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
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
### SPARC64 VI and SPARC64 VII

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

<table>
<thead>
<tr>
<th>Feature</th>
<th>SPARC64 VI</th>
<th>SPARC64 VII</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Frequency</td>
<td>2.4 GHz</td>
<td>2.88 GHz</td>
</tr>
<tr>
<td>Architecture (Cores / strands)</td>
<td>Dual Core, Two threads/core</td>
<td>Quad Core, Two threads/core</td>
</tr>
<tr>
<td>Technology</td>
<td>90 nm, sun 4u</td>
<td>65 nm, sun 4u</td>
</tr>
<tr>
<td>Power consumption</td>
<td>150 Watts (max)</td>
<td>150 Watts (max)</td>
</tr>
<tr>
<td>Compatibility</td>
<td>SPARC V7, V8, V9 VMT</td>
<td>SPARC V7, V8, V9 SMT</td>
</tr>
<tr>
<td>Multithreading Technique</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Sun SPARC Enterprise M-series: At a Glance

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

Measured Peak Memory B/W

(GB/sec)
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
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

Increasing Flexibility

Hard Partitions
- App Server
- Oracle Database
- Identity Server

Dynamic Domains (up to 24)

Oracle Solaris Containers (up to 8191)

Increasing Isolation

OS Virtualization
- Oracle 11gR2 Database
- File Server

Container 1

Container 2

Resource Management
- Oracle 11gR2 Database
- Oracle Sun Ray Server

Application

OS

Server

Oracle Solaris Resource Manager
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

Domain A
Online workload
15 Processors

Domain B
Batch workload
5 Processors

Night time Configuration

M9000

Domain A
Online workload
8 Processors

Domain B
Batch workload
12 Processors
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

- 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”

<table>
<thead>
<tr>
<th>system</th>
<th>Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>M9000-64</td>
<td>4TB</td>
</tr>
<tr>
<td>M9000-32</td>
<td>2TB</td>
</tr>
<tr>
<td>M8000</td>
<td>1TB</td>
</tr>
<tr>
<td>M5000</td>
<td>512GB</td>
</tr>
</tbody>
</table>

15x average speedup
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!!”

<table>
<thead>
<tr>
<th>system</th>
<th>sockets</th>
<th>cores</th>
<th>threads</th>
</tr>
</thead>
<tbody>
<tr>
<td>M9000-64</td>
<td>64</td>
<td>256</td>
<td>512</td>
</tr>
<tr>
<td>M9000-32</td>
<td>32</td>
<td>128</td>
<td>256</td>
</tr>
<tr>
<td>M8000</td>
<td>16</td>
<td>64</td>
<td>128</td>
</tr>
<tr>
<td>M5000</td>
<td>8</td>
<td>32</td>
<td>64</td>
</tr>
<tr>
<td>M4000</td>
<td>4</td>
<td>16</td>
<td>32</td>
</tr>
<tr>
<td>M3000</td>
<td>1</td>
<td>4</td>
<td>8</td>
</tr>
</tbody>
</table>
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
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`

![Graph of Oracle Database Smart Flash Cache Size (GB) vs. Transactions, Disk Rd IOPS, Flash Rd IOPS, Flash Wr IOPS]

![Graph of Oracle Database Smart Flash Cache Size (GB) vs. Transaction Response Time Improvement]
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
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
• **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%

• 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

Buy Oracle license and support online today at oracle.com/store