ORACLE
SOLARIS

Oracle Solaris 11—The Optimal Platform for Deploying Oracle Database

ORACLE WHITE PAPER | SEPTEMBER 2014

ORACLE

Table of Contents

Introduction

In more than 310,000 companies around the world, IT organizations deploy Oracle Database to support critical business processes such as order entry, financials, human resources, customer management, and logistics. Since Oracle Database implementations are essential to these operations, the database tier must meet rigorous requirements for performance, scalability, availability, and security. Engineered by Oracle to meet demanding mission-critical requirements, Oracle Solaris 11 is a state-of-the-art platform for deploying Oracle Database instances.

Oracle Solaris is tested and optimized for Oracle Database, Oracle Applications, and Oracle hardware products, including servers based on Oracle's SPARC T5, M5, and M6 processors. Oracle SuperCluster, an Oracle engineered system that features an extensively validated and integrated hardware and software stack, incorporates Oracle Solaris 11 to realize exceptional levels of performance, scalability, security, and reliability. Oracle invests heavily in the ongoing integration of Oracle Database on Oracle Solaris, striving to optimize all aspects of database deployments on the operating system.

This paper highlights how Oracle Solaris 11 brings distinctive benefits to Oracle Database deployments, including deployments of the latest release, Oracle Database 12*c*. It describes operating system internals and enhancements that help to improve database scalability, availability, security, and manageability, and how Oracle Solaris 11 and Oracle Database optimizations bring specific benefits for database deployments on SPARC servers. Although both single-instance Oracle Database and Oracle Real Applications Cluster (Oracle RAC) deployments can take advantage of Oracle Solaris features, this paper focuses largely on optimizations relative to single-instance deployments. This paper covers guidelines and best practices for consolidating databases on Oracle Solaris, including the use of Oracle Multitenant—a new option for database consolidation in Oracle Database 12*c*.

Oracle Solaris 11—The Optimal Oracle Database Platform

The Oracle Solaris 11 operating system includes features that enhance scalability, availability, and manageability to unparalleled levels. Key features include Oracle Solaris Zones for built-in virtualization, Predictive Self Healing for continuous availability, DTrace for advanced observability, ZFS for next-generation volume and file system management, and advanced user and process rights for enhanced security. The Oracle Solaris 11.2 release integrates a full OpenStack distribution and adds features such as Kernel Zones and Unified Archives. (More details about specific features are available at www.oracle.com/solaris.) Oracle Solaris Cluster extends Oracle Solaris deployments, providing support for mission-critical database workloads that require high availability.

Oracle has made significant investments to optimize Oracle Database workloads on Oracle Solaris 11. The Oracle Solaris and Oracle Database engineering teams work closely together to determine how operating system features can complement and enhance database deployments. Specifically, their joint integration and optimization efforts focus on these areas:

- **Improving performance and scalability.** With the goal of optimizing out-of-box performance for Oracle Database on Oracle Solaris, Oracle invests heavily in integration testing, performance analysis, and software optimizations. These investments yield leading performance and scalability results for online transaction processing (OLTP) and decision support system (DSS) workloads, both for production environments and when executing common industry benchmarks.
- **Simplifying implementation, troubleshooting, and management.** Oracle has developed a single Oracle Solaris software package that installs all the packages needed to deploy Oracle Database. Oracle also provides ORAchk (which replaces RACcheck) to proactively diagnose and report configuration problems on Oracle Database deployments. In addition, Oracle Database administrators can safely observe what's happening in real time on live production systems because of the integration of Oracle Database and Oracle Solaris DTrace.
- **Improving resource management and utilization.** Oracle Database and Oracle Solaris integration efforts focus on making efficient use of system resources, including compute, memory, and I/O. Oracle's SPARC processor-based servers are available with large numbers of cores and huge memory capacities, making these platforms ideal for consolidating resource-intensive database workloads. Oracle Solaris includes features that help to allocate and manage these resources effectively. Oracle Database takes advantage of operating system features, such as Oracle Solaris resource pools, to allocate resources and control their use. By setting an Oracle Database parameter in `init.ora`, database processes can be bound to a processor group associated with an Oracle Solaris resource pool.
- **Speeding time to database use and increasing service levels.** Optimizations in Oracle Database 12*c* and Oracle Solaris help to accelerate database startup. Oracle Solaris 11 features a new multithreaded kernel process ("vmtasks") that parallelizes the creation of shared memory. Oracle Database 12*c* also uses Optimized Shared Memory in Oracle Solaris, which allows shared memory to be resized without a database service outage. These optimizations allow databases to start quickly, even when deployed with large, terabyte-sized shared memories. The availability of large memories in SPARC servers and the ability of Oracle Database to initialize large memories quickly makes it possible to deploy some databases entirely in memory.
- **Enabling flexible deployment options and compatible approaches to consolidating Oracle Database**s**.** Oracle offers both database and server-based approaches to consolidation, including Oracle Multitenant and no-cost virtualization technologies in Oracle Solaris and Oracle's SPARC servers. These approaches isolate database workloads running within a single physical machine. Integration efforts between the Oracle Solaris and Oracle Database engineering teams make it possible to consolidate databases on large NUMA machines while maintaining resource affinity.

The remainder of this paper provides greater detail about these optimizations and additional details that explain why Oracle Solaris is the optimal platform for deploying Oracle Database.

Break-Through Performance and Proven Scalability

Oracle Database deployments must be capable of fast transaction speeds and high throughput, and have the ability to scale user populations and database capacities. When deployed on SPARC servers, the Oracle Solaris platform exhibits outstanding performance and scalability for Oracle Database instances. As Figure 1 shows, while executing data warehousing tests, Oracle Solaris demonstrates near linear scalability when scaling the number of processors from 1 to 32 on Oracle's SPARC M5-32 server.

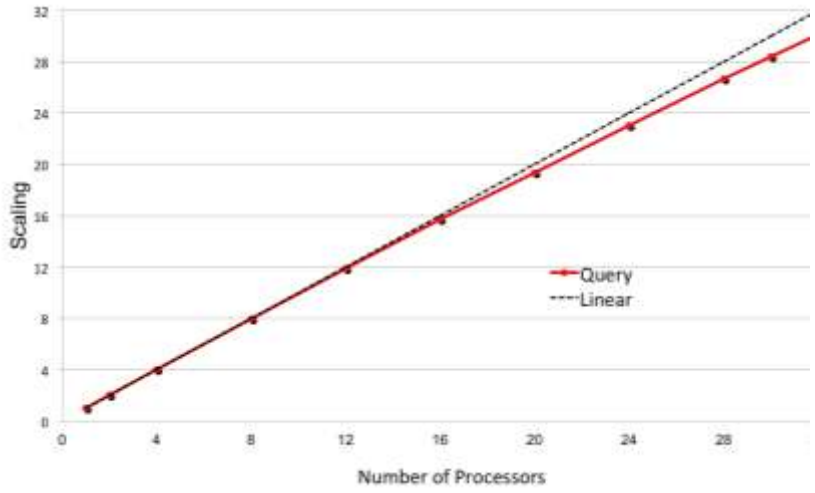

Figure 1. Oracle Solaris demonstrates near-linear scaling for data warehouse queries on Oracle's SPARC M5-32 server.

As memory prices have decreased, it is often feasible to configure large memory subsystems on SPARC servers and run databases entirely in memory. The virtual memory system in Oracle Solaris effectively scales to support large memories. Figure 2 illustrates how the use of large memories can accelerate database performance for sample scan and query operations—as much as seven times in some cases. Scans were performed with 4TB versus 12TB of memory and queries were executed using 1TB versus 8TB. (Results reflect queries cached in memory on the SPARC M5-32 server with a database workload using all 32 processors.)

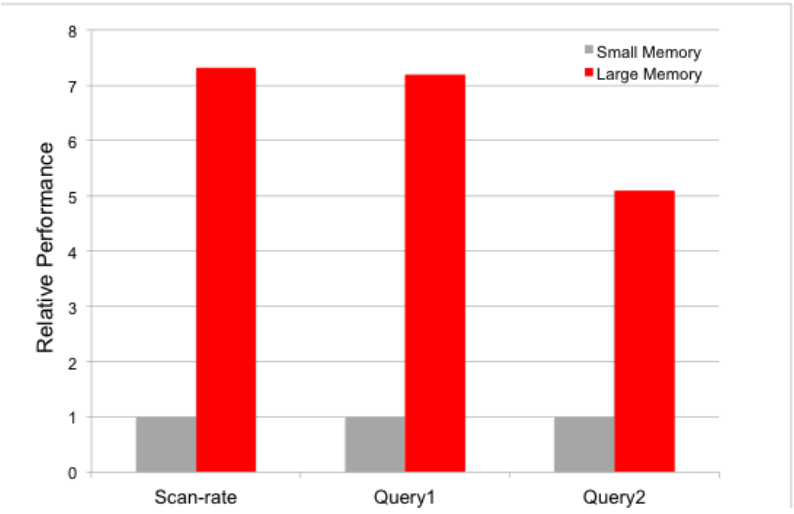

Figure 2. Large memory capacities improve database performance for database scan and query operations.

Oracle Database has a proven track record of scaling well both vertically and horizontally on Oracle Solaris. Servers based on SPARC T5, M5, or M6 processors have achieved numerous world records when running database applications such as data warehousing, online transaction processing (OLTP), and infrastructure applications. Whether Oracle Database is deployed as single or multiple instances or tested with online or batched workloads, Oracle Solaris demonstrates exceptional levels of performance and scale in customer deployments as well as in industry benchmarks. The following benchmarks offer a few proof points:

- **Database performance benchmarks (TPC-C and OLAP).** Oracle's flagship database products achieved world record performance running on Oracle's SPARC servers and the Oracle Solaris operating system. In the TPC-C benchmark, Oracle's SPARC T5-8 server (equipped with eight 3.6 GHz SPARC T5 processors) achieved a world-record single-system result of 8,552,523 tpmC running Oracle Database 11*g* Release 2 Enterprise Edition with Oracle Partitioning. As for Oracle Database 12*c*, a SPARC T5-8 server delivered world record query performance, together with near real-time analytic capability, under the Oracle OLAP Perf Version 3 workload running Oracle Database 12*c* Release 1 (12.1.0.1) on Oracle Solaris 11.[1]
- **Data warehousing benchmarks (TPC-H).** The SPARC T5-4 server produced single-server world record performance for TPC-H 3000GB and 10000GB benchmarks. These benchmarks simulate a complete solution that hosts robust decision support systems. The solution provides efficient processing of individual queries and the ability to manage large numbers of concurrent query processing streams, and it emphasizes data storage for more throughput, protection of the stored data, faster loading, and data refresh during business operations.[2]
- **SAP Standard Application Sales and Distribution (SD) Benchmark**. Oracle's SPARC M6-32 server produced a world record two-tier result for 32 processors on the SAP Standard Application SD Benchmark using SAP enhancement package 5 for SAP ERP 6.0.[3]
- **Virtualization benchmark (SPECvirt_sc2010).** Oracle's SPARC servers running Oracle Solaris 11 offer superior virtualization performance. Oracle's SPARC T5-2 server delivered a two-socket world record SPECvirt_sc2010 result of 4270 at 264 virtual machines.[4]

Accelerating Database Deployments

Because Oracle Database instances support critical business functions, it's important to be able to deploy and provision database services quickly and reliably. As a part of ongoing integration work, Oracle Solaris and Oracle Database engineers collaborated to simplify database installation, upgrade planning, and cloning on Oracle Solaris. with the goal of speeding deployments and improving the availability of database services.

Easing Database Installations and Upgrades

For Oracle Solaris 11.2, Oracle supplies an installation package group that installs all of the software needed to support Oracle Database implementations. The package group is called:

```
group/prerequisite/oracle/oracle-rdbms-server-12-1-preinstall
```

Installing this package group creates an Oracle Database configuration on the Oracle Solaris 11.2 operating system, reducing the risk of installation errors and accelerating time-to-deployment. This package group makes sure that that

1 Sources: TPC-C result ID #113032601 submitted 03/26/13 using Oracle Database 11*g* Release 2 Enterprise Edition with Oracle Partitioning, Oracle's SPARC T5-8 server, and Oracle Solaris 11.1 (www.tpc.org/1792). TPC Benchmark, TPC-C, and tpmC are trademarks of the Transaction Processing Performance Council. See also "SPARC T5-8 Delivers World Record Oracle OLAP Perf Version 3 Benchmark Result on Oracle Database 12c" (blogs.oracle.com/BestPerf/tags/t4).

2 Sources: SPARC T5-4 3TB results as of 6/7/13 (www.tpc.org/3288). SPARC T5-4 10TB results as of 11/25/13 (www.tpc.org/3293). TPC-H is a trademark of the Transaction Processing Performance Council (TPC).

3 Source: Certification Number: 2014008, www.sap.com/benchmark. Results as of March 26, 2014.

4 Source: Results from www.spec.org as of 3/5/2014 (http://www.spec.org/virt_sc2010/results/res2014q1/)

all necessary packages required for a graphical interface installation of Oracle Database 12*c* are present on the system, regardless of the server package group (`solaris-minimal-server`, `solaris-small-server`, etc.) that was used to install Oracle Solaris.

Oracle also provides ORAchk (the Oracle Configuration Audit Tool), which replaces RACcheck. (Because ORAchk performs health checks on an expanding set of Oracle products, RACcheck was renamed to ORAchk.) ORAchk proactively scans for problems across various layers of an Oracle Database deployment stack, detecting and reporting important patch recommendations and configuration settings within an Oracle Database system. It also streamlines the process of upgrades, conducting both pre-upgrade and post-upgrade readiness checks. These checks assess the upgrade readiness of Oracle RAC, Oracle Clusterware, and Oracle Solaris components and automate many manual steps detailed in upgrade-related documents. (For more information on ORAchk, see My Oracle Support Note 1268927.2.)

Rapid Provisioning by Cloning

When managing production databases, there is often a need to clone a copy of the operating system and database environment. Often an administrator will clone an existing production database to support application development and testing, troubleshooting, off-lining backups, or to maintain the continuity of database services while patching or upgrading the environment. The cloning process for a large production database can often be time consuming. Oracle Solaris Zones technology, a built-in virtualization technology, can dramatically accelerate cloning using Oracle Solaris ZFS, a state-of-the-art high-integrity file system in Oracle Solaris. As described in the paper "*How to Accelerate Test and Development Through Rapid Cloning of Production Databases and Operating Environments*," it's possible to replicate a database in an Oracle Solaris Zone up to 50 times faster than using manual database cloning methods. (Initializing the operating system in the new zone consumes most of the required time. Duplicating the database is largely instantaneous because ZFS features copy-on-write technology.)

Oracle Multitenant also empowers fast database cloning when database storage resides on Oracle ZFS Storage Appliances. Oracle Multitenant is an option for Oracle Database 12*c* that consolidates multiple pluggable databases (PDBs) in a single container database (CDB). Unlike earlier Oracle Database releases that intermingled data dictionary tables, Oracle Database 12*c* separates user data and metadata from the metadata that describes the Oracle system. This simplifies the cloning of PDBs in Oracle Multitenant. What makes the cloning process so fast is the use of Oracle ZFS Storage Appliances. Since an appliance can instantaneously snapshot a ZFS file system on which a source PDB resides, it can provision a cloned PDB almost immediately. The article "*Oracle Multitenant on SPARC Servers and Oracle Solaris*" describes how using the Oracle ZFS Storage Appliance to "snapshot copy" a 1.3TB PDB can re-create a pluggable database in just under six minutes—compared to a full copy clone that takes more than nine hours to replicate.

Enhancing Service Availability

Oracle has engineered additional memory optimizations that speed database initialization and decrease the need for restarts, eliminating downtime and increasing database availability. Oracle Solaris and Oracle Database 12*c* have been optimized for fast database startup. Optimizations include in-kernel parallel allocation of shared memory, faster spawning of background processes, and deferred database SGA allocation. Together these optimizations improve start-up time significantly. Even database instances with an SGA as large as 30 TB can be started in just a couple of minutes.

Accelerating Database Startup

In earlier versions of Oracle Solaris, shared memory operations (such as page creation, locking, and destruction) were single threaded and executed by the thread making the system call. Hence, the start of an Oracle Database instance with a large amount of memory would bottleneck on the single-threaded SGA initialization process. With huge memory capacities possible in servers based on SPARC T5, M5, or M6 processors, terabytes of memory are now commonly available to Oracle Database instances.

To eliminate an SGA initialization bottleneck, Oracle Solaris 11 features a new multithreaded kernel process called "vmtasks" that accelerates the creation, locking, and destruction of pages in shared memory. This behind-the-scenes process divides a shared memory segment at page boundaries and gives each thread a chunk of pages to process, thereby parallelizing the creation and destruction of shared memory segments and accelerating database startup and shutdown. Each shared memory operation is given a number of threads equal to the value returned by the MAX function (where *NCPU* is the total number of server compute threads):

max number of vmtasks (threads) per operation = MAX(16,*NCPU*/8)

On Oracle's SPARC T5-2 server with 256 compute threads, for example, the operating system assigns up to 32 threads for each shared memory operation. Best of all, since this optimization is the default behavior for Oracle Solaris 11 on all platform architectures, it requires no administrative action or tuning.

Oracle Database 12*c* also introduces a group of new background processes, ora_SA*nn*, that accelerate database startup. Oracle Database 12*c* starts up while these background processes are initializing the SGA in parallel. To assess the impact of deferring SGA initialization in this way, Oracle engineers conducted a study that compared startup times for Oracle Database 12*c* and Oracle Database 11*g* while scaling the SGA size in line with the number of sockets. As shown in Table 1, deferring SGA initialization in Oracle Database 12*c* resulted in significantly faster startup times—more than twice as fast as the initialization of Oracle Database 12*c* and Oracle Database 11*g* without the use of this feature. Deferring SGA initialization is the default out-of-the-box behavior with Oracle Database 12*c* and this feature is optimized for run-time environments such as NUMA configurations.

TABLE 1. SPEEDUP USING DEFERRED SGA INITIALIZATION

SGA Size (in GB) / Number of Sockets	Oracle Database 12*c* Deferred vs. Oracle Database 12*c* Non-Deferred	Oracle Database 12*c* Deferred vs. Oracle Database 11*g* Release 2
1024 / 4	2.45x	2.10x
2048 / 8	2.34x	2.41x
4096 / 16	3.00x	2.84x
6144 / 24	2.64x	2.34x
8192 / 32	2.63x	2.48x

Dynamic SGA Resizing

Oracle Solaris 11 includes a new NUMA-optimized, granular shared memory called Optimized Shared Memory that replaces Dynamic Intimate Shared Memory and enables dynamic SGA resizing. Oracle Database 12*c* can take advantage of Optimized Shared Memory automatically on systems running Oracle Solaris 11. Because of this feature, the database can automatically manage Oracle Database 12*c* SGA shared memory segments without an administrator having to reserve memory and reboot the instance whenever the SGA is resized. By reducing the number of Oracle Database restarts, this capability improves database availability and increases service levels.

More information about this feature is available in the Oracle Database 12*c* administrator's reference documentation (see *Administering Oracle Database on Oracle Solaris* and My Oracle Support Note 1579199.1, "*Oracle Database 12c Takes Advantage of Optimized Shared Memory Feature on Oracle Solaris*").

Increasing Throughput

Oracle Solaris allows server resources to be applied effectively to database workloads, which helps to optimize query response times and transaction throughput. Oracle Solaris and SPARC servers feature built-in virtualization techniques that allow system resources to be allocated to specific workloads (a workload consolidation discussion, later in this paper, discusses these techniques and best practices). In addition to efficient resource management, Oracle Solaris features the ability to identify a "critical thread" to the process scheduler, which helps to optimize transaction throughput and performance.

In modern multiprocessor chip designs, threads compete for more than just processor time—they also compete for shared CPU resources such as execution pipelines, cache, floating point units, and so forth. Administrators can use manual methods, manipulating resource pools and assigning processor sets, to dedicate resources to higher priority processes. In Oracle Solaris 11, critical threads are a cross-architecture optimization that simplifies the task of allocating resources to important threads. Using the command line interface, an administrator can raise the priority of an identified critical application thread to 60 and select the FX scheduling class. When a thread is deemed critical, Oracle Solaris tries to provision more exclusive access to computing resources for that thread.

The Oracle Database log writer process (LGWR) logs transactions for recovery purposes, and can often be a throughput bottleneck. In the past, it has been a common best practice to statically bind LGWR to a processor core. On Oracle Solaris 11, an administrator with the appropriate privilege can use the following command to elevate LGWR to the priority of a critical thread:

```
# priocntl -s -c FX -m 60 -p 60 -i pid <PID_of_LGWR>
```

Oracle Database 12*c* automatically takes advantage of this Oracle Solaris 11 feature and best practice, assigning critical thread status to LGWR at startup. For earlier versions of Oracle Database, administrators can use the `priocntl` command to designate LGWR manually as a critical thread.

Enhancements to the virtual memory system in Oracle Solaris 11 also help to increase database performance and scale. Oracle Database can allocate large memory pages (2GB in size) to its system global area (SGA), the group of shared memory areas dedicated to an Oracle Database instance. Large page sizes improve database performance by reducing costly Transaction Lookaside Buffer (TLB) misses in the CPU architecture. In addition, enhancements to memory prediction in Oracle Solaris enable monitoring of memory page use and adjustments to the page size to match application and database needs, which also helps to increase performance.

Enhancing Observability

For real-time troubleshooting and debugging, Oracle Solaris DTrace is a dynamic tracing capability that provides extensive system observability. The integration of Oracle Solaris DTrace probes with Oracle Database 12*c* allows administrators to see what's happening on live production systems safely and in real time.

Oracle Solaris DTrace is now integrated into Oracle Database V$ views, providing end-to-end visibility into database transaction I/Os—tracing the full I/O path from storage devices through the Oracle Solaris kernel into Oracle Database 12*c* itself. During runtime on a production system, V$KERNEL_IO_OUTLIER can collect and store data that provides insights into "outlier" I/O operations (I/Os that exceed a latency threshold of 500ms). The database administrator can trace the specific I/O requester, the I/O device and the exact time spent in each layer. This

advanced observability allows an administrator to identify the root cause of an I/O issue and resolve problems more quickly, improving Oracle Database 12*c* performance.

Consolidating Database Workloads

Consolidating database workloads facilitates an efficient use of infrastructure resources, which helps to lower costs. Since available system resources are pooled, it can also help to streamline database provisioning, speeding up the time it takes to respond and fulfill requests for new instances. Consolidation is often the first logical step towards a more-agile database as a service (DaaS) or cloud-based service delivery model.

When consolidating many Oracle Database instances, it's important to remember that different workloads have different requirements. To accommodate the requirements of each workload, database architects must balance different goals for isolation, performance, availability, and management ease (for example, for cloning, backup, and so on) along with cost.

Since databases frequently contain sensitive data—such as financial or personal data (for example, credit card, identity, or healthcare information)—a foremost concern is isolation. The ultimate method of isolating a database is, of course, to run it on a single server. Physical isolation is more costly since system and storage resources are often underutilized. It's also more complicated to manage an increasingly large number of servers that sprawl across a data center. When multiple databases are merged on a single system to realize efficiencies, deployments must carefully encapsulate database instances, safeguard processes, and protect sensitive data. Consolidating Oracle Database workloads on SPARC servers and Oracle Solaris offers excellent flexibility since there are several options available for multitenant consolidation. The consolidation options fall into two categories: database approaches and server-based approaches.

Database Consolidation Approaches

Two database-oriented approaches are frequently used to consolidate single instance databases:

- "Stacking" or colocating database instances on the same server
- In-database virtualization with the Oracle Multitenant option for Oracle Database 12*c*

Both methods result in better resource utilization when compared to separate, physically isolated deployments on individual servers.

Colocating database instances on the same server helps to reduce capital expenses (CapEx) and operating expenses (OpEx), but the stacking approach can still be somewhat management-intensive. Administrators often turn to intricate resource management strategies to acquire adequate resources, such as using instance caging to control CPU usage, setting `sga_target` and `pga_aggregrate_target` to control memory usage, or binding database instances to NUMA nodes by defining `processor_group_name` in `init.ora`. In addition, database instances must be managed individually with respect to backup, restore, patching, and disaster recovery.

Oracle Database 12*c* offers a new database consolidation model—Oracle Multitenant—in which multiple pluggable databases (PDBs) are consolidated within a single container database (CDB). Oracle Multitenant provides the isolation of a single database while allowing multiple PDBs to share the SGA and background processes of the same CDB, reducing resource requirements significantly. In a scalability study of Oracle Multitenant on Oracle SuperCluster T5-8, Oracle engineers measured the savings gained by using Oracle Multitenant compared to the approach of stacking single-instance databases. Oracle Multitenant enabled 80 percent better performance and the consolidation of 50 percent more databases, decreasing resource requirements by 64 cores, 368GB of memory,

and 225,000 IOPS. Figure 3 illustrates a few results of the resource efficiency testing (for details, see the "*Oracle Multitenant on SuperCluster T5-8: Scalability Study*").

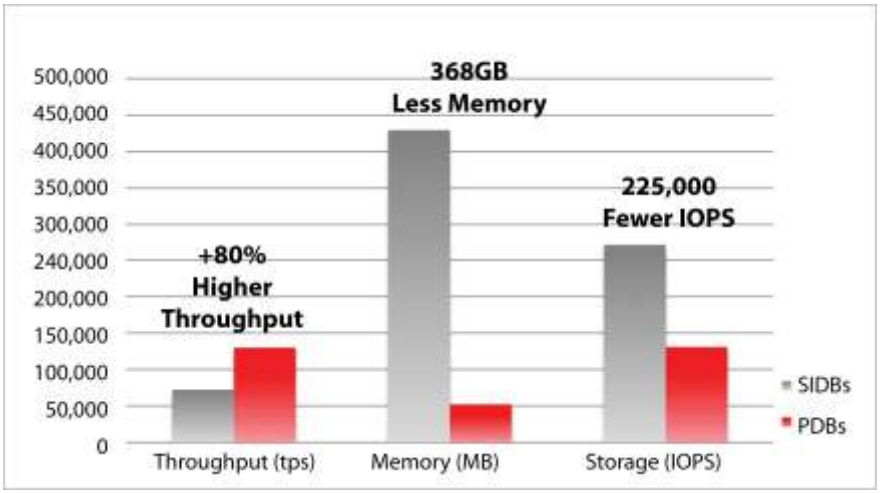

Figure 3. Scaling Benefits of Oracle Multitenant on Oracle SuperCluster T5-8.

Oracle Solaris resource pools are also a mechanism that enables NUMA-aware database consolidations. By establishing an Oracle Solaris resource pool and associating it with a `processor_group_name` in `init.ora`, all database processes are then bound to that processor group. For NUMA architectures, assigning specific resources in this way (known as NUMA binding or "pinning") helps to improve performance since a processor can access local rather than non-local memory. This optimization is a direct result of joint engineering between the Oracle Solaris and Oracle Database product teams.

Of course, it's also possible to combine the use of deployment methods: stacking database instances and Oracle Multitenant databases on a single SPARC server platform and implementing a hybrid consolidation approach that also uses server-based virtualization technologies. In practice, many deployments combine database and server-based approaches.

Server-Based Consolidation Approaches

Oracle's SPARC servers and Oracle Solaris feature built-in virtualization technologies (Figure 4) that are available at no extra cost:

- **Physical domains (also called PDoms)**: This technique (available on Oracle's servers that use SPARC M5 or M6 processors) divides a single physical machine into independent isolated servers.
- **Oracle VM Server for SPARC**: This hypervisor-based technology (available on servers that use SPARC T5, M5, and M6 processors) allows multiple OS instances to coexist on the same physical infrastructure with dynamic reallocation of hardware resources.
- **Oracle Solaris Zones**: This software approach (available on any SPARC server running Oracle Solaris 10 or Oracle Solaris 11) allows many applications to coexist within a single operating system instance with assigned resources.

Many articles and white papers discuss these technologies in detail (see references listed at the end of this paper). Oracle Multitenant works with all these server virtualization options and provides an additional layer of database virtualization.

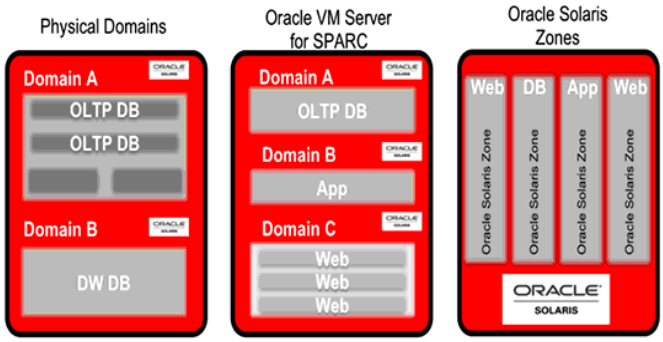

Figure 4. Zero-cost virtualization technologies are built into SPARC servers and Oracle Solaris.

These three server virtualization techniques provide different benefits. Physical domains supply full hardware isolation; a hardware or software failure on one physical domain does not affect other domains. Physical domains, however, are less flexible and are available only on high-end servers based on SPARC M5 and M6 processors.

Unlike some commercial virtualization technologies, Oracle VM Server for SPARC and Oracle Solaris Zones scale extremely well with near-zero overhead. Using Oracle Solaris Zones offers the greatest flexibility because zones are implemented in software—an administrator can easily and dynamically allocate and modify zone resources. However, zones do not fully isolate resources from a hardware perspective. A disk failure or a system memory fault, for example, can impact more than one zone. For this reason zones are often implemented in conjunction with an approach that enhances resource isolation such as Oracle VM Server for SPARC or physical domains.

A shared benefit of these virtualization approaches is that they can be configured to create hard partitions that comply with Oracle Database and application licensing. When only a subset of cores on a large system is needed for a database instance, using these server-based approaches to construct a hard partition can generate substantial software savings. Oracle's SPARC M5-32 server, for example, has 32 six-core processors. Using these server-based technologies to create virtual environments, each with a subset of cores, can considerably reduce licensing costs for Oracle Database and Oracle Database options. For certification information on supported virtualization and partitioning technologies for Oracle Database and RAC product releases, see the site www.oracle.com/technetwork/database/virtualizationmatrix-172995.html.

Another benefit of using these server-based virtualization technologies is that they enable different Oracle Database and Oracle Solaris versions to be consolidated on the same server. A single SPARC server, for example, can actually support Oracle Database 12*c* as well as earlier releases of Oracle Database and applications. As an example, Figure 5 depicts a hybrid deployment that combines the use of database and server-based consolidation approaches on a single SPARC M5-32 server. The server supports stacked single-instance and Oracle Multitenant implementations of Oracle Database 12*c* on Oracle VM Server for SPARC domains that host Oracle Solaris 11. Earlier version databases are deployed on special types of zones that can run earlier operating system versions. In this way, Oracle Solaris supports the consolidation of legacy databases and applications on the same machine.

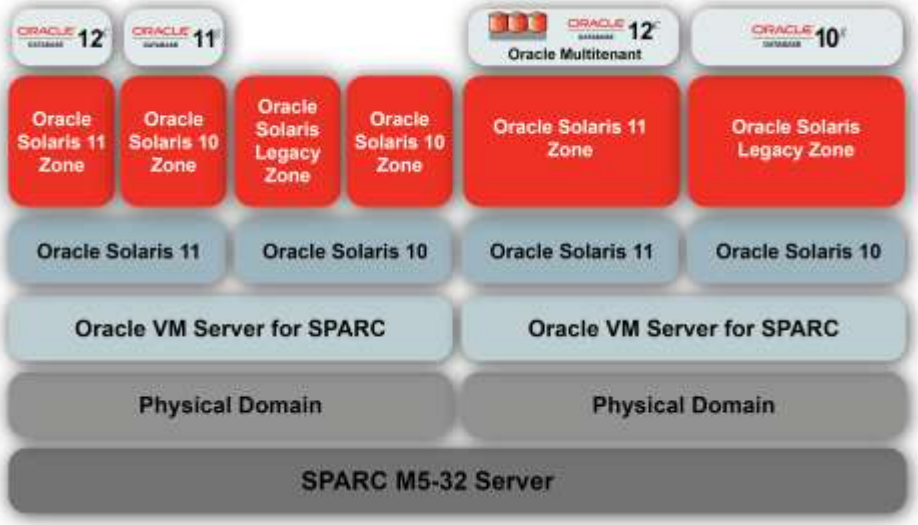

Figure 5. A hybrid approach can combine database and server-based consolidation approaches to optimize resource utilization, scalability, and management efficiency.

Selecting a Deployment Approach

As shown in Figure 5, SPARC servers running Oracle Solaris 11 offer tremendous flexibility for consolidating Oracle Database instances. Given the range of technologies and configuration possibilities, configuring an optimal solution for a given database workload might seem like a daunting task. To simplify configuration decisions, the next few sections provide implementation guidelines and best practices for using Oracle Solaris Zones and Oracle VM Server for SPARC in conjunction with Oracle Database workloads.

Choosing Oracle Solaris Zones

Oracle Solaris Zones are a software construct that enables bare-metal I/O performance. They are extremely lightweight, allowing thousands of virtual environments to be defined with little overhead or impact to system performance. An administrator manages zones via an Oracle Solaris instance in the global zone, controlling what resources (CPU, drivers, file systems, and so on) are available in each non-global zone. Users and applications see only the processes and resources isolated within the non-global zone, and have no visibility of data or processes in other zones. Zones also provide a convenient administrative boundary, allowing zone administration to be limited or delegated if required. Oracle Solaris 11.2 extends zone functionality with Kernel Zones that can be independently patched or upgraded.

When deploying Oracle Database instances, use non-global Oracle Solaris Zones in the following cases:

- **To provide isolation.** Oracle Solaris Zones isolate database instances with respect to fault, operation, network, security management, and resource allocation control. A failure or reboot of one non-global zone has no impact on other non-global zones (unless, of course, a failure is due to a shared component). A zone reboot is much faster than a full server reboot (seconds versus minutes), so a database in a rebooted zone is available more quickly than in the case of a full server reboot.
- **To supply a unique identity.** If a database instance requires a separate identity (for example, an independent host name and IP address), an Oracle Solaris Zone is a uniquely identifiable virtual environment. Oracle Solaris enables network virtualization, allowing an administrator to create virtual network interfaces (VNICs) with unique hostnames and IP addresses (Figure 6). Each VNIC is implicitly connected to a virtual switch that corresponds to

a physical interface. By using Oracle Solaris Zones and applying networking resource controls, administrators can manage network bandwidth and align its use with required service levels.

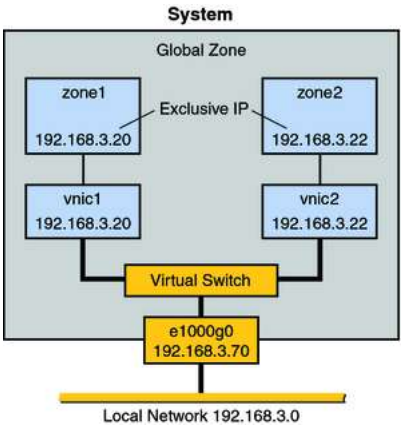

Figure 6. Oracle Solaris Zones enable VNICs (with unique hostnames and IP addresses) that are subject to resource controls.

- **To create independently managed database environments.** Oracle Solaris Zones create autonomous environments that can be booted, patched, and shut down without impacting other application environments.
- **To support more than one Oracle RAC instance in a single database domain.** For some test and development efforts, it might be desirable to create multiple Oracle RAC instances within a single physical server (this tactic, of course, should be avoided for production Oracle RAC databases). Oracle Solaris Zones are virtual environments that function like independent physical hosts when installing separate Oracle RAC instances.
- **To migrate database instances between virtual environments.** Oracle Solaris Zones simplify database instance migration from one environment to another. When a database needs more CPU power, for example, it's possible to add CPUs to an Oracle Solaris Zone and reboot the zone. If a database needs more compute capacity than what's available in the physical server, then the zone can be migrated to a zone on a larger server.

When deploying databases with Oracle Automatic Storage Management (ASM) and consolidating them using Oracle Solaris Zones, Oracle Database 12*c* features Oracle Flex ASM, a new deployment model that increases database instance availability and reduces Oracle ASM related resource consumption. Oracle Database 11*g* requires a separate Oracle ASM instance per zone. With Oracle Database 12*c*, Oracle Flex ASM allows a database server to use a non-local Oracle ASM instance. In this way, Oracle Flex ASM reduces the number of ASM instances needed in a consolidated environment.

Best Practices for Oracle Databases on Oracle Solaris Zones

Using Oracle Solaris Zones is strongly recommended for Oracle Database instance isolation and containment, regardless of other isolation methods used when consolidating databases. Planning how CPU resources are allocated to individual zones helps to optimize performance for database instances running in Oracle Solaris Zones. When sizing CPU resources for a zone, administrators should take into account the workload requirements (such as the number of active Oracle Database instances) in the zone and the specific deployment scenario. Optimal performance occurs when each database instance has exclusive access to the pipelines, cache, and other resources within an individual core and an adequate number of complete cores are configured for the workload. Smaller database instances can be configured on a single core. For larger database instances, creating larger zones with multiple cores is recommended.

To allocate CPU threads (vCPUs) to particular database workloads, configure a resource pool with an associated processor set. As an example, the following commands configure a processor set and assign all eight vCPUs from a single core of a SPARC T4-2 server:

```
# poolcfg -c 'create pset dbPset_set(uint pset.min=8;uint pset.max=8)'
# poolcfg -c 'create pool dbPool'
# poolcfg -c 'associate pool dbPool(pset dbPset)'
# poolcfg -c 'transfer to pset dbPset(cpu 0;cpu 1; cpu 2; cpu 3; cpu 4; cpu 5;
    cpu 6; cpu 7)'
# pooladm -c
# pooladm
```

The resource pool and its processor set are then assigned to a zone:

```
# zonecfg -z my-zone
zonecfg: my-zone> set pool=dbPool
zonecfg: my-zone > end
zonecfg: my-zone > commit
zonecfg: my-zone > exit
```

When consolidating multiple instances that use the same Oracle Database release, it is best practice to use a shared release area in conjunction with Oracle Solaris Zones. Install the Oracle Database software in the global zone and then mount the installation directory in each instance's non-global zone. For example, the following commands mount the shared release area `/u01` in the zone `"my-zone"`.

```
# zonecfg -z my-zone
zonecfg:my-zone> add fs
zonecfg:my-zone:fs> set dir=/u01
zonecfg:my-zone:fs> set special=/u01
zonecfg:my-zone:fs> set type=lofs
zonecfg:my-zone:fs> end
zonecfg:my-zone> commit
zonecfg: my-zone> exit
```

Note that a zone must be rebooted for configuration changes to take effect. Using a shared release area simplifies the process of software maintenance since patches and updates are applied to a single location that can be accessed by multiple instances.

When consolidating more than 10 database instances on a server, apply the patch for bug ID #18499306 on a NUMA server.

More information about deploying Oracle Database on Oracle Solaris Zones is available in the white paper, "*Encapsulating Oracle Databases with Oracle Solaris 11 Zones*."

Choosing Oracle VM Server for SPARC

Taking a hybrid approach—combining Oracle Solaris Zones and Oracle VM Server for SPARC—extends the flexibility and separation of Oracle Database deployments. Oracle VM Server for SPARC (previously called Sun Logical Domains) uses a hypervisor layer built into the SPARC processor that manages virtual machines called domains. Server resources (CPUs, memory, PCIe cards, network, and storage) can be divided up and allocated to each domain, providing strong encapsulation. In addition, each domain runs an independent operating system using the designated resources.

When deploying Oracle Database, Oracle Solaris Zones help to address requirements for isolation, separate identification, and ease of database migration. A hybrid approach of using Oracle VM Server for SPARC and Oracle Solaris Zones is ideal when a deployment must tackle the following additional requirements:

» **To support different versions of Oracle Solaris and Oracle Database.** Since each Oracle VM Server for SPARC domain runs its own operating system and patch set, a domain can host earlier Oracle Database

releases. For example, a domain can host Oracle Solaris 10 with specific patches and support earlier releases of Oracle Database (such as Oracle Database 10*g* or Oracle Database 11*g*). Oracle Solaris 10 offers special legacy zones that are able to host some Oracle Solaris 8 or Oracle Solaris 9 legacy applications, offering an easy migration path for aging, high-cost legacy systems. (Figure 5 on page 11 depicts legacy applications deployed on Oracle Solaris legacy zones.)

- **To isolate memory between database servers.** Each Oracle VM Server for SPARC domain gets its own resources, including memory, so there is strict isolation between Oracle Database instances.
- **To isolate faults within a virtual machine.** An Oracle VM Server for SPARC root domain has no dependencies on any other virtual machine on the server. If a failure occurs in a domain, it is localized to that domain and there is no impact on any other domain.
- **To migrate an active domain easily to another physical machine.** Oracle VM Server for SPARC supports live migration—moving an active domain to another physical machine—even across the same or different generations of supported SPARC servers. Live migration allows users to continue to access database services running on the domain.

Best Practice Using Oracle VM Server for SPARC

Initially, all system resources are allocated to the Oracle VM Server for SPARC control domain. After releasing a subset of resources, an administrator can assign resources to other logical domains, configuring domains in units of CPU threads (vCPUs). By default Oracle VM Server for SPARC tries to assign vCPUs within the same core to achieve the best performance but this does not always occur. To force Oracle VM Server for SPARC to configure domains in units of entire CPU cores instead of individual CPU threads, an administrator can specify the use of whole cores. Creating a domain with CPU whole cores can bring performance advantages that stem from processor affinity. Perhaps more importantly, this strategy is in line with Oracle hard partitioning requirements for software licensing.

To run an application such as Oracle Database that follows the Oracle hard partition licensing scheme, a domain must be configured with CPU whole cores and a cap must be set for the maximum number of cores (this cap must correspond to the number of cores specified by the Oracle software license). Using a whole-core configuration and setting a maximum CPU cap limits the number of CPU cores that a particular domain can use.

While an administrator can use a graphical interface (for example, Oracle Enterprise Manager) to set up compliant domains, the following two commands can also be used to create a domain that adheres to Oracle hard partition licensing:

Set the number of CPU whole cores for the domain.

```
# ldm set-core number-of-CPU-cores domain
```

Set the maximum number of CPU cores for the domain.

```
# ldm set-domain max-cores=max-number-of-CPU-cores domain
```

For more information, see the papers "Hard Partitioning With Oracle VM Server for SPARC" and the "Oracle Partitioning Policy".

Enterprise-Level Security and Management

Encapsulating Oracle Database instances with Oracle Multitenant, Oracle Solaris Zones, and Oracle VM Server for SPARC helps to protect database workloads from interruption and unauthorized access to database records. Oracle Solaris offers proven security to safeguard Oracle Database implementations, with advanced features such as system- and network-enforced security, role-based access controls, and sophisticated validation, monitoring, and

auditing capabilities. Because of this advanced feature set and proven stability, Oracle Solaris is widely deployed to support security-sensitive Oracle Database implementations.

Validating Configurations to Meet Compliance Standards

Oracle Solaris 11 provides security advances that help to maintain service levels, detect intrusions, and protect against data and system compromise. Checking systems periodically to confirm that they meet configuration standards (and have not been subject to tampering) delivers database services that are more stable and secure.

Oracle Solaris 11 includes a framework that manages a variety of compliance benchmarks and assessments. The framework is based on the Security Content Automation Protocol (SCAP), a line of standards managed by the U.S. National Institute of Standards and Technology (NIST) that provides a standardized approach to maintaining enterprise system security, including:

- Automatically verifying the presence of critical updates
- Checking system security configuration settings
- Examining systems for signs of compromise

Oracle Solaris 11.2 includes an Oracle Solaris Security Policy benchmark with two profiles (Baseline and Recommended) and an Oracle Solaris Payment Card Industry PCI-DSS benchmark. (For details see the *Oracle Solaris 11.2 Security Compliance Guide*.) At the file system level, the Oracle Solaris Basic Audit Reporting Tool also enables organizations to detect changes to files, directories, and other file system objects using a flexible policy that can be used across an entire system or just for security-critical files.

Oracle Solaris supports Immutable Zones, which are zones with read-only roots. Configuring Immutable Zones can protect a system from malicious or accidental tampering of system binaries or configuration files during run-time. Both global and non-global zones can be configured as Immutable Zones.

At installation, eliminating unnecessary software packages and services helps to minimize possible avenues of attack. Hardening or minimizing an operating system is a common practice that shrinks the attack surface by installing only the software packages needed for operation. Oracle Solaris 11.2 provides a `solaris-minimal-server` package group that simplifies the installation of hardened systems. Installing the `oracle-rdbms-server-12-1-preinstall` package group in conjunction with the `solaris-minimal-server` package group creates a tested and minimized Oracle Database configuration on Oracle Solaris 11.2.

Monitoring and Auditing

With the potential for many individual databases, end users, and administrators operating within a consolidated database architecture, proper activity monitoring and auditing is essential. Oracle Solaris 11 supports monitoring and auditing at the workload, network, and database layers, which helps to detect anomalies and unusual activity.

The Oracle Solaris audit facility records actions taken by users and services on the operating system. Enabled by default in Oracle Solaris 11, the audit facility uses policies that can track system, administrator, and end user activity. When properly configured, the Oracle Solaris audit facility creates a baseline that reflects typical system activity, detects unauthorized actions, senses attempts to circumvent security controls, and collects evidence to support audit or incident response requirements. Since built-in virtualization capabilities such as Oracle VM Server for SPARC and Oracle Solaris Zones are often strategic components in Oracle Database deployments, auditing can be configured to record events associated with domain and zone configuration, management, and usage. Auditing can also be configured to record access to Oracle Integrated Lights Out Manager (Oracle ILOM) functionality and interactions with a SPARC system's service processor. Oracle ILOM generates a set of audit records for logins, administrative actions, and configuration changes.

Within Oracle Database itself, the database implementation includes native, fine-grained auditing that allows security administrators to set auditing policies at the database object level. Database auditing policies can help to identify potential issues while minimizing the amount of audit data generated.

Securing Network Traffic

Using Oracle Solaris Zones to virtualize network interfaces and creating VLANs helps to isolate and protect network traffic. By configuring virtual networks within Oracle Solaris, network traffic is easily isolated for greater security and data protection. Assigning database instances to specific IP addresses (that correspond to certain network interfaces and VLANs) limits communication only to those networks.

Oracle Solaris also includes an IP Filter feature that can selectively record inbound and outbound network communications. Each record generated by IP Filter includes a timestamp and the source and destination address and port, as well as whether the communication attempt was permitted or denied by the host-based firewall. For security-critical services, even the logging of permitted requests can help to meet stringent security and compliance mandates. IP Filter can be used at both the domain and non-global zone levels so that organizations can segment network policies and activity records according to deployment requirements.

Cryptographic Support

Oracle Solaris provides a robust cryptographic framework for applications that require encryption, decryption, hashing, signature generation and verification, certificate generation and verification, and message authentication. The U.S. National Institute of Standards and Technology (NIST) has validated this framework to the FIPS 140-2 Level 1 standard. Running applications in FIPS 140 mode is a requirement for many regulated industries and U.S. government agencies that process sensitive but unclassified information.

Servers based on the SPARC T5, M5, or M6 processors feature on-core cryptography engines that can accelerate encryption and decryption operations, including secure database queries. The Oracle Solaris library, `libsoftcrypto`, provides a software interface to access the instruction set in these SPARC processors. Transparent Data Encryption (TDE) with Oracle Database on these systems, for example, can take advantage of the native encryption and decryption capabilities of these processing engines, accelerating encryption and decryption without the need for additional hardware. Oracle VM Server for SPARC can also use these on-core cryptography engines when performing live migration tasks.

End-to-End Database Infrastructure Management

Oracle Enterprise Manager supplies a comprehensive “single pane of glass” interface for managing all components in the Oracle stack—Oracle hardware, firmware, virtual systems (including domains and zones), and operating system instances—and for managing patching, migrations, and updates. By adding other software in the Oracle Enterprise Manager family, such as Oracle Cloud Management Pack for Oracle Database or Oracle Cloud Management Pack for Oracle Middleware, administrators can gain fine-grained control of Oracle Database instances and Oracle Applications from a fully integrated enterprise management tool. Oracle Cloud Management Pack for Oracle Database can also enable self-service database provisioning with applied resource quotas. An integrated self-service approach helps to simplify database infrastructure management, saving time and labor costs while improving productivity.

Conclusion

Oracle Solaris is an enterprise-class operating environment that enables peak database performance and scalability levels, strict data isolation, robust security, and ease of management. Built-in virtualization technologies facilitate flexible consolidation options, allowing solution architects to tailor configurations that meet design requirements while reducing cost and simplifying management. It's no surprise that Oracle Solaris continues to be the leading enterprise platform and why it is extensively deployed to host Oracle Database instances—including deployments of Oracle Database 12*c* and the Oracle Multitenant option.

Oracle strives to provide best-in-class technologies to support its flagship Oracle Database product. That's why the company invests in engineering, testing, and enhancements that make Oracle Solaris and Oracle's SPARC servers the optimal platform for Oracle Database deployments. The Oracle Solaris and Oracle Database engineering teams have worked closely to improve all aspects of database deployments, from simplifying installation to accelerating performance to troubleshooting I/O and more. Joint integration has produced valuable optimizations, including the results described in this paper. The collaboration efforts have produced world-record performance and excellent database scalability with additional processors and memory.

The extensive Oracle Solaris community, along with many "how to" articles, can help database architects get started in designing an optimized database infrastructure. Contact your Oracle representative to learn more or visit oracle.com/solaris.

For More Information

For more information, visit the resources listed in Table 2.

TABLE 2. RESOURCES

Web Resources	**URL**
Oracle Solaris home page	oracle.com/solaris
Oracle Database home page	oracle.com/database
Oracle benchmark results	oracle.com/benchmarks
Oracle Enterprise Manager 12*c* home page	oracle.com/enterprise-manager
Oracle Solaris download page	oracle.com/technetwork/server-storage/solaris11/downloads/
White Papers and Technical Articles	URL
Oracle Multitenant	oracle.com/technetwork/database/multitenant-wp-12c-1949736.pdf
Encapsulating Oracle Databases with Oracle Solaris 11 Zones—Consolidation with Strong Isolation	oracle.com/technetwork/database/database-cloud/private/dbcloud-s11-zones-wp-1911914.pdf
Oracle Multitenant on SPARC Servers and Oracle Solaris	oracle.com/technetwork/articles/servers-storage-admin/multitenant-on-sparc-solaris-2016889.html
Oracle Multitenant on SuperCluster T5-8: Scalability Study	oracle.com/technetwork/database/multitenant/learn-more/oraclemultitenantt5-8-final-2185108.pdf
How to Accelerate Test and Development Through Rapid Cloning of Production Databases and Operating Environments	oracle.com/technetwork/server-storage/hardware-solutions/o13-022-rapid-cloning-db-1919816.pdf
Oracle's SPARC M5-32 and SPARC M6-32 Servers: Domaining Best Practices	http://www.oracle.com/technetwork/server-storage/sun-sparc-enterprise/documentation/o13-056-m5-domaining-1991544.pdf
Hard Partitioning With Oracle VM Server for SPARC	oracle.com/technetwork/server-storage/vm/ovm-sparc-hard-partitioning-1403135.pdf
Oracle Partitioning Policy	oracle.com/us/corporate/pricing/partitioning-070609.pdf
Configuring Oracle Solaris ZFS for an Oracle Database	oracle.com/technetwork/server-storage/solaris10/config-solaris-zfs-wp-167894.pdf
Virtualization and partitioning for Oracle Database license compliance	oracle.com/technetwork/database/virtualizationmatrix-172995.html
Taking Your First Steps with Oracle Solaris 11	oracle.com/technetwork/articles/servers-storage-admin/o11-112-s11-first-steps-524819.html

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

Authors: Ritu Kamboj, Ginny Henningsen

CONNECT WITH US

blogs.oracle.com/oracle
facebook.com/oracle
twitter.com/oracle
oracle.com

Hardware and Software, Engineered to Work Together