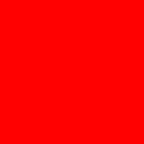




# ORACLE®

## Optimizing Oracle Database on Sun SPARC Enterprise M-series Servers

Ravindra Talashikar, Sun Microsystems  
Glenn Fawcett, Oracle Corporation  
Allan Packer, Oracle Corporation



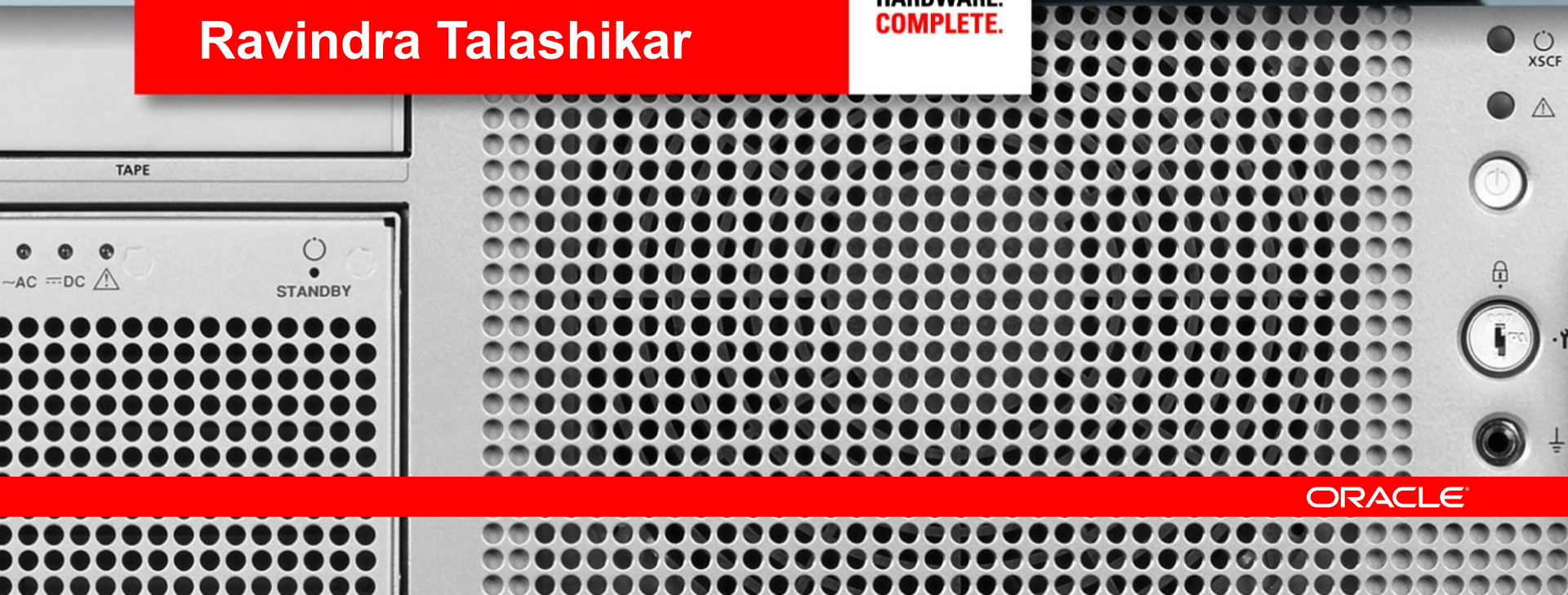
The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development,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



# M-Series Intro

Ravindra Talashikar

SOFTWARE.  
HARDWARE.  
COMPLETE.



# Outline

- Enterprise System requirements and optimal Oracle DB performance: What matters?
- SPARC64 VII and Sun SPARC Enterprise M-series server architecture
- M-series products : An Enterprise class range of servers for Oracle Database
- Oracle Solaris for M-series : An Enterprise OS for an Enterprise class server range

# Enterprise System Architecture: What matters for Oracle Database?

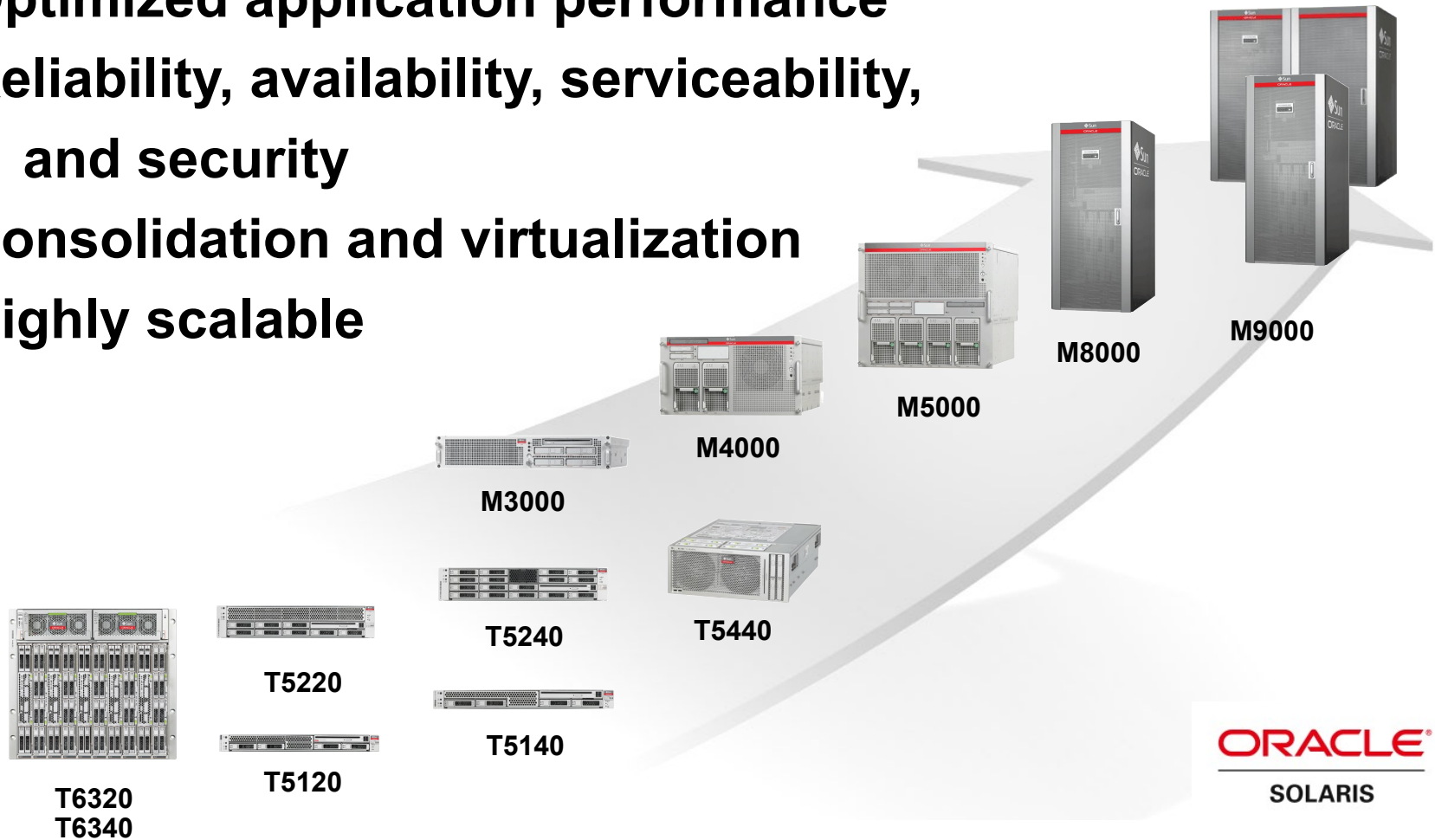
- Need balanced CPU / Memory / I/O capacity
- Need scalable architecture
- Need mainframe-class manageability for Data Center ready systems
- Need Resource Management (Server Domaining) features for DB consolidation
- Lower memory latency differences
- High and scalable memory bandwidth
- Need configuration flexibility: mix & match CPUs
- Need state of the art RAS features

**M-series is designed for these requirements**

# SPARC Enterprise Servers

Over 20 Years Of Mission Critical Computing

- Optimized application performance
- Reliability, availability, serviceability, and security
- Consolidation and virtualization
- Highly scalable



**ORACLE**  
SOLARIS

**ORACLE**

# SPARC64 VI and SPARC64 VII

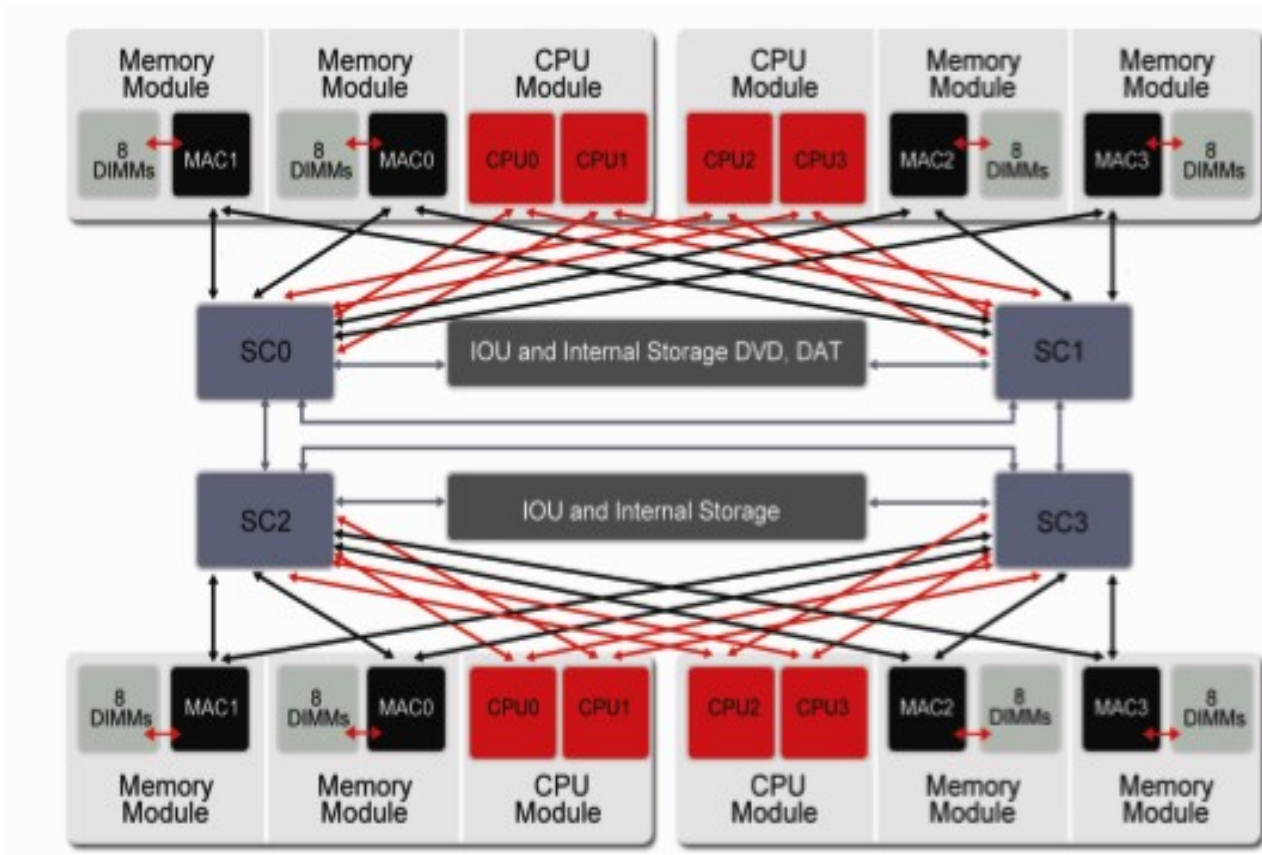
- SPARC64 VI and SPARC64 VII: Fundamental building blocks of M-series servers

	SPARC64 VI	SPARC64 VII
<b>Max Frequency</b>	<b>2.4 GHz</b>	<b>2.88 GHz</b>
<b>Architecture (Cores / strands)</b>	<b>Dual Core, Two threads/core</b>	<b>Quad Core. Two threads/core</b>
<b>Technology</b>	<b>90 nm, sun 4u</b>	<b>65 nm, sun 4u</b>
<b>Power consumption</b>	<b>150 Watts (max)</b>	<b>150 Watts (max)</b>
<b>Compatibility</b>	<b>SPARC V7, V8, V9</b>	<b>SPARC V7, V8, V9</b>
<b>Multithreading Technique</b>	<b>VMT</b>	<b>SMT</b>

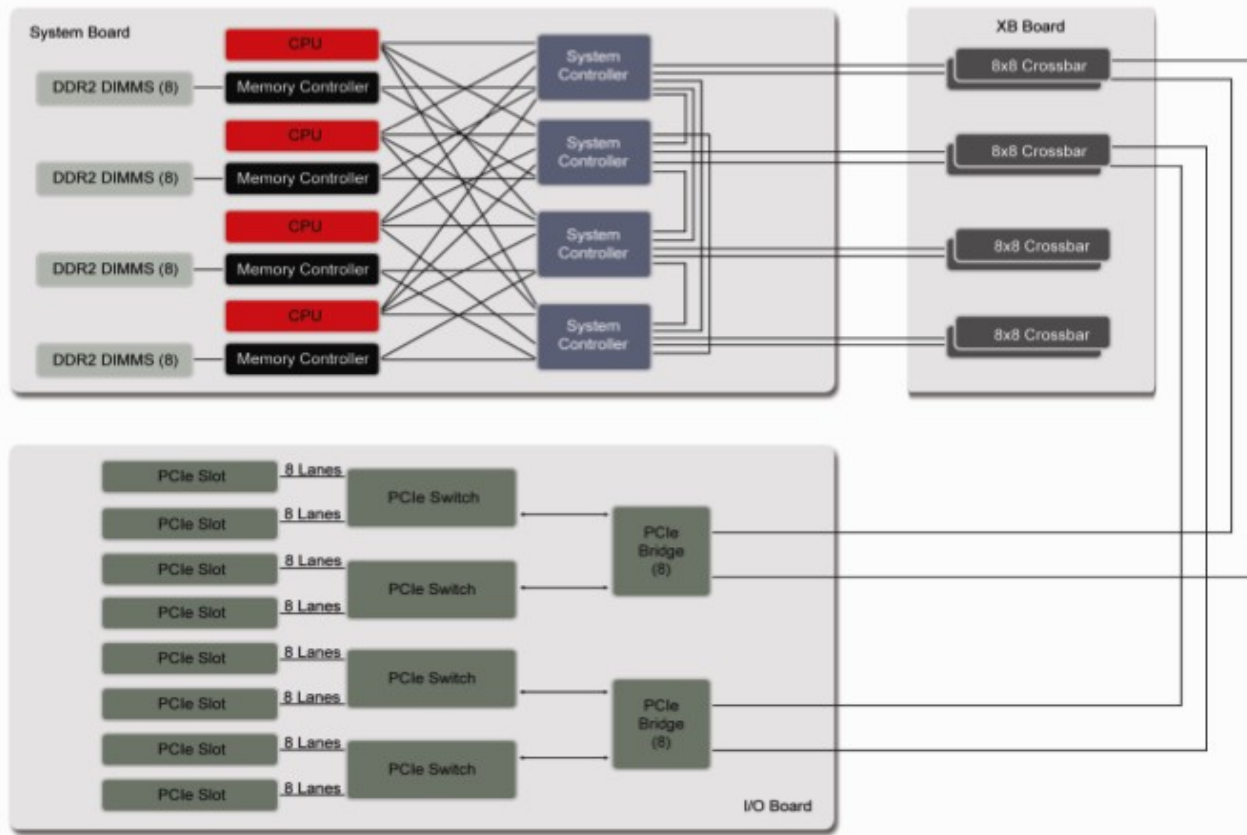
# Sun SPARC Enterprise M-series: At a Glance

	Space	Mixed CPUs (SPARC64 VI & VII)	Max Processors	Max Memory	External I/O	Dynamic Domains
<b>M3000</b>	2 RU	NO	1	64 GB	1 Port	1
<b>M4000</b>	6 RU	YES	4	256 GB	2 Units	2
<b>M5000</b>	10 RU	YES	8	512 GB	4 Units	4
<b>M8000</b>	1 Cabinet	YES	16	1 TB	8 Units	16
<b>M9000</b>	1 or 2 Cabinets	YES	32 or 64	2 or 4 TB	16 Units	24

# M5000 : Mid Range Server Architecture

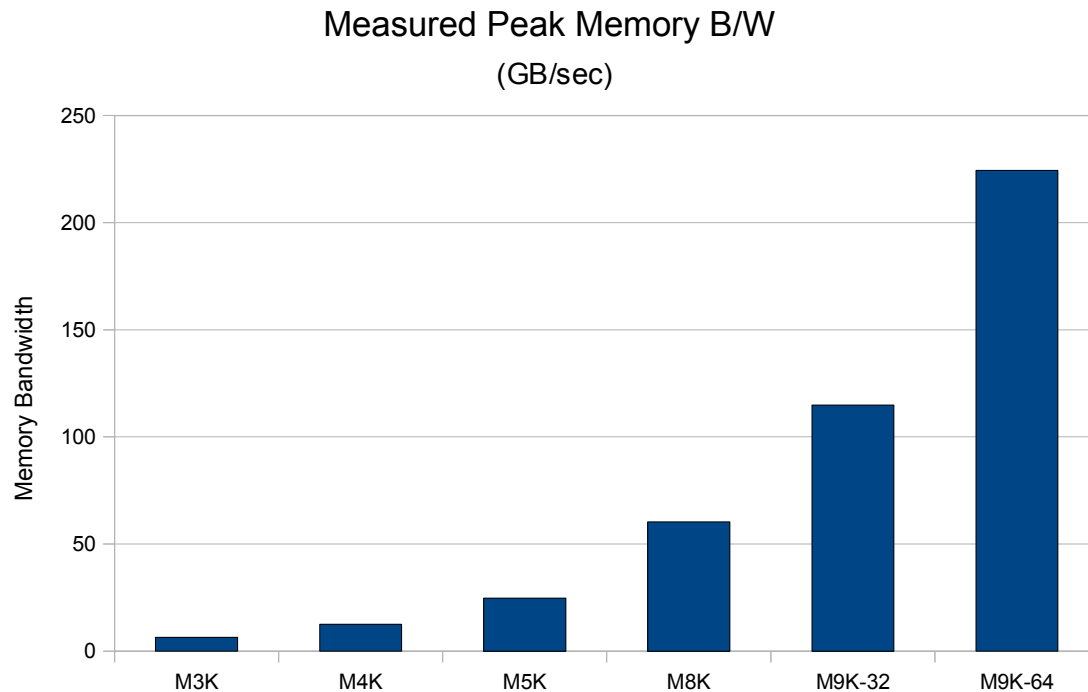


# M8000 – M9000 : Basic building block



# M-series : Memory subsystem for DB performance

- DB workloads are memory heavy and extremely sensitive to memory subsystem performance
- You need to feed CPUs if you want to keep them busy



# Oracle Solaris optimizations for DB performance on M-series

## Key Solaris improvements

- Core-aware scheduling: Better system utilization
- Memory Placement Optimization: Benefits from locality
- Scalability improvements: Benefit from scaling up!
  - Improved concurrency while using mutexes
  - Scalable clock tick processing
  - Improved scalability for call-out event handling : Faster synchronization for Oracle processes
- For high end M-series servers, Oracle recommends **Oracle Solaris 10 10/09 release (S10 Update 8) or later**

A man in a dark suit, light blue shirt, and striped tie is sitting in a black office chair. He is gesturing with his right hand as if speaking. Behind him are several rows of server racks. The racks have a perforated metal front panel. On the left, a rack has a 'TAPE' label and a 'STANDBY' indicator. On the right, a rack has a power button, a warning triangle, a lock icon, and a power jack.

# Effective Deployment

## Glenn Fawcett

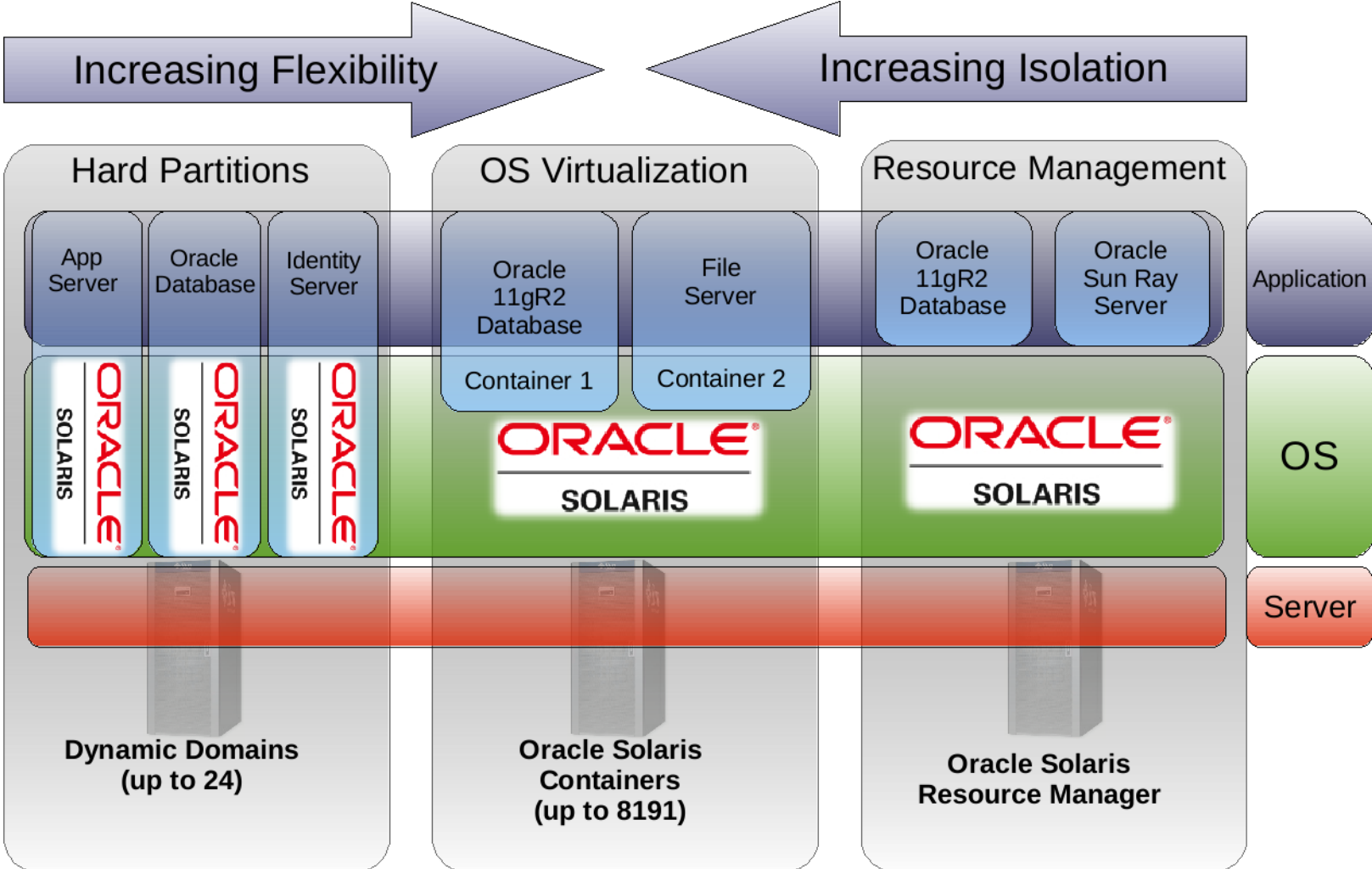
**SOFTWARE.  
HARDWARE.  
COMPLETE.**

# Overview

- **M-series Virtualization**
  - Dynamic System Domains
  - Oracle Solaris Containers
  - DBRM w/ Instance Caging
- **Configuring Oracle for M-series servers**
  - Memory Considerations
  - Parallelism
  - Concurrency



# The Power of Virtualization



# M-series Virtualization

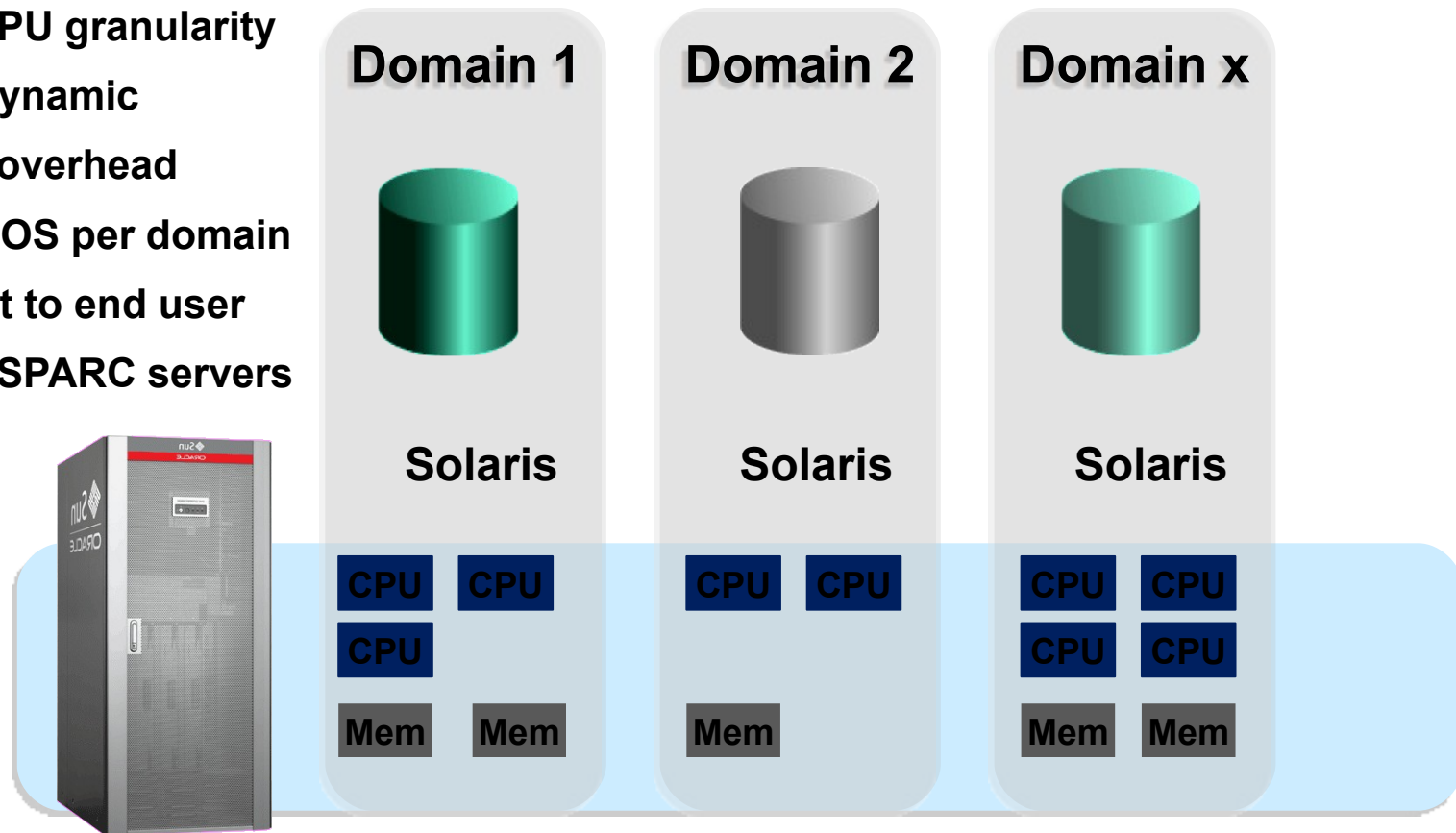
Choosing the right level for the application

- **Dynamic System Domains**
  - Looks like a standalone server
  - Most secure
  - Hard partitioning of resources
- **Oracle Solaris Containers**
  - Shared Global OS
  - Easy provisioning
  - Mobility of OS instances supported
  - Consolidate multiple OS versions with branded containers
- **Mix and Match**
  - Solaris Containers within a Domain

# Sun Dynamic System Domains

## Hardware Partitions

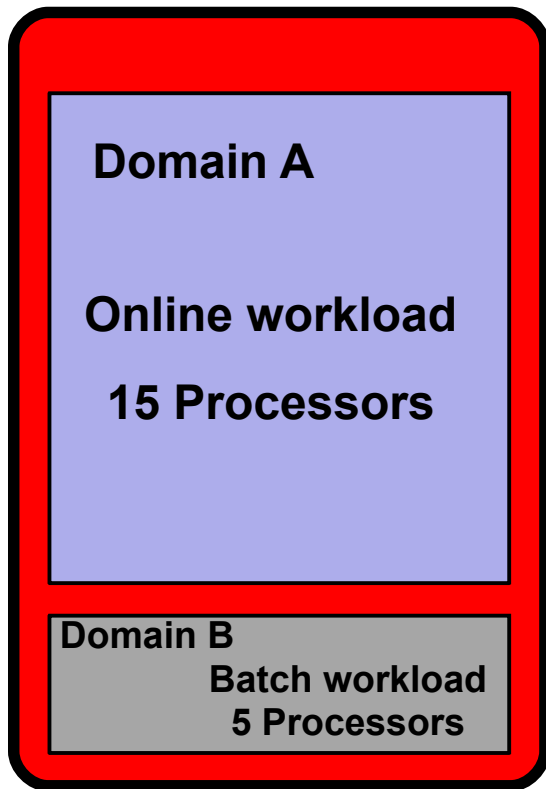
- Complete isolation
- Resource, security, service, fault
- Single-CPU granularity
  - Dynamic
  - No overhead
- Separate OS per domain
  - No cost to end user
- M-Series SPARC servers



# Domain Resizing

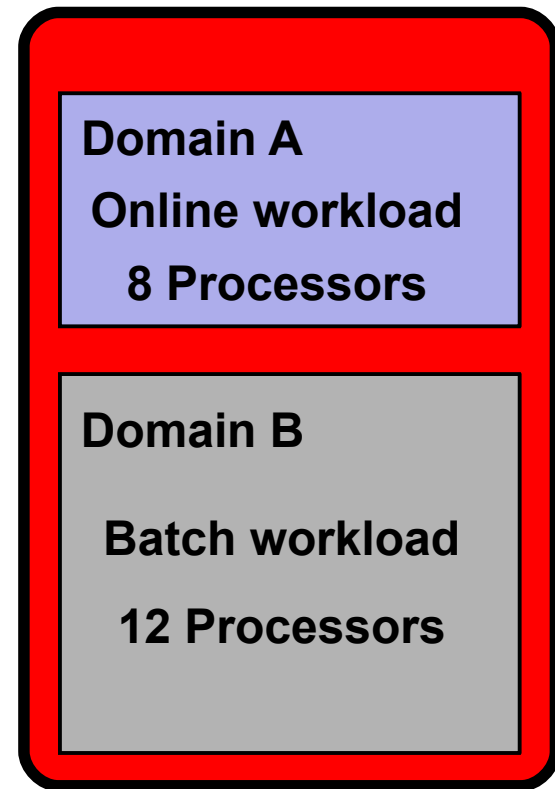
Better Resource Utilization

## Daytime Configuration



**M9000**

## Night time Configuration



**M9000**

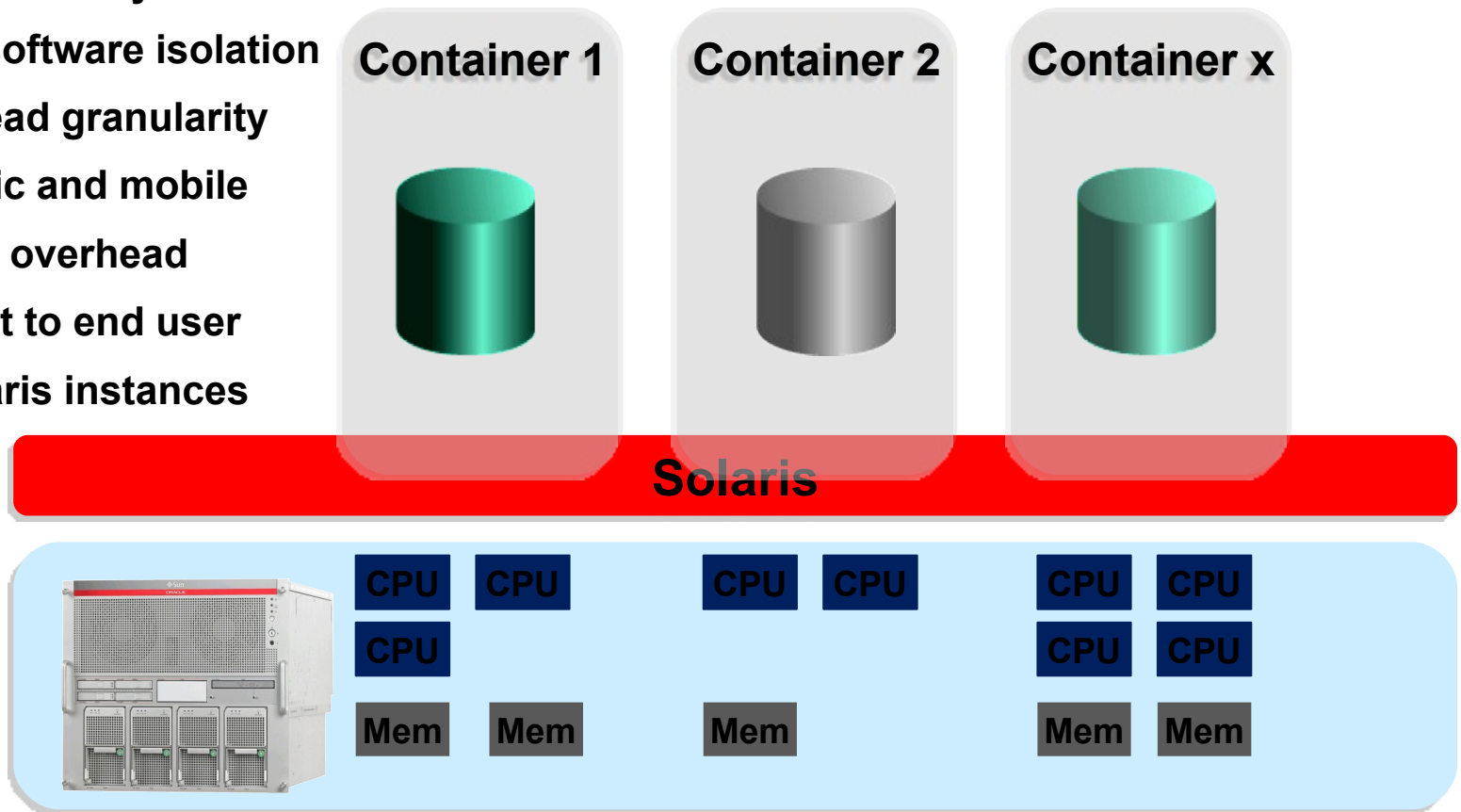
# Solaris Containers

Built-in Virtualization on Any Solaris System

- **Limitless partitioning—one license**
- **Highly efficient consolidation tool**
  - Thousands of applications on one system
- **Container cloning, migration**
- **Instant restart**
- **Ideal for many scenarios**
  - Highly secure isolation
  - Lightweight “test” environments
  - Dynamic environments with resource sharing
  - High performance, especially multithreading
  - Rapid prototyping testbeds on same hardware and OS

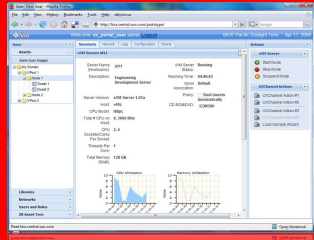
# Oracle Solaris Containers

- One OS instance for all containers
  - Separate file system
- Complete software isolation
  - Sub-thread granularity
  - Dynamic and mobile
    - Low overhead
  - No cost to end user
- All Solaris instances

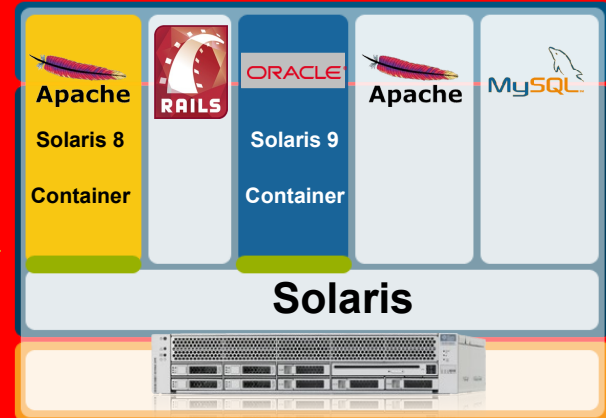
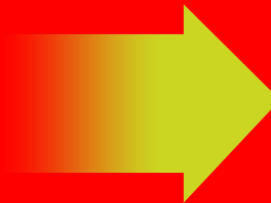


# Solaris 8 and Solaris 9 Containers

## Consolidate Legacy Applications



### Oracle Enterprise Manager Ops Center



- **Rapid conversion from older environments**
- **Simplifies move to newer, more efficient SPARC systems**
- **Lower power/cooling/space/support costs**
- **Physical-to-Virtual (P2V) consolidation**

# Overview

- **M-series Virtualization**
  - Dynamic System Domains
  - Oracle Solaris Containers
  - DBRM w/ Instance Caging
- **Configuring Oracle for M-series servers**
  - Memory Considerations
  - Parallelism
  - Concurrency



# Configuring Oracle for M-series

## Memory considerations

Large memory densities possible with a single system image

“In-Memory” parallel query with 11gR2 becomes interesting

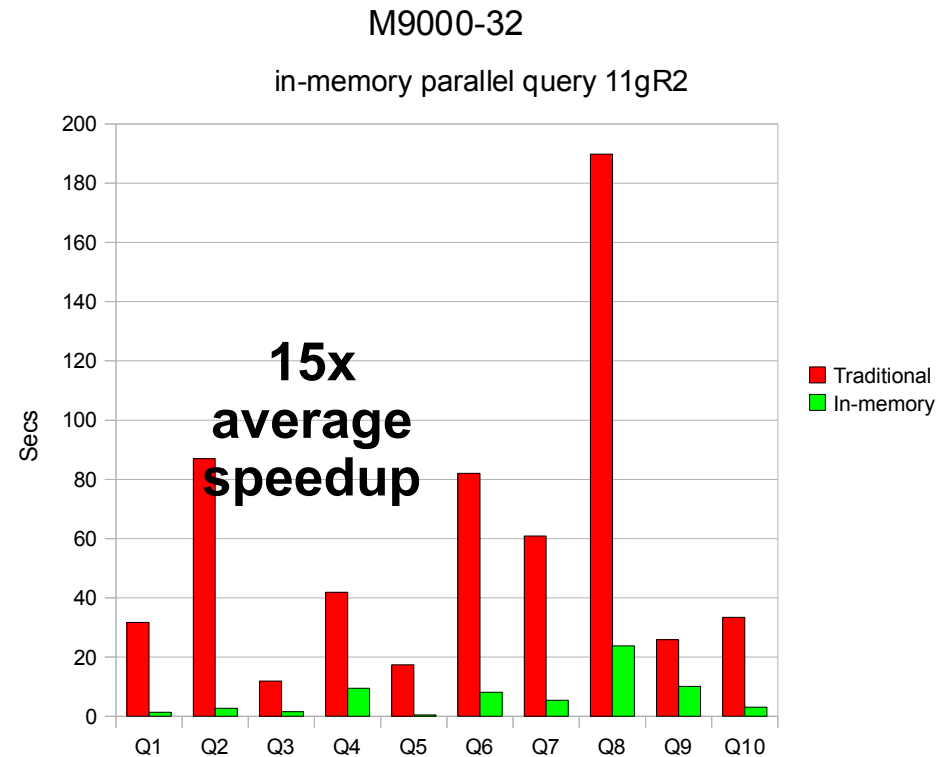
## Dynamic ISM for SGA

NUMA optimized

Grow/Shrink memory dynamically

Enabled with “sga\_max\_size”

<i>system</i>	<i>Memory</i>
M9000-64	4TB
M9000-32	2TB
M8000	1TB
M5000	512GB



# Configuring Oracle for M-series

Sockets / Cores / Threads

- **SPARC64 VII+ processor**
  - 4 cores / 8 threads each
- **Huge number of threads in a single system image**
- **Support high-level of parallelism and concurrency**
- **“cpu\_count = #threads!!”**

<i>system</i>	<i>sockets</i>	<i>cores</i>	<i>threads</i>
M9000-64	64	256	512
M9000-32	32	128	256
M8000	16	64	128
M5000	8	32	64
M4000	4	16	32
M3000	1	4	8

# Configuring Oracle for M-series

## DSS considerations

### Use large pages for the PGA

8K is default... 4M is recommended

```
_realfree_heap_pagesize_hint=4194304
```

### Increase message size for parallel query

Default is too low "1500"

```
parallel_execution_message_size = 16384
```

### Parallel query

Query slaves = 2x number of threads

10gR2 : Parallelism set on per table basis

11gR2 : Auto parallelism will select based on resources...  
queueing can help as well.

# Configuring Oracle for M-series

## Parallelism across the board

### Data Pump ( impdp / expdp )

```
expdp system/manager directory=my_dir full=n schemas=gcbc  
dumpfile=exp%u.dmp logfile=expdp.log parallel=4
```

### Index builds / rebuilds

```
SQL> create index abc_idx on abc(c1)  
unrecoverable parallel 8;
```

### Stats Collection

#### 10gR2 :

```
SQL> DBMS_STATS.GATHER_SCHEMA_STATS  
(OWNNAME=>'scott',ESTIMATE_PERCENT=>$2,DEGREE=>32);
```

#### 11gR2 : use defaults.

### RMAN backup

Parallelism via multiple channels... especially when  
RMAN compression is in-use.

# Configuring Oracle for M-series

Concurrency : Connection management.

Connection pooling

App tier persistent connections best

Connection pooling within the DB

Faulty application logic causes connection storms

- Minimize dynamic connections
- Set max=min connections
- Limit overall number of connections.

For CPU bound case

(#connections = #threads)



# Best Practices

Allan Packer

SOFTWARE.  
HARDWARE.  
COMPLETE.



ORACLE

# Overview

- **M-series Best Practices**
  - Oracle Database Smart Flash Cache
  - DISM
  - Project-based Tunables
  - Resource Management
  - NUMA
  - RAC

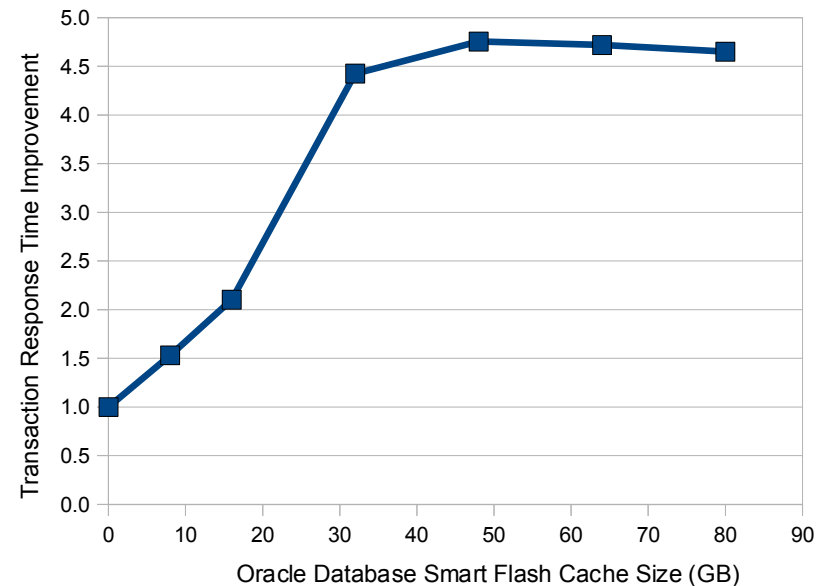
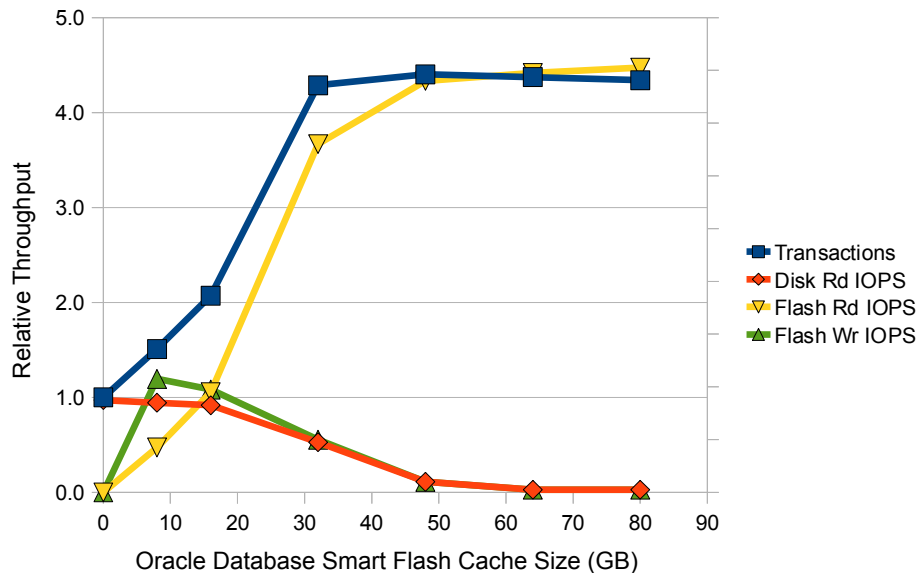


# Oracle Database Smart Flash Cache

Extends the SGA beyond Main Memory to Flash Cache

- **Available from Oracle Database 11gR2**

- Supported on Solaris and Oracle Enterprise Linux
- Increases transaction throughput and reduces response times
- Best suited to read-mostly and read-only workloads
- Sun Storage F5100 Flash Array or Sun Flash Accelerator F20 PCIe Card
- *db\_flash\_cache\_file = "+dg1/lffile\_asm"*
- *db\_flash\_cache\_size = 50G*



# Oracle Solaris on M-series

## Project-based Tunables

- **Solaris 9 and earlier**

- Changes were needed in `/etc/system`, for example:
  - `set semsys:seminfo_semmni=100`  
`set semsys:seminfo_semmsl=256`  
`set shmsys:shminfo_shmmax=17179869184`  
`set shmsys:shminfo_shmmni=100`
- Only took effect after a system reboot

- **Solaris 10 and later**

- Project-based tunables
  - `# projadd oracle`  
`# echo oracle::::project=oracle>> /etc/user_attr`  
`# prctl -n project.max-shm-memory -v 16gb -r -i project`  
`oracle`
- Running Oracle Database in an Oracle Solaris Container
  - Set `zone.max-shm-memory` for the Container
- Changes can be made dynamically

# Oracle Solaris on M-series

## DISM

- **Dynamic Intimate Shared Memory (DISM) supports dynamic tuning of the SGA size**
  - Dynamic tuning can be automatic or manual
  - **DISM is similar to ISM**
    - Kernel virtual to physical memory address translations are shared between processes attaching to same shared mem
    - Large pages supported (from 4MB to 256MB instead of 8KB)
  - **But...**
    - Requires swap space, unlike ISM
    - Shared memory is not locked automatically
      - Locking/unlocking done by ora\_dism process
  - Only supported on SPARC-based systems

# Oracle Solaris on M-series

## DISM

- **Best Practice – turn DISM off unless you need it**
  - Significant performance degradation if DISM not configured correctly
  - Refer to whitepaper for details of correct configuration

# Oracle Solaris on M-series

## Resource Management

- **Instance Caging**
  - Limit the amount of CPU resource consumed by an instance by setting `CPU_COUNT` in `init.ora`
  - Be aware that the second thread per core is seen as a CPU and can cause unexpected effects
- **Oracle Solaris Resource Manager (SRM)**
  - Includes pools, psets, scheduling classes, per-project CPU resource shares,...
  - Be careful before using in combination with Oracle Database Resource Manager (DRM)
    - DRM applies only to a single database instance
    - SRM applies to a single Solaris instance
    - Neither understands the other

# Oracle Solaris on M-series

## Best Practices – General

- **NUMA**

- From Oracle Solaris 9, Oracle Database's NUMA features can be used with any Sun system with NUMA characteristics
- Turned off by default in 11.2.0.1, but supported if turned on
- Enable it with an `init.ora` parameter
  - From 11.2.0.1: `_enable_NUMA_support=TRUE`
  - Earlier versions: `_enable_NUMA_optimization=TRUE`

# Oracle Solaris on M-series

## Best Practices – General

- **Storing Data on M-Series for Oracle Database**
  - Which to use?
    - **ASM vs ZFS vs UFS Direct I/O vs VxFS,...**
  - **Best Practice: Use ASM**
- **Accelerating I/O**
  - **Use Sun Storage F5100 Flash Array or Sun Flash Accelerator F20 PCIe Card for DB Smart Flash Cache**
  - **Use F5100 with COMSTAR for general purpose accelerated I/O**

# Oracle Solaris on M-series

## Best Practices – RAC

- **Jumbo Frames**

- A recommended Best Practice
- Can boost throughput by around 20%
- Cf. <http://www.sun.com/products/networking/ethernet/jumbo/>

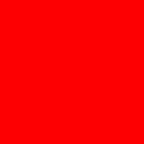
- **LMS Tuning**

- Need 4 LMS processes to saturate 1 Gbit NIC using 8K blocks (untuned)
- With 10 Gbit NIC, can achieve 3x throughput with 3x LMS processes
- Placing LMS processes in a processor set and fencing interrupts can boost throughput 40%
- Latency can improve by about 30% by enabling UDP checksum offload, disabling RX soft rings, and disabling interrupt blanking

# In Conclusion

- **M-series Best Practices**
  - **Oracle Database Smart Flash Cache**
    - **Increase SGA beyond main memory**
  - **DISM**
    - **For dynamic SGA memory tuning**
    - **Configure it correctly, or turn it off!**
  - **Project-based Tunables**
    - **Do it dynamically!**
  - **Resource Management**
    - **Use Solaris features along with DRM**
  - **NUMA**
    - **Need to turn it on (not on by default)**
  - **RAC**
    - **Tuning can make a difference**





The preceding 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 Products Available Online



## Oracle Store

SHOP NOW

Buy Oracle license and support  
online today at  
[oracle.com/store](https://www.oracle.com/store)