ORACLE SOLARIS

ORACLE DATABASE

An Oracle White Paper
October 2013

Encapsulating Oracle Databases with Oracle Solaris 11 Zones

Consolidation with Strong Isolation

ORACLE

Introduction

In database cloud deployments, companies host multiple databases for use by various internal groups (private clouds) or external clients (public or community clouds). Whenever multiple databases are deployed together on shared infrastructure, the solution must take into account the degree of isolation each database will require with respect to faults, operations, security, and shared resources.

In many database cloud deployments, Oracle Database features and options will provide the required isolation. This allows consolidating multiple Oracle databases natively onto a shared infrastructure, without the need for further isolation. In native consolidations, all databases share a single Oracle Grid Infrastructure. This approach is described in detail in the Oracle white paper "Best Practices for Database Consolidation in Private Clouds".

Database clouds hosting databases with security or compliance considerations have higher requirements for isolation. These could include sensitive data with privacy requirements or data from multiple companies that cannot be aware of each other (that is, a public cloud). Such deployments may need to apply additional technologies or controls beyond those available in a native consolidation.

Implementing higher degrees of isolation can be accomplished by encapsulating each database environment. Encapsulation can be accomplished with physical or logical isolation techniques. This paper will describe the options and make a detailed analysis of how Oracle Solaris 11 Zones efficiently provide encapsulation to Oracle database clouds.

Physical and Logical Encapsulation

Physically encapsulating a database dedicates a hardware infrastructure for that database. Dedicated servers or an isolated system domain within a server are typical approaches. This model is generally not a preferred cloud model because it does not make efficient use of physical resources: when the database is not busy, unused system resources are idle. If the server is sized for occasional peak workloads, significant resources will be idle most of the time. Therefore, this model will be applicable for only the most extreme isolation or performance requirements.

Logical encapsulation with virtualization provides high levels of isolation and addresses the requirements of many scenarios that need strong isolation. This allows for databases to be consolidated by running multiple virtual environments, each hosting a database, on a shared hardware infrastructure. By consolidating on shared resources, those resources can be more efficiently utilized. The two options for logical encapsulation are virtual machines (VMs), which are hypervisor-based, and virtual operating environments, which are OS-based.

VMs offer very strong encapsulation. However each VM carries its own software footprint including an OS instance, which in turn runs its own scheduler and requires its own memory and swap, all of which is managed by the underlying hypervisor. This adds maintenance and performance overhead and I/O latency to the deployment. Since all I/O is managed through a hypervisor, heavy workloads may create bottlenecks, leading to unacceptable response times. This is especially true for database deployments, which tend to be I/O-bound rather than CPU-bound.

Virtual operating environments, such as Oracle Solaris 11 Zones, are a different approach to virtualization. While a VM is hypervisor-based, an Oracle Solaris Zone is a pure software construct managed by a single Oracle Solaris instance (the global zone). Zones provide strong isolation and also provide bare-metal I/O performance. They are so lightweight that thousands could be deployed on a system with negligible impact on overall system performance. Oracle Solaris 11 Zones are available on any platform—SPARC or x86—that supports Oracle Solaris 11. Some platforms, such as Oracle SuperCluster, offer extra capabilities that enhance the value of zone-based solutions.

Oracle Solaris 11 Zones provide very strong isolation for applications and are also an administrative boundary. The applications and users in an Oracle Solaris 11 Zone perceive a dedicated system (for example, namespace and file system mount points), but in fact they can use only the system resources (CPU, drivers, file systems, and so on) that the administrator of the global zone makes available to them. Users logged in to a zone cannot see other zones. The administrative commands available in a zone are limited, and can be further limited or selectively expanded by the administrator of the global zone.

For readers familiar with Oracle Solaris 10, it is important to note that Oracle Solaris 11 Zones are a significant evolution beyond Oracle Solaris 10 Containers. Improvements, such as complete integration with Oracle Solaris 11 networking and simpler lifecycle management, are two important examples.

This paper will describe the isolation that Oracle Solaris 11 Zones provide in a database consolidation deployment in which each database is encapsulated in its own virtual operating environment; that is, each database is encapsulated in its own Oracle Solaris 11 Zone, as shown in Figure 1. [1]

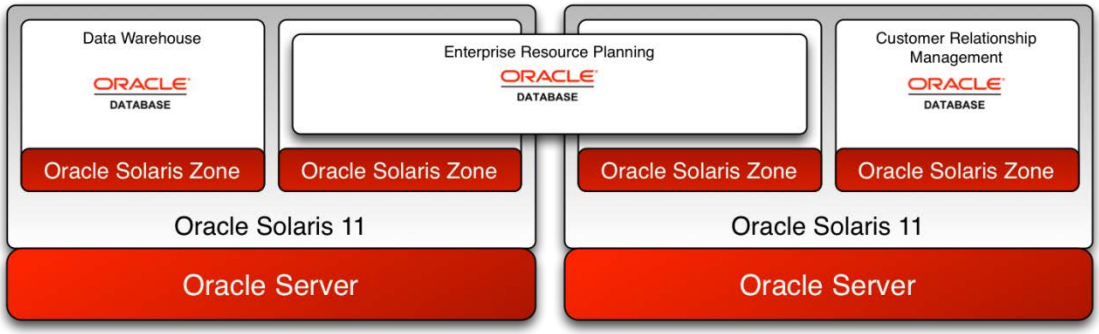

Figure 1. Encapsulating Databases in Oracle Solaris 11 Zones

In this deployment model, multiple zones are provisioned onto physical servers that are clustered together in a cloud pool, and each zone hosts an instance of an Oracle Database 11*g* Release 2 as Oracle Real Application Clusters (Oracle RAC) or as an Oracle RAC one-node database. While there are several options for allocating and sharing resources among zones, this paper will describe a model in which resources are partitioned, providing the strongest degree of isolation available with Oracle Solaris 11 Zones, enabling the consolidation of tenants who must be thoroughly segregated from each other.

Isolation in Database Clouds

There are four dimensions of inter-tenant isolation to consider in a consolidated environment. Those items and their requirements for strong inter-tenant isolation are as follows:

- **Faults**: While the failure of a shared component (such as a server) inevitably impacts all tenants, the failure of one tenant's software or dedicated resources should not impact other tenants.
- **Operational**: Replacement or maintenance of a shared resource may impact all tenants, but the operations performed on a specific tenant's environment should not impact other tenants.

[1] Virtual operating environments can be deployed in VMs, for example, Oracle Solaris 11 Zones running in Oracle VM Server for SPARC. Also, Oracle Solaris 11 supports Oracle Solaris 10 Zones, which are operating environments that emulate Oracle Solaris 10. This paper does not examine these deployment options.

- **Security**: Each tenant's data and runtime environment must be shielded from other tenants, and possibly isolated from the provider of the cloud environment (for example, a tenant in a public cloud should expect their data to be confidential—the provider cannot read it).
- **Resources**: Tenants need sufficient resources to meet their SLAs. Tenants are not able to consume resources that are not granted to them.

Database Isolation with Oracle Solaris 11 Zones

Deploying databases in Oracle Solaris 11 Zones addresses each type of isolation very effectively. The following tables describe the level of isolation that zones provide. In some cases, this is the same as when deploying natively on the operating system, but we will see many examples where zones provide stronger isolation, and we will see cases where deploying on Oracle SuperCluster provides added isolation.

Fault Isolation

TABLE 1. FAULT ISOLATION

FAILURE TYPE	SCOPE	IMPACT ON CO-TENANTS IN AN ORACLE SOLARIS 11 ZONES ENVIRONMENT
Hardware	Server	The failure of a cluster node (server) will affect all the database connections that exist on that node. This behavior is the same in native deployments. For this reason, databases are deployed in a cluster environment (Oracle RAC or Oracle RAC one-node environment) to guarantee continued availability of the database service.
	Network NIC or switch	A NIC can be shared by several zones or assigned to a specific zone. In either case, the failure of a single NIC usually has minor impact since a redundant NIC should be available as a best practice. The same is true for a failed switch: as a best practice there should be more than one switch, so a single switch failure should not cause a service outage for any database. Oracle RAC's Single Client Access Name (SCAN) should be configured on top of Oracle Solaris IP Multipathing (IPMP) to leverage the multiple NICs available for the public network, and Oracle RAC's Highly Available Virtual IP (HAIP) should be configured for Oracle RAC's private interconnect. For both, the impact of a NIC or switch failure will be the same in native deployments.
	Storage disk	Similar to a network, storage is also designed for resiliency. Any component failures are handled by a redundant partner. For added protection, use Oracle Automatic Storage Management redundancy to protect against disk failures or even storage failures. Behavior will be the same in native deployments.
Software	Database	A database failure on a native consolidation affects other databases in rare cases such as the following: if an Oracle RAC database instance becomes unresponsive, then one of the neighboring node's Lock Monitor Services (LMS) processes will request Cluster Synchronization Services (CSS) to perform a "member kill" operation against the unresponsive instance. In rare cases, where the unresponsive instance cannot be killed, an invasive operation such as node reboot is invoked. In these rare cases, other database instances will be affected since the entire server is rebooted.

		However when databases are deployed in zones, the reboot request will not reboot the entire server, but only the zone hosting the unresponsive instance. This reboot will be invisible to co-tenants—their zones will continue to operate without any impact from the zone reboot. A further benefit is that a zone reboot is much faster than a server reboot (seconds versus minutes). Therefore, the database in the rebooted zone will come back online faster than if the server were rebooted.
	Oracle Grid Infrastructure or Oracle Automatic Storage Management instance	Each database encapsulated in its own zone has its own Oracle Grid Infrastructure. Therefore, the failure of an Oracle Grid Infrastructure or Oracle Automatic Storage Management instance in a zone does not affect databases running in other zones. On a native deployment, there is one Oracle Grid Infrastructure instance shared by all databases on a node. Therefore, an Oracle Grid Infrastructure failure impacts all databases running on that node.
	Global Zone	The Oracle Solaris 11 global zone operates all of the zones running on the server, so a failure of the Oracle Solaris 11 global zone impacts all zones and database instances on the server. This impact is the same as in a native deployment, where all databases are hosted in the global zone.
	Zone hosting the database	A failure or reboot of a zone does not impact other zones or their tenants. Besides addressing the fault scenario described above, per-zone rebooting can be very useful in test environments. If a tester wishes to simulate a node reboot and observe its effect on the hosted application, the tester can reboot the zone without impacting any other zones on the server.
Administrative	Error or attack	A root user in a zone cannot execute any command that will impact any other zones: the zone is an administrative boundary. Since such privileges are not available, any accidental or malicious attempts to delete or alter other zones are prohibited. This provides a safety net that is not available on native deployments: per-zone administrators can be given full root privileges to their environments without endangering other environments. Zones can be created from templates (documented in the `zonecfg` man page). This enables consistent, repeatable zone deployments, thus reducing errors that could grant unintended privileges.

Operational Isolation

TABLE 2. OPERATIONAL ISOLATION

OPERATIONAL SCENARIO	IMPACT ON CO-TENANTS IN AN ORACLE SOLARIS 11 ZONES ENVIRONMENT
General administration	An administrator of a zone can operate only on that zone, within the administrative and resource limits set forth by the global zone administrator. The zone administrator will perceive a dedicated system and will be able to install and patch software, configure firewalls, monitor network traffic of the zone, and so on. This localized authority and visibility are the keys to enabling Oracle Solaris 11 Zones to be used as the foundation for clouds with effective, delegated administration: each zone owner has the scope to fully manage their environment. This is not possible in a native deployment since each administrator would need root privileges in the global zone and would be able to access system-wide resources. Note that root access to the global zone should be given to the smallest possible group of administrators since they will have visibility to all devices and file systems in non-global zones. Role Based Access Control (RBAC) allows taking this a step further by creating administrator accounts with only the minimum set of required privileges. New in Oracle Solaris 11, delegated zone administration offers the ability to grant users access to the global zone, but only for specific operations on specific zones. For example, an administrator could be given the ability to reboot their specific zone without any visibility of or control over other zones.
Oracle Grid Infrastructure updates	In this model, each database has its own Oracle Grid Infrastructure, and each database and Oracle Grid Infrastructure combination is in a dedicated zone, so the database and its Oracle Grid Infrastructure can be patched and managed together, and independently of any other database/Oracle Grid Infrastructure installations on the server. In contrast, if multiple databases are consolidated on a native platform, each database on a given node will share a single Oracle Grid Infrastructure home, and updating that home affects all the databases running on the node.
Oracle Home updates	In the model this paper describes, each zone hosts exactly one database and exactly one Oracle Home. Therefore, the two can be managed in tandem without impacting any other databases. In a native platform consolidation, a database can have its own Oracle Home or share an Oracle Home with other databases. If there are numerous databases with their own homes, management becomes more challenging, particularly if specific naming conventions are employed. If several databases share a home in a database consolidation, then patching the home will impact all databases running from that home. Also, any user who is part of the OSDBA group for that home will have SYSDBA access to all database instances running from that home. (Oracle Database Vault can be used to address this issue.)
OS updates—global zone	All zones are managed by the global zone. Therefore, any action that affects the global zone affects all zones and the databases running in them. For example, upgrading the global zone will affect every zone in the same manner. To minimize this impact, global zone updates are always made on an inactive boot environment. The active environment is not affected during this operation—all zones and the applications they host continue to operate without impact. The new environment becomes active after a single reboot of the server. (This is the same approach employed in native Oracle Solaris 11 deployments.)

OS updates—non-global zone	Each Oracle Solaris 11 Zone on a server runs at the same OS software update level. Therefore, the consolidated databases must all support the same Oracle Solaris 11 software update level; and when one zone is updated, all are updated. In effect, this brings the same "single OS version" requirement that a native deployment implies.
Updates on Oracle Solaris 11 systems—general	One advantage of Oracle Solaris 11 is the ability to quickly switch via reboot back to a pre-upgrade environment using an alternate boot environment (ABE). This applies to zone and non-zone Oracle Solaris 11 deployments.
Adding devices	A non-global zone must be rebooted to see new devices (such as a storage LUN or network interface) after they are assigned from the global zone. This requirement might be lifted in future releases of Oracle Solaris 11. Note that rebooting a zone does not impact other zones. Also, by running Oracle RAC, the impact of this reboot is mitigated, since services will continue running in the zone(s) of the Oracle RAC cluster that are not rebooted. Zones hosting Oracle Database 11*g* Release 2 on Oracle SuperCluster offer an advantage for storage management when Oracle Exadata Storage Servers are used. In this scenario, the Oracle Database 11*g* Release 2 instance doesn't manage "local" disks or devices presented via Oracle Solaris. Instead it communicates with Oracle Exadata Storage Servers with a special protocol over InfiniBand. Oracle Exadata Storage Servers manage the physical disks, so the disks are not seen directly by Oracle Solaris. In this environment, additional storage can be presented to the zone and the Oracle Database 11*g* Release 2 instance without rebooting either. On all Oracle Solaris platforms, the global zone sees new devices dynamically, with no reboot needed.
Backup/recovery of zone contents (including Oracle Home)	Backups can be done at the zone level by running a backup client in each zone. If the backup data from each database must be kept separate from the others, then each backup client will connect to its own file system. If the backup data for each database can be stored in a single file system, each zone could mount a single shared file system provided by the global zone. This reduces the number of file systems to manage (though note that the data from each zone is not isolated in this case). If the entire backup activity can be centralized, backups can be controlled from the global zone and the backup data written to backup devices that are connected to and visible in the global zone only. Compared to a per-zone backup architecture, this reduces complexity and can lead to significant savings of bandwidth. Recovery of a zone is much faster than recovering a system, because it involves simply an "unpack" of an archived zone installation.
Migrating a zone to another system	A zone can be cold-migrated to a zone on another physical system. The zone on the source machine is halted and detached before being attached on the target server. The source and target servers can be running different releases of Oracle Solaris 11; the target system must have the same or later versions of the operating system packages. See the "Migrating Oracle Solaris Systems and Migrating Non-Global Zones" chapter of the "*Oracle Solaris Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*" guide.

Security Isolation

The security isolation provided by Oracle Solaris Zones applies to both administrative and runtime considerations. Oracle Solaris 10 Zones has been evaluated under the Common Criteria at Evaluation Assurance Level 4+. Oracle Solaris 11 Zones technology is currently under evaluation.[2]

TABLE 3. SECURITY ISOLATION

SECURITY ISOLATION USE CASE	SECURITY HANDLING WITH ORACLE SOLARIS 11 ZONES
Administrative privileges in a zone	Among other default limitations, the zone root user cannot create devices; other limitations may be defined when the zone is created. These privileges may be modified—either relaxed or further restricted—by the global zone administrator after zone creation. After such a change, the zone must be rebooted for the (persistent) change to take effect. Because of the reduced privileges for a zone root user, many customers consider a zone a safer database operating environment than the global zone, since this "restricted root admin" concept can be applied. Note that root access to the global zone should be given to the smallest possible group of administrators since they will have visibility to all devices and file systems in non-global zones. Additionally, RBAC should be used to provide administrative granularity.
File/storage access	Oracle Database Vault should be used to protect client data, so that even the cloud provider cannot read the client data. This applies to native deployments as well.
Network traffic security	In most data centers, network switches will be shared by NICs from multiple server pools. Point-to-point traffic between associated NICs can be isolated by assigning them to dedicated VLANs (virtual LANs). This is a common best practice even in non-cloud deployments. Oracle Solaris 11 provides the ability to create VNICs (virtual network interfaces). A physical NIC may host one or more VNICs. A VNIC can be assigned to a VLAN, and this isolates the end-to-end traffic between VNICs on that VLAN. When a VLAN is assigned to a non-global zone's VNIC, the traffic is fully isolated and other non-global zones cannot see that traffic. So if all non-global zones on a node are given their own VNICs, and those VNICs are grouped in properly segregated VLANs, then a root user in any given zone will be able to see only the traffic routed to that zone. For example, a root user in a zone could run `snoop` and would see only the traffic routed to that zone. By default, the zone administrator will have the ability to spoof MAC and IP addresses. The global zone administrator can prevent this by setting the "allowed-address" property on the zone. Also, the Data Link Protection feature can be used in the global zone to manage the protocols that can be sent by a non-global zone.

[2] See http://www.oracle.com/technetwork/topics/security/security-evaluations-099357.html for more details.

Resource Isolation

Table 4 describes how to achieve resource isolation with zones. Later sections describe how to change allocations and observe resource usage.

TABLE 4. RESOURCE ISOLATION

RESOURCE	RESOURCE ISOLATION WITH ORACLE SOLARIS 11 ZONES
Memory	The deployment model described in this paper assumes that every database on the system could be active at the same time. Therefore, we will allocate memory in a manner that will prevent memory paging, which would cause performance issues. Memory use by a zone should be limited by setting a maximum value on the amount of virtual memory that can be allocated by processes in the zone. The global zone administrator defines the virtual memory available to each zone at zone creation with the "swap" property of the "capped-memory" resource. This defines the maximum amount of virtual memory that the zone will be allowed to use. To prevent paging, we must ensure that there is enough physical memory to handle all of the zones simultaneously. First, we will reserve 4 GB of memory for global zone operations; 4 GB for the ZFS ARC cache [3]; and 1/32 of the total physical memory for free pages. So on a system with 512 GB, we would have 512 – (4+4+16) = 488 GB to allocate to zones [4]. Remember that we will limit the virtual memory of each zone; that is, the sum of all allocated virtual memory must be no more than 488 GB in our example. We do not limit physical memory for each zone in this model. In addition to predictable, optimal memory use, limiting virtual memory has an added benefit: if a DBA (accidentally or otherwise) assigns SGA/PGA values that exceed the zone's virtual memory (swap) limit, the database will not start. Without this limit, a DBA could "steal" memory from other zones, impacting other users and workloads. This control is not available for managing applications running in the global zone. Note: Oracle Solaris Dynamic Intimate Shared Memory (DISM) should not be used with the Oracle database. Because of this, changes to the SGA size are not dynamic as noted in the Resource Allocation Changes table (Table 5).

[3] Although ZFS is not used as the data store, it is always active in an Oracle Solaris 11 environment. If the ARC cache is not limited, it will occupy a large amount of memory during system startup. This will delay startup of the databases. Hence, the recommended limit on ZFS ARC cache.

[4] To fine-tune the amount of memory reserved for the global zone, the `zonestat` utility can be used to show how much memory the global zone is using. That memory will be listed as `system`. This value does not include the ZFS ARC.

CPU	Before allocating CPU capacity to non-global zones, the amount of CPU that will be reserved for the global zone must be considered. Although most of the processing for a non-global zone is performed by the CPUs assigned to the zone, the global zone manages the overall system including I/O. Assigning one core (from a SPARC T-Series or SPARC M-Series server from Oracle) to the global zone will typically be sufficient. However, if applications are deployed in the global zone (not recommended) or the deployment has high I/O, the CPU needed by the global zone may be higher. For assigning CPUs to non-global zones in this deployment model, CPU resources are allocated to a zone by creating a resource pool; assigning a processor set to the resource pool; and attaching the resource pool to the zone. On a hyper-threaded server, the CPUs (that is, threads) assigned to a zone should be from the same socket. For a zone running Oracle Database, it is not necessary to assign threads from the same core of a SPARC T-Series machine. Threads on the same socket provide sufficient proximity since they have the same distance from the common memory. This approach guarantees the processing capacity available for the zone. Also note that this approach is recognized by Oracle as a hard partition for software licensing[5][6]. In a native deployment, CPUs can be allocated using Oracle Solaris Projects or, preferably, managed with the database's Instance Caging feature.
LWPs	The maximum number of lightweight processes should be limited for the zone. This will prevent system-wide impact from an administrative error or misbehaving application that creates very large numbers of processes or LWPs: if the maximum number of processes or LWPs for the system is reached, the global process table of the global zone will fill up and no more processes can be created by any user or processes within a zone. To determine the proper value, the zone should be observed over time to monitor the process behavior. A value should be selected that will provide headroom and prevent excessive use. For a zone running Oracle Database 11*g* Release 2, a typical value will be in the range of 5,000 to 10,000. In a native deployment, processes can be limited using Oracle Solaris Projects or, preferably, managed with the database's Instance Caging feature.
Processes	As with LWPs, processes should be observed on a well-behaving system, and an appropriate maximum applied for the zone.

[5] See http://www.oracle.com/technetwork/articles/systems-hardware-architecture/deploying-oracle-database-solaris-168405.pdf for specifics and http://www.oracle.com/us/corporate/pricing/specialty-topics/index.html for additional licensing topics relevant to database cloud deployments.

[6] There are other approaches available for allocating and sharing CPUs among zones, but they are not recognized by Oracle as hard partitions for software licensing.

Network traffic	The traffic of each VNIC is segregated from other VNICs down to the data link layer; therefore, when a VNIC is assigned to a zone, that zone owns what appears to be a physical link. The zone's administrator can apply IP operations such as configuring IP Filter and IPsec. The *Oracle Solaris 11 Network Interfaces and Network Virtualization* administration guide (see the section "For More Information" for a pointer to this guide) refers to this as a basic virtual network. It is one of the key Oracle Solaris 11 features enabling the creation of virtual private clouds, since zone administrators have greater flexibility but remain isolated from other zones on the system. Each VNIC can be assigned a bandwidth limit by the global zone administrator with the `maxbw` property. This can be used to ensure that no individual non-global zone will be able to consume all of an interface's bandwidth. One consideration when deploying Oracle Database 11*g* Release 2 and Oracle RAC in zones is the use of public IP addresses. The consumption of IP addresses is a potential concern because some data centers have a notional cost associated with IP addresses, because they are scarce resources in an IPv4 deployment. When deploying Oracle Database 11*g* Release 2 RAC in Oracle Solaris 11 zones, you will need: • A host IP address for each zone • A public VIP (virtual IP) address for each zone (because each node of an Oracle Grid Infrastructure cluster needs a public VIP, and in this deployment model each zone is equivalent to a node) • A SCAN (Single Client Access Name), which resolves to three VIP addresses So for a two-node Oracle RAC cluster occupying one zone on the first server and one zone on the second, that would be seven public addresses: 2 nodes * (1 IP address per node) + 2 nodes * (1 public VIP address per node) + 3 SCAN VIPs. Note that you can configure a zone without a physical address, which would reduce the number back to five. In this scenario, communication with the zone takes place through the cluster node public VIP address provided by the database. Note that all of the above applies to IP traffic on Ethernet interfaces and also to IPoIB traffic on InfiniBand interfaces if the IB fabric supports multiple IB partitions. Also note that these capabilities assume the use of the exclusive-IP stack for the zone. This is the default option. The alternative is shared-IP, but that model does not support the segregation of each zone's traffic and is not addressed by this paper. VNICs are available in native deployments also.
Storage IOPs or capacity between databases	Storage IOPs or bandwidth can be limited or managed on Oracle Exadata and Oracle SuperCluster platforms using the Exadata IO Resource Manager feature. This is available to database instances running both in the global zone and in local zones. When deploying on non-engineered systems with IB fabrics, or on SAN or NAS storage, Exadata IO Resource Manager is not available and limiting storage bandwidth must be hand-crafted to accommodate the storage device. Some devices such as Oracle's Pillar Axiom offer better controls than others, but this is beyond the scope of this paper.

Resource Allocation Changes

After resources have been allocated to zones and their databases, those resource allocations can be adjusted as described in Table 5.

TABLE 5. RESOURCE ALLOCATION CHANGES

RESOURCE	RESOURCE MANAGEMENT WITH ORACLE SOLARIS 11 ZONES
Memory	A zone's virtual memory limit can be adjusted dynamically by the global zone administrator using the command `prctl`. For the change to persist across a zone/server reboot, the change must also be made with the `zonecfg` command. The database's SGA and PGA memory allocations do not automatically adjust if the zone's memory limit is changed. They must be updated manually. The database must be restarted to recognize the new SGA/PGA setting. This behavior is the same on a native deployment.
CPU	If the database is started without enabling Instance Caging, the database will ask the OS how many CPUs are available, and will assume it can use all of them. In the deployment model described in this white paper, the OS will report the number of CPUs (or threads, if running on a multi-threaded processor) that were bound to the zone via assigning a processor set to the zone's resource pool. When the database has been started without enabling Instance Caging, adjusting CPUs allocated to a zone will be automatically detected by the database because the database checks the OS periodically for the CPU count and adapts itself accordingly. To adjust the CPUs allocated to a zone in the processor set model, the global zone administrator executes the `poolcfg -dc` command to transfer processors between active pools/processor sets (psets) on the system. Changes made with `poolcfg -dc` are effective immediately. Be aware that CPUs cannot be assigned in violation of the minimum and maximum processors assigned to a pset (`pset.min` and `pset.max`), so moving CPUs may require adjusting the min/max for the pool (performed with `poolcfg`). The change will not persist across a reboot. For persistent changes, update `pset.min` and `pset.max` in the persistent configuration (`poolcfg` without `-d`). The persistent configuration only records min and max for each processor set. For specific CPU assignments (for example, CPU 0, 1, 2, and 3 assigned to a zone), then do not use the pools persistent configuration at all. Instead, create a script that executes `poolcfg -dc` on every boot. An Oracle Solaris Service Management Facility service or `rc` script would accomplish this. **Note**: Instance Caging and Database Resource Manager are not necessary in this deployment architecture. If multiple databases are hosted in one zone, Instance Caging and DBRM may come into play. This might be discussed in a later version of this paper. (In a native deployment, those are the key technologies used for managing resources in a consolidation deployment.) Also note that if Instance Caging is enabled, the database uses CPU_COUNT as a heuristic for deciding things such as how many foreground processes to spawn. In this scenario, changing the CPUs assigned to the zone will not be noticed by the database until CPU_COUNT is manually changed, at which time the database adapts to the new value. This could be useful if the database were co-located with other databases or applications in the zone. However, we are dedicating a single zone for each database for maximum isolation. Therefore, the recommendation for this deployment model is to not enable Instance Caging.

Network traffic	The bandwidth of a VNIC, and therefore of the zone's traffic, can be changed dynamically by updating the VNIC's `maxbw` property with the `dladm` command. This works the same in native deployments.

Resource Metering

In order for cloud providers to charge their customers for resource usage, providers must be able to meter the consumption of the resources for which they wish to charge. Oracle Solaris 11 offers Extended Accounting, which collects statistics of resource use (including CPU and memory) by each process in a zone.

Monitoring network consumption of a VNIC assigned to a zone is done with the `dlstat` command and extensions to `dladm` and `netstat`.

Zones are not required for using Extended Accounting or network traffic analysis; those features are available on native deployments also.

Zones can be further analyzed with the `zonestat` tool and the `libzonestat` library (for application developers).

Conclusion

Cloud providers have several architectural options for offering database as a service (DBaaS). When services will be provided to tenants who must be isolated from each other, a tenant is isolated by encapsulation.

Encapsulation may be physical or logical. Some use cases will require complete physical isolation (dedicated hardware). Common use cases can leverage the strong isolation provided by virtualization technologies to host multiple tenants on shared hardware and software environments. This enables higher utilization of the shared resources, with fewer separate environments to manage and monitor.

Oracle Solaris 11 Zones provide a no-cost, high-performance encapsulation solution that enforces very strong application and administrative isolation. They are an ideal platform for building clouds that consolidate multiple tenants who must be isolated from each other.

For More Information

TABLE 6. RESOURCES

TOPIC/TITLE	URL
Database consolidation onto private clouds	http://www.oracle.com/us/products/database/database-private-cloud-wp-360048.pdf
"Best Practices for Database Consolidation in Private Clouds"	http://www.oracle.com//technetwork/database/focus-areas/database-cloud/database-cons-best-practices-1561461.pdf
Oracle Solaris 11 virtualization	http://www.oracle.com/technetwork/server-storage/solaris11/technologies/virtualization-306056.html
Oracle Solaris 11 how-to guides	http://www.oracle.com/technetwork/server-storage/solaris11/documentation/how-to-517481.html
Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management	http://docs.oracle.com/cd/E23824_01/html/821-1460/index.html
Network interfaces and network virtualization	http://docs.oracle.com/cd/E23824_01/html/821-1458/index.html
Oracle Solaris 11 networking	http://www.oracle.com/technetwork/server-storage/solaris11/documentation/o11-137-s11-net-virt-mgmt-525114.pdf
Application traffic restriction on Oracle Solaris 11	http://www.oracle.com/technetwork/articles/servers-storage-admin/o11-095-s11-app-traffic-525038.html
Supported virtualization partitioning technologies for Oracle Database and Oracle RAC product releases	http://www.oracle.com/technetwork/database/virtualizationmatrix-172995.html
"Secure Database Consolidation Using the SPARC SuperCluster T4-4 Platform"	http://www.oracle.com/technetwork/server-storage/sun-sparc-enterprise/documentation/o12-087-1878511.pdf
"Best Practices for Deploying Oracle Solaris Zones with Oracle Database 11*g* on SPARC SuperCluster"	http://www.oracle.com/technetwork/server-storage/engineered-systems/sparc-supercluster/deploying-zones-11gr2-supercluster-1875864.pdf
Deploying an enterprise database cloud using Oracle SuperCluster	http://www.oracle.com/technetwork/server-storage/engineered-systems/sparc-supercluster/db-cloud-using-sparc-supercluster-1875862.pdf

Appendix A: Oracle Solaris 10 Containers

The Oracle Solaris 10 operating system introduced Oracle Solaris Containers, which evolved into Oracle Solaris 11 Zones. Oracle Solaris Containers are widely adopted for test, development, and production deployments. Because of the popularity and longevity of this solution, there is a considerable body of collateral available and much of it will be of interest when working with Oracle Solaris 11 Zones. The following table lists several Oracle Solaris 10 documents that may be useful references when preparing to deploy Oracle Database in Oracle Solaris 11 Zones.

TABLE 7. ORACLE SOLARIS 10 RESOURCES

TOPIC/TITLE	URL
"Deploying Oracle Database on the Solaris Platform – An Introduction"	http://www.oracle.com/technetwork/articles/systems-hardware-architecture/deploying-oracle-database-solaris-168405.pdf
"Best Practices for Running Oracle Databases in Oracle Solaris Containers"	http://www.oracle.com/technetwork/server-storage/solaris/solaris-oracle-db-wp-168019.pdf
"Best Practices for Deploying Oracle RAC Inside Oracle Solaris Containers"	http://www.oracle.com/technetwork/articles/systems-hardware-architecture/deploying-rac-in-containers-168438.pdf
"Highly Available and Scalable Oracle RAC Networking with Oracle Solaris 10 IPMP"	http://www.oracle.com/technetwork/articles/systems-hardware-architecture/ha-rac-networking-ipmp-168440.pdf
Oracle Solaris 10 Container guide	https://blogs.oracle.com/solarium/entry/new_version_container_guide_3

Encapsulating Oracle Databases with Oracle Solaris 11 Zones
October 2013, Revision 1.3
Author: Burt Clouse
Contributing Authors:Troy Anthony, David Brean, Glenn Brunette, Detlef Drewanz, Nicolas Droux, Ulrich Graef, Raj Kammend , Stephen Lawrence, Mikel Manitius, Michael Ramchand, Tim Read, Sebastian Solbach, Hartmut Streppel, and Nitin Vengurlekar

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com

Oracle is committed to developing practices and products that help protect the environment

Hardware and Software, Engineered to Work Together