.85GHz	SPARC T4 3.0GHz
Max Processor Chips	**1**	**1**	**2**	**4**
Max Cores	**8**	**8**	**16**	**32**
Max Threads/Strands	**64**	**64**	**128**	**256**
DIMM Slots	16	16	32	64
Max Memory	256GB	256GB	512GB	1TB
Drive Bays	2	8	6	8
I/O Slots	2 x PCIe 2.0 EM, 2 NEM, 1 REM, 1 FEM slots	6 LP x 8 PCIe 2.0, 4 x 1GbE ports, 2 x 10GBE XAUI ports	10 x PCIe 2.0, 4 x 1GbE ports, 4 x 10GbE XAUI ports	16 x PCIe 2.0 EM, 4 x 1GbE ports, 8 x 10GbE XAUI ports
Form Factor/RU	Blade	Rack 2U	Rack 3U	Rack 5 U

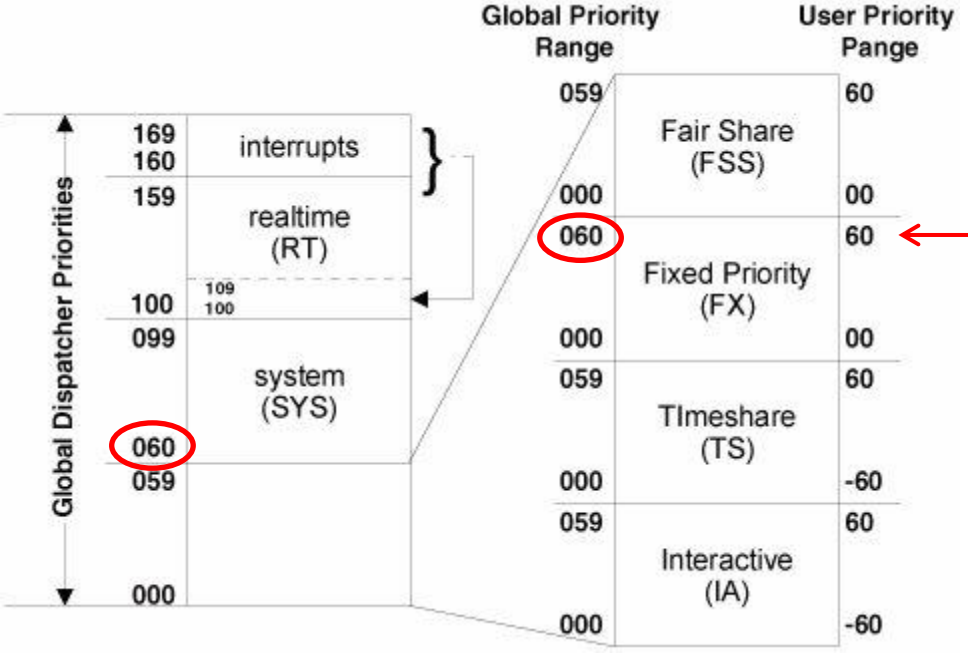
„CPUs“ in Solaris

SPARC T4 Dynamic Threading



- Many of the resources on the SPARC T4 core are shared between threads
  - Load-buffers, store-buffers, pick-queue, working-register-file, reorder-buffer, etc.
- Resources are dynamically configured between threads each cycle
  - No synchronization required

Solaris Critical Threads Optimization

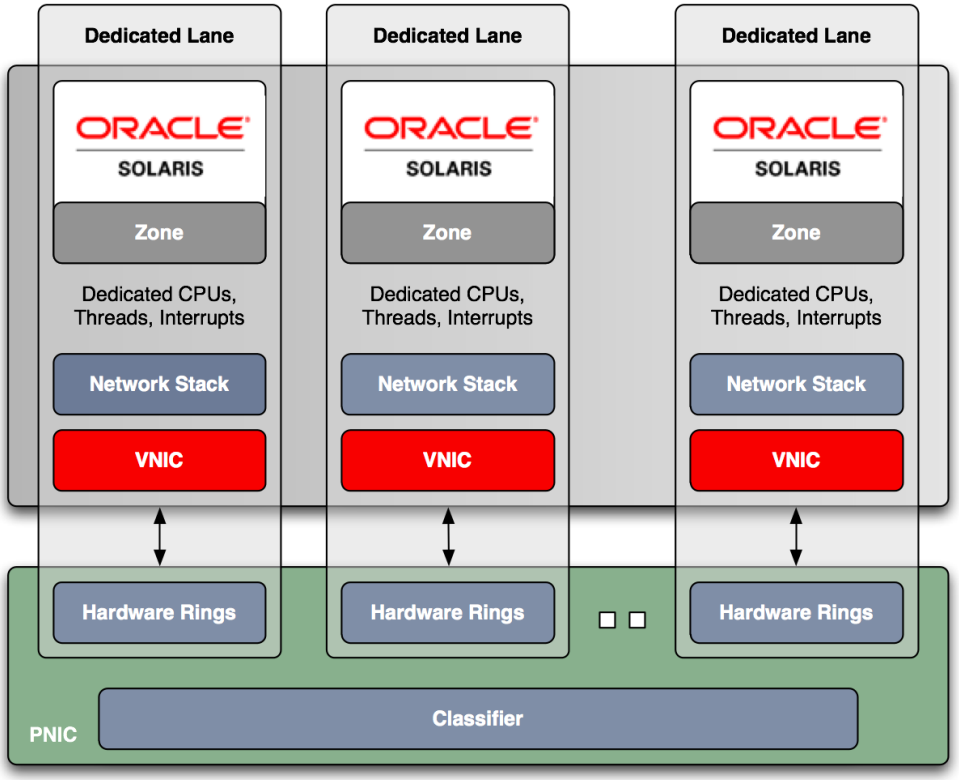


- Priorisierte Zuordnung eines LWP zu einem freien Core
  - PID 88, LWP 10
    ```
    # priocntl -s -c FX -m 60 -p 60 -i pid 88/10
    ```
- Über Partitionierung
  - Anlegen eines Prozessorset über einen Core frei von Interrupts
    Dann Binden eines einzelnen LPW
  - Option für OVM 2.1 für SPARC Gäste (`max-ipc`)
- Besonderes Potential in Middleware
- Optimierung auch in Bezug auf andere CPU-Ressourcen wie Caches, Turboboost

Parallel Network Virtualization Architecture

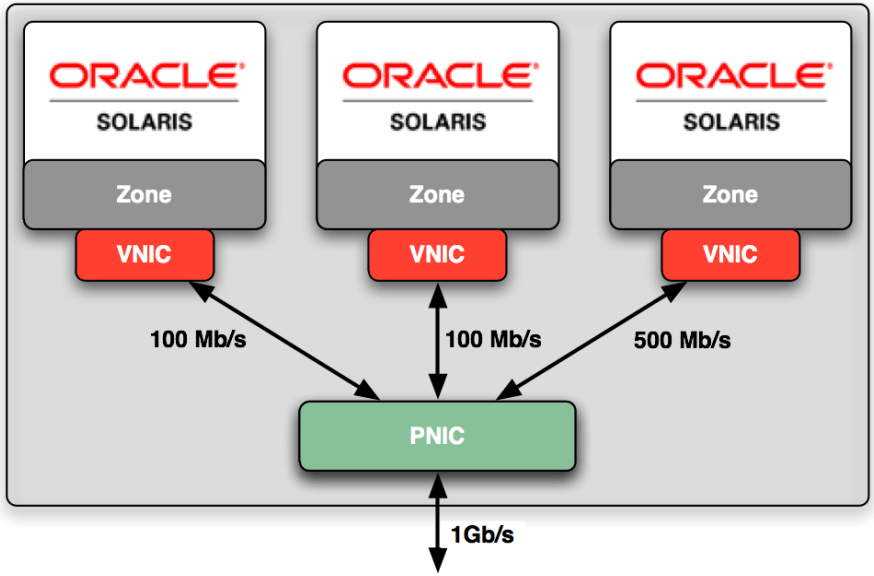
Solaris 11

- Virtualization and QoS designed-in
- Independent Hardware Lanes with dedicated resources
  (CPUs, I/O threads, interrupts):
  - from the NIC to applications
- VNIC behaves just like a regular NIC
  (link speed, stats, MAC address)
- Hardware and software fanouts for best scalability
- Adaptive polling mode depending on load



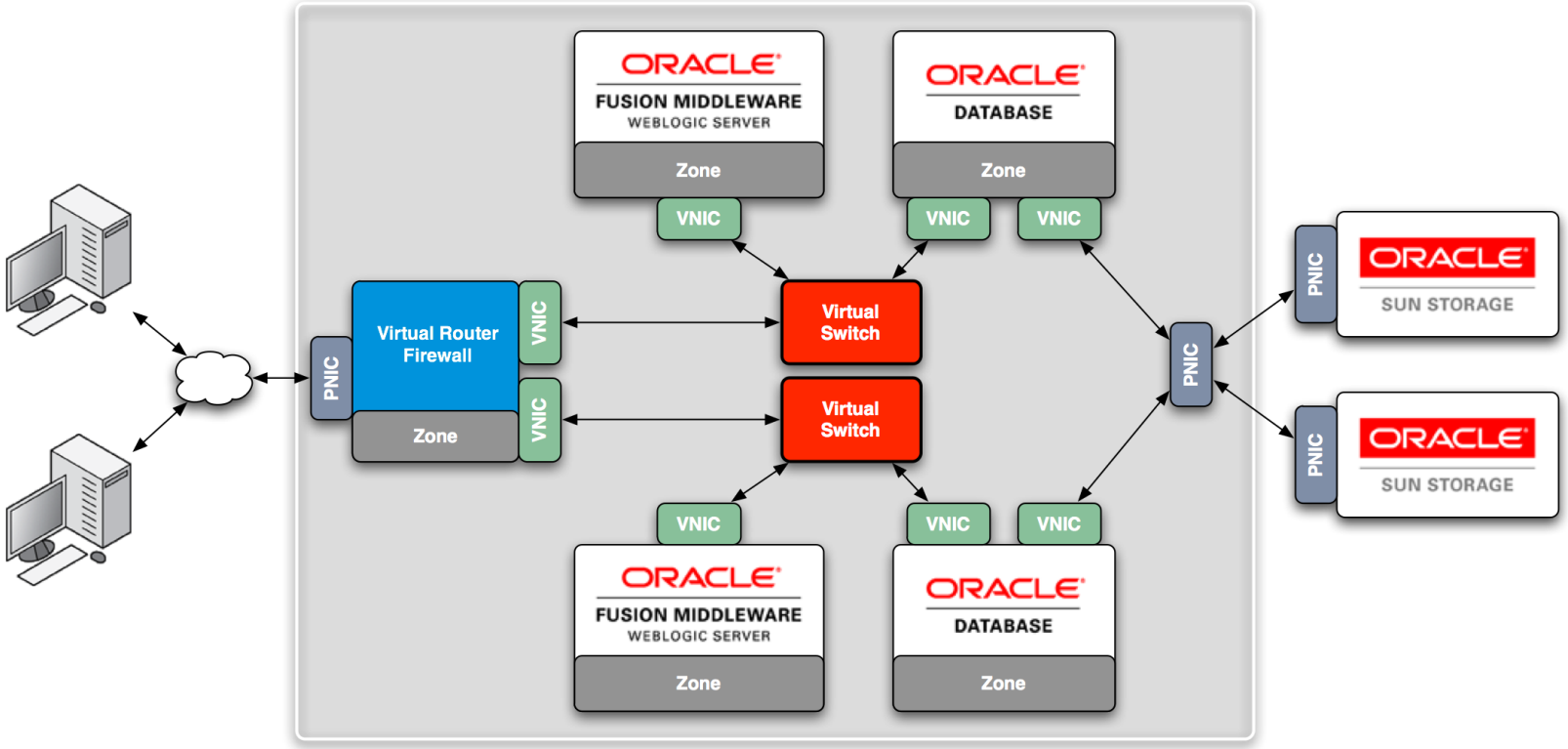
Network Resource Control

- Set bandwidth limit on a VNIC (virtual link speed)
- QoS integrated in the core stack, no separate component to configure
- Constrain the CPUs used by VNICs or data links by CPU ids or pool names
- Integrated with Solaris resource management and zones




```
# dladm create-vnic -l net0 \
-p maxbw=100M vnic0
```

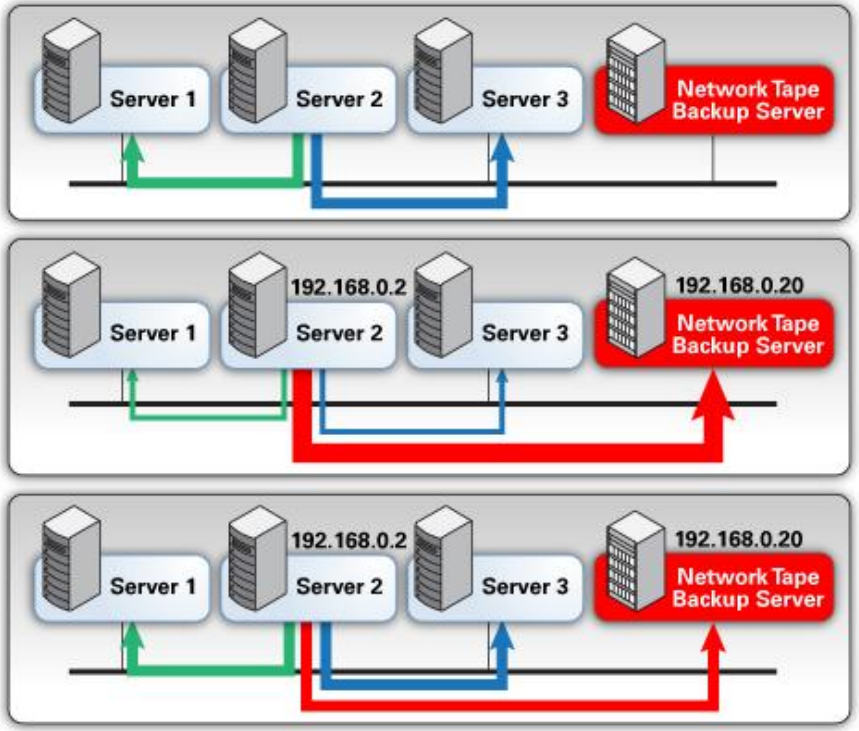
Virtual Multi-Tiered Architecture



Controlling and Observing Flows

Control the Un-Controllable

- Built-in QoS can be applied to traffic flows specified by the administrator
- Managed by flowadm(1M) and specified by source and destination IP addresses, protocol, port number, etc.
- Flows can be observed in real time with flowstat(1M), or a history can be obtained using extended accounting



Example

- Creating a software based VNIC

```
# dladm create-vnic -l ixgbe0 -p rxrings=sw,txrings=sw vnic0
```

- Quality of Service with VNICs and network flows

```
# dladm set-linkprop -p maxbw=10m vnic0
# flowadm add-flow -l vnic0 transport=tcp,local_port=80 httpflow
# flowadm set-flowprop -p maxbw=5M httpflow
```

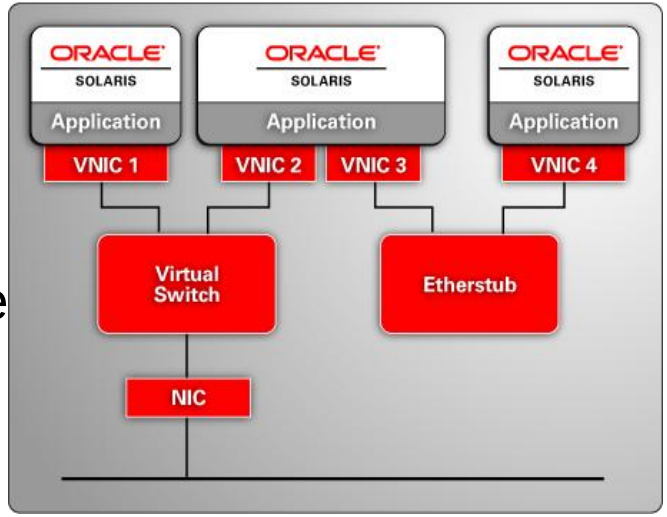
Example (2)

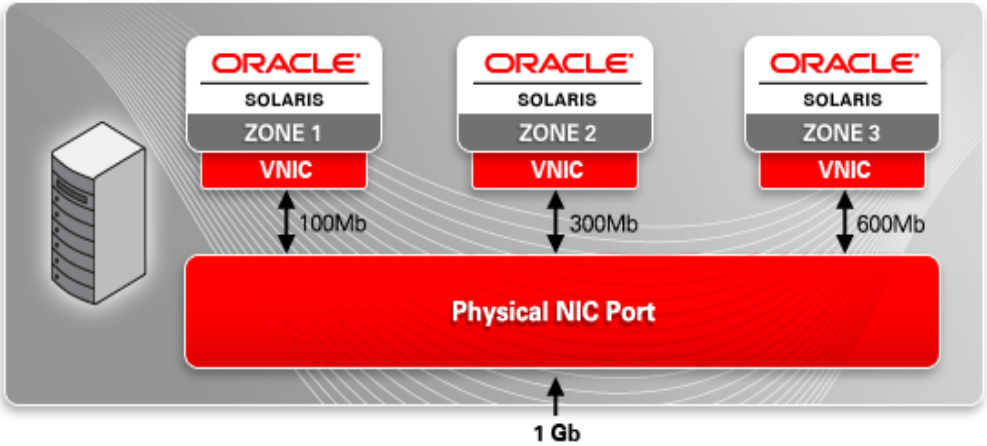
Monitoring and Accounting

```
# flowadm show-flow -s -i 1
FLOW IPACKETS RBYTES IERRORS OPACKETS OBYTES OERRORS
httpflow 278891 19754223 0 232390 29558178 0
httpflow 5551 393179 0 4626 588354 0
httpflow 5616 397800 0 4680 595296 0
# acctadm -e extended -f /var/log/net.log net
# flowadm show-usage -f /var/log/net.log
FLOW DURATION IPACKETS RBYTES OPACKETS OBYTES BANDWIDTH
httpflow 1620 513064 36337108 427407 54349626 0.447 Mbps
```

Solaris 11 Built-in Network Virtualization

- Virtual NICs and resource control
  - Set priority, bandwidth
  - Integrated with Solaris zones
  - Secure, zero overhead and observable





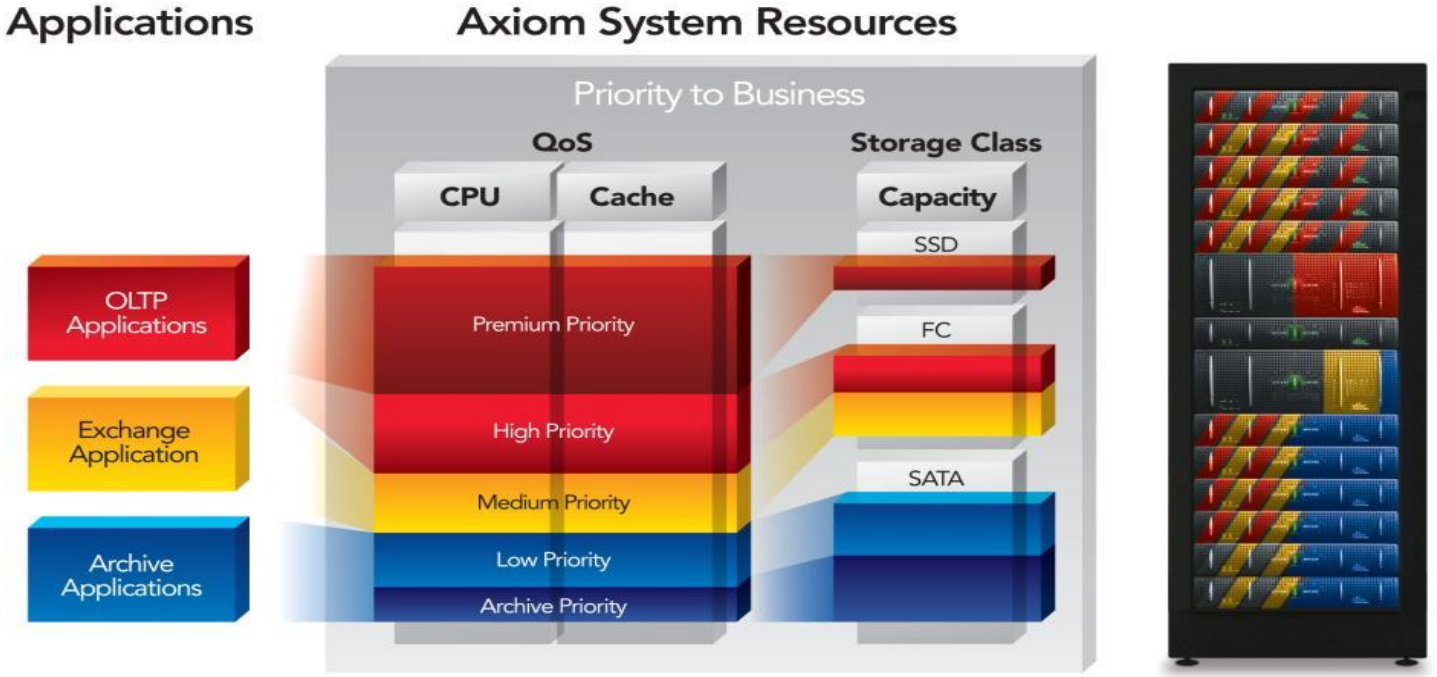
Solaris Zones

15x lower overhead vs. VMWare

4x lower latency vs. KVM

Storage Quality of Service

Oracle Pillar Axiom 600



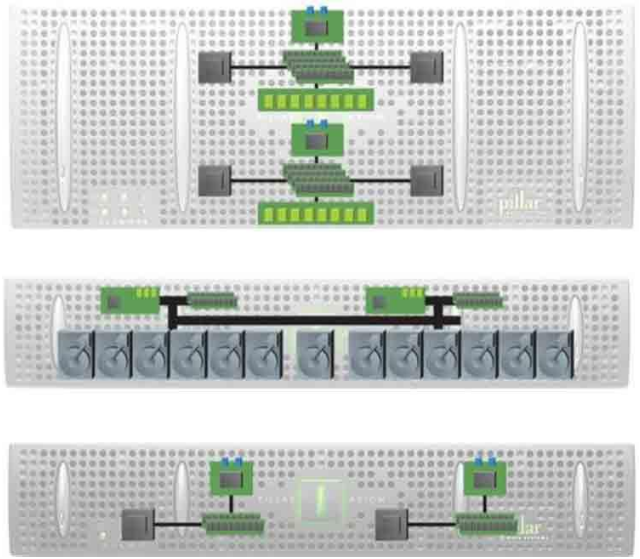
Sun Axiom 600 enables IT Administrators to utilize 80% of storage capacity without performance degradation

– twice the industry average

Source: Gartner Group, 2010

Axiom 600: Modular Architecture

Basic Building Blocks



Slammer – System Controller

- 2 Active-Active SAN or NAS control units per slammer
- 1 to 4 Slammers per Axiom

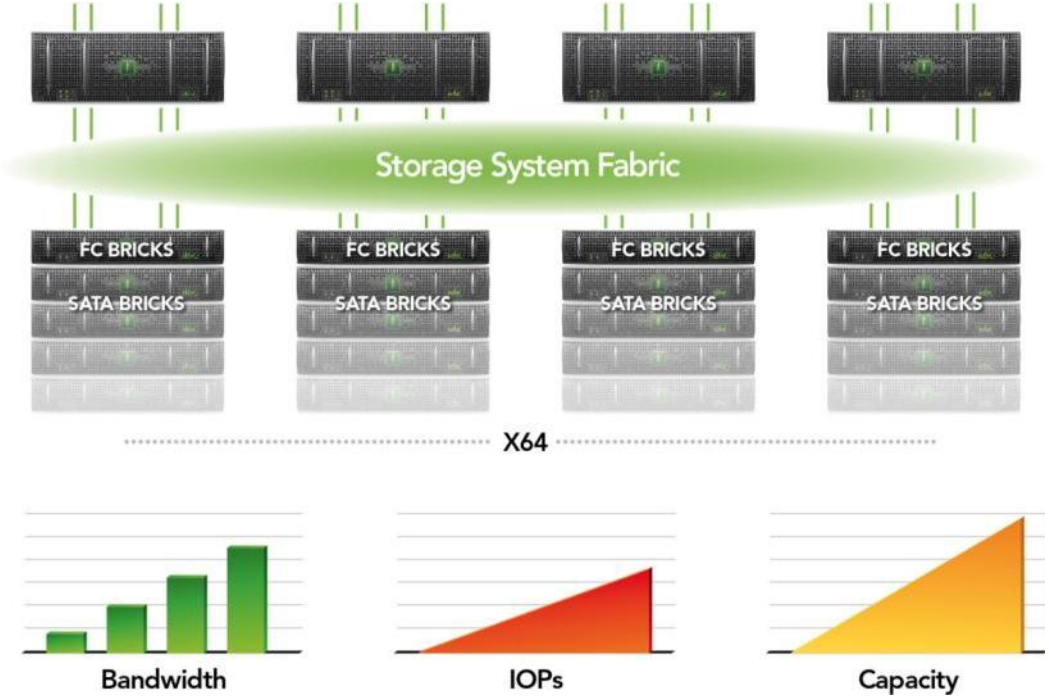
Brick – Drive Enclosure

- 12-13 drives + 2 RAID controllers per brick
- 1 to 64 Bricks per Axiom

Pilot – UI/Management Controller

- 2 Active-Standby Control Units
- 1 Pilot per Axiom

Scale-up and Scale-out Design



Linear scaling of bandwidth and performance by adding Slammers
Linear scaling of performance and capacity by adding Bricks

Oracle's SAN Storage

5th generation of Oracle's Pillar Axiom 600 Storage System

Axiom 600 – one model that linearly scales both capacity and performance

SINGLE ACTIVE-ACTIVE SLAMMER with ONE BRICK

2 Control Units
13 Drives
12TB Capacity
48GB Cache

Up to 4 ACTIVE-ACTIVE SLAMMERS with 64 Bricks

8 Control Units
Up to 832 drives
Up to 1.6PB Capacity
192GB Cache
128 RAID Controllers
SATA, FC, or SSD Storage Classes

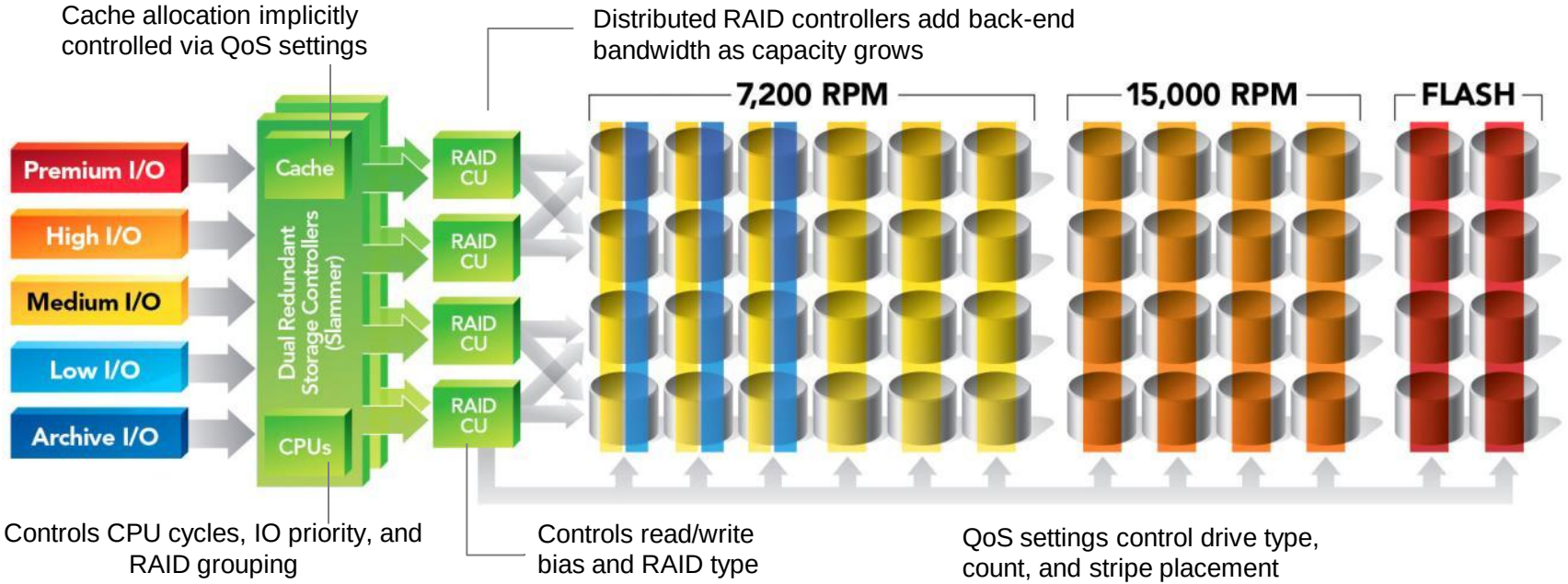
All models include…

- Patented Quality of Service (QoS) Software
- All Protocols: FC, iSCSI, CIFS, NFS
- All Management Software: Application Profiles, Thin Provisioning, Path Management, MaxMan
- Data Protection and Mobility Software: Copy Services, Snapshots, R/W Clones
- Engineered integration with Oracle software: EHCC, OEM, ASM, SAM, Oracle VM

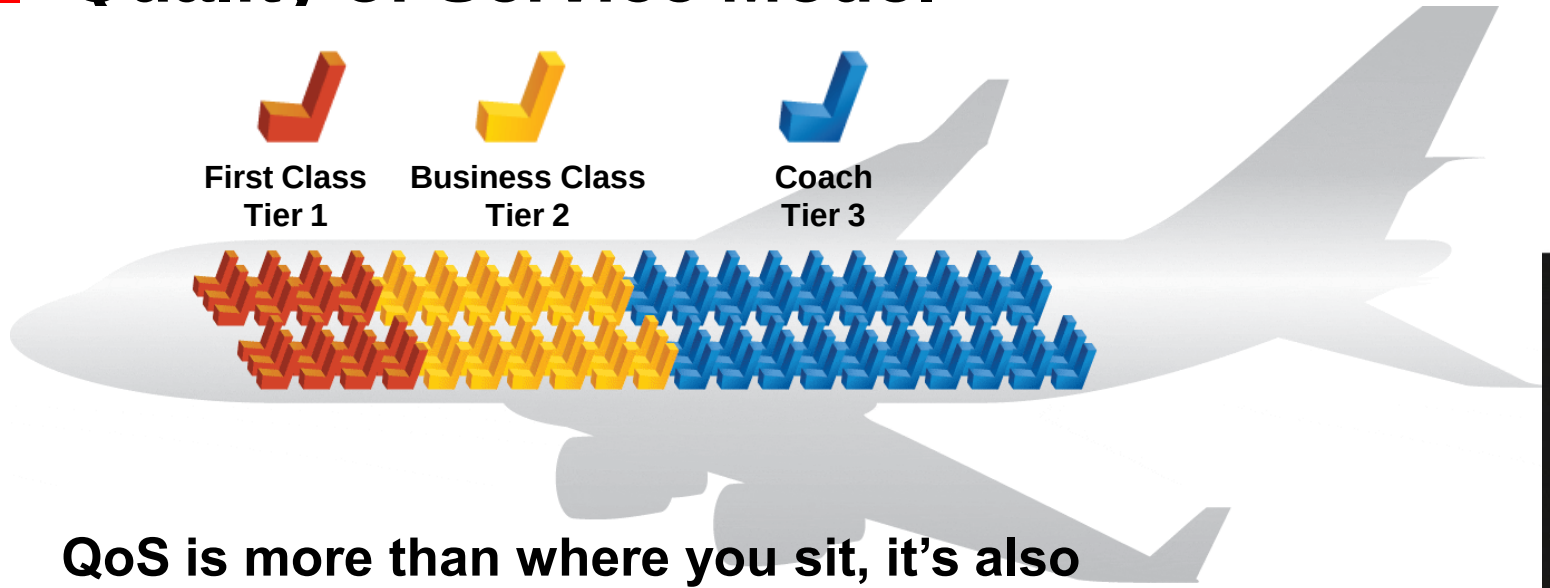
ORACLE

Pillar Axiom: Patented QoS

Deterministic IO Prioritization: The End of archaic FIFO queue management

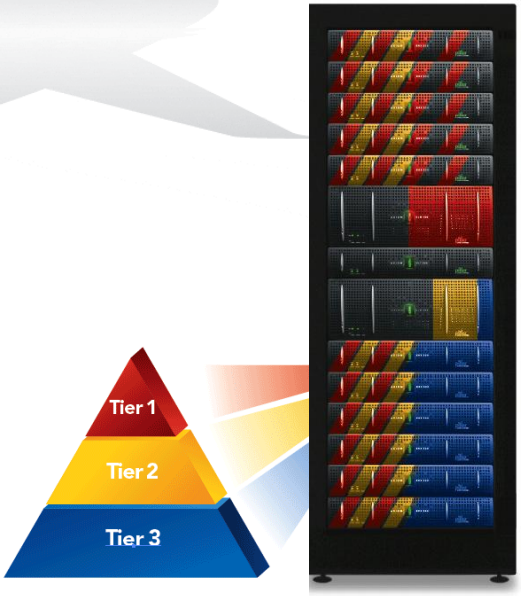


Quality of Service Model



QoS is more than where you sit, it's also the priority and class of service you get

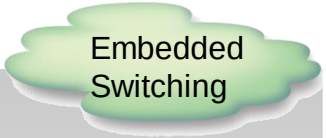
- How fast you board and exit
- The number of attendants per passenger
- The seat size and leg room
- The entertainment selections



System Wide Quality of Service

- Software
- Virtualization maps
- QoS policy management
- Predictive modeling

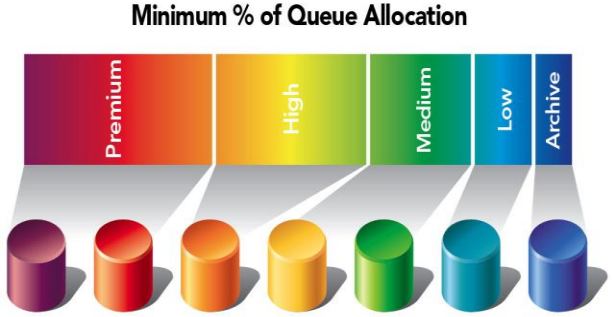
- I/O prioritization
- I/O allocation
- Cache prioritization
- Dynamic cache algorithms
- Slammer load balancing



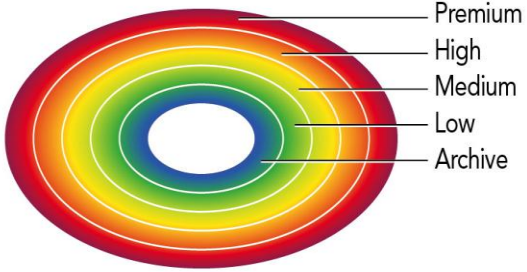
- Fabric bandwidth prioritization and allocation

- RAID configuration
- Mirroring
- Brick load balancing

- Disk access prioritization
- Data layout
- Disk block partitioning
- Disk block prioritization

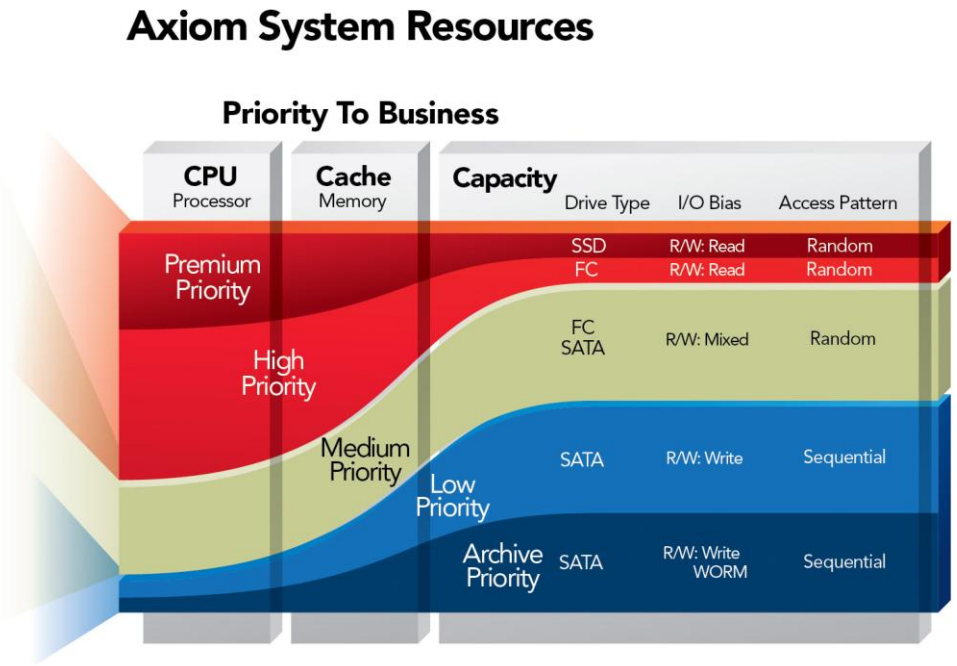


Logical Volumes



Data Layout and Block Prioritization Bands

Isolating Workloads with Quality of Service

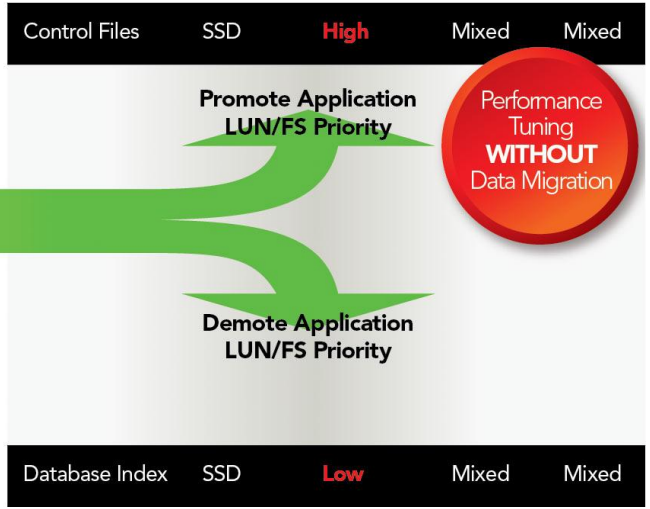


Reset the Quality of Service

Within the Same Storage Class

Manage Application Priority Temporarily or Permanently

Data Type	Storage Class	LUN Performance Profile		
		Priority	Access Bias	I/O Bias
Control Files	FC	High	Mixed	Mixed
Database Index	SSD	Medium	Mixed	Mixed
Database Tables	SATA	Medium	Mixed	Mixed
Temporary Files	SATA	Medium	Mixed	Mixed
Online Redo Log Files	FC	High	Sequential	Write
Archive Log Files	SATA	Low	Sequential	Write

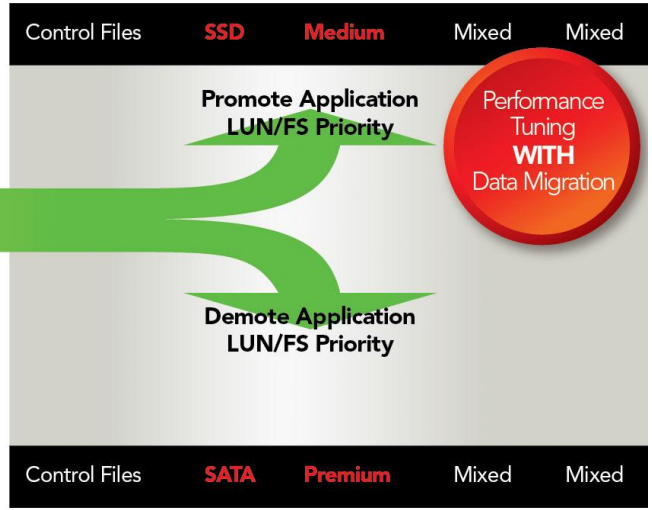


Reset the Quality of Service

Across Storage Classes

Manage Application Priority Permanently

Data Type	Storage Class	LUN Performance Profile		
		Priority	Access Bias	I/O Bias
Control Files	FC	High	Mixed	Mixed
Database Index	SSD	Medium	Mixed	Mixed
Database Tables	SATA	Medium	Mixed	Mixed
Temporary Files	SATA	Medium	Mixed	Mixed
Online Redo Log Files	FC	High	Sequential	Write
Archive Log Files	SATA	Low	Sequential	Write



Isolating Workloads with Storage Domains

Create Multiple Axioms Within a Pillar Axiom

Agenda

- Ressource-Management
- Aktuelle CPU-Features und ihre Abstraktion im OS
- Netzwerk-Virtualisierung und Bandbreiten-Management
- Storage Quality of Service

Weitere Informationen

- Franz Haberhauer: Ressource Management für und mit modernen Rechnerarchitekturen. Uptimes GUUG FFG 2012, S. 33-37
- Oracle‘s SPARC T4-1, SPARC T4-2, SPARC T4-4, and SPARC T4-1B Server Architecture, An Oracle White Paper http://www.oracle.com/technetwork/server-storage/sun-sparc-enterprise/documentation/o11-090-sparc-t4-arch-496245.pdf, Dynamic Threading S. 11-12, Critical Thread Optimization S. 22
- Tuning the SPARC CPU to Optimize Workload Performance on SPARC T4, An Oracle White Paper, September 2011, http://www.oracle.com/technetwork/server-storage/vm/ovm-sparc-t4-505241.pdf
- Oracle Solaris 11 Networking Virtualization Technology http://www.oracle.com/technetwork/server-storage/solaris11/technologies/networkvirtualization-312278.html
- Oracle Solaris 11 Administration: Network Interfaces and Network Virtualization – Chapter 21 Managing Network Resources http://docs.oracle.com/cd/E23824_01/html/821-1458/gfkbr.html
- Delivering Quality of Service with Pillar Axiom 600, An Oracle White Paper, September 2011, http://www.oracle.com/us/products/servers-storage/storage/san/improving-shared-storage-wp-488796.pdf

Q&A

Hardware and Software

ORACLE

Engineered to Work Together


ORACLE®