SUN ZFS STORAGE APPLIANCE

An Oracle White Paper
Updated March 2013

Protecting Oracle Exadata with the Sun ZFS Storage Appliance: Configuration Best Practices

Introduction

Database, system, and storage administrators are faced with a common dilemma when it comes to backup and recovery of databases—how to back up more data, more often, in less time, and within the same budget. Moreover, practical challenges associated with real-world outages mandate that data protection systems be simple and reliable to ensure smooth operation under compromised conditions. The Sun ZFS Storage Appliance from Oracle helps administrators meet these challenges by providing a cost-effective and high-bandwidth storage system that combines the simplicity of the Network File System (NFS) protocol with ZFS-enhanced disk reliability. Through Sun ZFS Storage Appliance technology, administrators can reduce the capital and operational costs associated with data protection while maintaining strict service level agreements with end customers.

The Sun ZFS Storage Appliance is an easy-to-deploy unified storage system uniquely suited for protecting data contained in general-purpose Oracle databases and the Oracle Exadata Database Machine. With native QDR InfiniBand (IB) and 10 gigabit (Gb) Ethernet connectivity, the Sun ZFS Storage Appliance is an ideal match for Oracle Exadata. These high-bandwidth interconnects reduce backup and recovery time, as well as reduce backup application licensing and support fees, compared to traditional NAS storage systems. When combined with the incremental update backup technology of the Oracle 11*g*R2 Recovery Manager (Oracle RMAN), Sun ZFS Storage Appliance storage solutions deliver increases in storage efficiency that can further reduce recovery time and simplify system administration.

When deploying a Sun ZFS Storage Appliance for backup and recovery of Oracle databases, backup window and recovery time objectives (RTO) must be met to ensure timely recovery in the event of a disaster. This paper describes best practices for setting up the Sun ZFS Storage Appliance for optimal backup and recovery of Oracle databases and includes specific tuning guidelines for Oracle Exadata. Summary tables describe the capacity and throughput of a Sun ZFS Storage Appliance to help system planners identify the most appropriate system for meeting their specific needs.

This paper addresses the following topics:

- Overview of Oracle Exadata with the Sun ZFS Storage Appliance
- Data protection with the Sun ZFS Storage Appliance
- Best practices for configuring the Sun ZFS Storage Appliance for database backup and recovery
- Implementation guidelines for using the Sun ZFS Storage Appliance with Oracle Exadata

For customers seeking maximum system utilization and efficiency, additional protocol choices, including Fibre Channel (FC) and 1 Gb Ethernet connectivity, allow unprecedented flexibility and control for sharing information. By employing Sun ZFS Storage Appliance snapshot and cloning capabilities, database backups can be used for more value-added work and provide system administrators with additional recovery options. For example, snapshot deployment of database clones provides development, quality assurance (QA), and other types of organization with up-to-date copies of production data, while these additional copies add yet another safeguard in the unlikely event of an outage. For details about database cloning with the Sun ZFS Storage Appliance see the “Database Cloning using Oracle Sun ZFS Storage Appliance and Oracle Data Guard” white paper at http://www.oracle.com/technetwork/database/features/availability/maa-db-clone-szfssa-172997.pdf.

Overview of Oracle Exadata with the Sun ZFS Storage Appliance

Oracle Exadata is a complete package of software, servers, storage, and networking, including a high-bandwidth private IB network. Oracle Exadata includes the following technology to optimize data protection systems based on the Sun ZFS Storage Appliance:

- QDR IB fabric provides 40 Gb of bandwidth per port between the database servers, storage cells, and the Sun ZFS Storage Appliance.
- Oracle Recovery Manager (Oracle RMAN) is a native backup and recovery tool for Oracle databases that simplifies backup and recovery administration to the Sun ZFS Storage Appliance.
- Oracle Direct NFS (dNFS) is an optimized NFS client for Oracle databases that provides a high-bandwidth solution for data transfer with the Sun ZFS Storage Appliance.
- Backup and restore operations can be automatically parallelized across all database nodes, Oracle Exadata storage cells, and Sun ZFS Storage Appliance interfaces and controllers for maximum scalability and throughput.
- Oracle RMAN incrementally updated backups allow incremental backups to a Sun ZFS Storage Appliance to be converted to full backups to reduce restore time and efficiently refresh and deploy snapshots and clones of production data.
- An Oracle RMAN clone can produce a complete copy of any database stored in Oracle Exadata on the Sun ZFS Storage Appliance.

The combined features of Oracle Exadata and the Sun ZFS Storage Appliance provide IT administrators with advanced tools to meet nearly arbitrary recovery point objectives (RPO) and recovery time objectives (RTO) and ensure business continuity requirements are met.

Configuration Best Practices

This section describes best practices for choosing a performance-optimized and fault-tolerant configuration suitable for protecting data contained in an Oracle Exadata Database Machine.

Configuring the Hardware

This section describes the hardware bill of materials and software configuration settings needed to meet the throughput results shown in the tables below.

The hardware configuration includes:

- *Head:* Clustered Sun ZFS Storage 74x0 appliance with a minimum of four CPU sockets and 512 GB DRAM per head.
- *Front-end network:* A minimum of two IB host channel adapters (HCAs) per head.
- *Back-end SAS configuration*:
    - Two SAS 2 host bus adapters (HBAs) per head for a 2-tray or 4-tray configuration.
    - Three SAS 2 HBAs per head for a 6-tray configuration.
    - Four SAS 2 HBAs per head for an 8-tray configuration.
- *Write flash configuration:* Four write-optimized flash devices for configurations supporting clone operations and incrementally applied backups. Write-optimized flash may be omitted for systems that support only backup and restore of backup sets.
- *Number of trays:* Two to eight depending on throughput and capacity requirements.
- *Drive size:* 2 TB or 3 TB based on capacity requirements.
- *Read flash configuration*: Optionally, two to four read-optimized flash devices per head for systems that support additional processing, such as development or QA.

Usable capacity depends on the number of drive trays configured in the Sun ZFS Storage 74x0 appliance. Table 1 shows approximate raw and usable capacity for RAID-Z protection with 2, 4, 6, and 8 trays implemented with 2 TB disk drives. Usable capacity can be doubled by using 3TB disk drives.

TABLE 1. APPROXIMATE RAW AND USABLE CAPACITY FOR 1 TB DISK DRIVE CONFIGURATIONS INCLUDING FOUR WRITE-OPTIMIZED FLASH DEVICES.

NUMBER OF TRAYS	USABLE CAPACITY
2	27
4	59
6	86
8	118

System throughput depends on the hardware and software configuration. Important hardware parameters include the number of Sun ZFS Storage Appliance heads and drive trays. Important software parameters include the number of Sun ZFS Storage Appliance pools and shares as well as the number of Oracle RMAN channels.

A typical Sun ZFS Storage Appliance Oracle RMAN configuration would include 2 heads, 4 trays, 2 pools. The number of active Oracle RMAN channels configured would be dependent of the administration and performance demands. Typically, the use of one share per pool would configured for a management-optimized system. The use of eight shares would be configured performance-optimized system.

Configuring Pools, Shares, and Networks

The Sun ZFS Storage Appliance is a general-purpose system designed to support a wide range of data access requirements and workloads. Best practices for optimizing a Sun ZFS Storage Appliance pool, share, and network configuration to support backup and restore processing are summarized in the following sections.

Pool Configuration

This section describes design considerations to determine the most appropriate pool configuration for the Sun ZFS Storage Appliance for Oracle RMAN backup and restore operations based on data protection and performance requirements. The system planner should consider pool protection based on the following guidelines:

- Use parity-based protection for general-purpose and capacity-optimized systems:
  - RAID-Z for protection from single-drive failure on systems subject to random workloads
  - RAID-Z2 for protection from two-drive failure on systems with streaming workloads only
- Use mirroring for high-performance with incrementally applied backup.
- Configure pools based on performance requirements:
  - Configure a single pool for management-optimized systems.
  - Configure two pools for performance-optimized systems. Two-pool systems can be configured by using half the drives from each tray.
- Configure log device protection:
  - Stripe log devices for RAID-Z and mirrored pool configurations.
  - Mirror log devices for RAID-Z2 pool configurations.

Share Configuration

The default options for Sun ZFS Storage Appliance shares provide a good starting point for general-purpose workloads. Sun ZFS Storage Appliance shares can be optimized for Oracle RMAN backup and restore operations as follows:

- Create a project to store all shares related to backup and recovery of a single database. For a two-pool implementation, create two projects, one for each pool.
- Configure the shares supporting Oracle RMAN backup and estore workloads with the following values:
  - Database record size (`recordsize`): **128 kB**
  - Synchronous write bias (`logbias`): **Throughput** (for processing backup sets and image copies) or **Latency** (for incrementally applied backups)

- Cache device usage (`secondary cache`): **None** (for backup sets) or **All** (when supporting incrementally applied backups or database clone operations)
- Data compression (`compression`): **Off** for performance-optimized systems, **LZJB** for capacity-optimized systems
- Number of shares per pool: **1** for management-optimized systems, **8** for performance-optimized systems

Additional share configuration options, such as `gzip` compression or replication, can be applied to shares used to support Oracle Exadata backup and restore, as customer requirements mandate.

Customers implementing additional Sun ZFS Storage Appliance data services should consider implementation-specific testing to verify the implications of deviations from the practices described above.

Network Configuration

This section describes where to install the IB HCAs in the Sun ZFS Storage Appliance chassis, how to cable the HCAs to the Oracle Exadata IB switch, how to configure the IP network multipathing (IPMP) groups, and how to configure routing in the Sun ZFS Storage Appliance. The basic network configuration steps are as follows:

1. Install IB HCAs in PCI slots 4 and 5. Additional HCAs can be installed in slots 3 and 6.
2. Connect the *upper* port of each HCA into one of the following ports of the *upper* leaf switch: 5B, 6A, 6B, 7A, 7B, or 12A.
3. Connect the *lower* port of each HCA into one of the following ports of the *lower* leaf switch: 5B, 6A, 6B, 7A, 7B, or 12A.
4. Configure `ibp0`, `ibp1`, `ibp2`, and `ibp3` with address 0.0.0.0/8 (necessary for IPMP), connected mode, and partition key (the default is `ffff`). To identify the partition key used by the Oracle Exadata system, run the following command as the root user:
   ```
   # cat /sys/class/net/ib0/pkey
   ```
5. Configure the active/passive IPMP group over `ibd0` and `ibd3` with `ibd0` active and `ibd3` passive.
6. Configure the active/passive IPMP group over `ibd1` and `ibd2` with `ibd2` active and `ibd1` passive.
7. Enable adaptive routing to ensure traffic is load balanced appropriately when multiple IP addresses on the same subnet are owned by the same head. This occurs after a cluster failover.

For customers seeking additional IB connectivity, more IB HCAs can be installed and configured. For details, see the *Sun ZFS Storage 7x20 Appliance Installation Guide* at http://download.oracle.com/docs/cd/E22471_01/html/821-1673/index.html.

IB partition keys for the additional HCAs can be supplied by the administrator of the IB network. The principles in this section can be applied to a 10 Gb Ethernet implementation by applying the network configuration to the `ixgbe` interfaces instead of the `ibp` interfaces.

Configuring Oracle RMAN and the Oracle Database Instance

Oracle RMAN is an essential component for protecting the content of Oracle Exadata. Oracle RMAN can be used to create backup sets, image copies, and incrementally updated backups of Oracle Exadata content on Sun ZFS Storage Appliances. To optimize performance of Oracle RMAN backups from Oracle Exadata to a Sun ZFS Storage Appliance, the database administrator should apply the following best practices:

- Load balance Oracle RMAN channels evenly across the nodes of the database machine.
- Load balance Oracle RMAN channels evenly across Sun ZFS Storage Appliance shares and controllers.
- Run two to four Oracle RMAN channels per drive tray.

To optimize buffering of the Oracle RMAN channel to the Sun ZFS Storage Appliance, you can tune the values of several hidden instance parameters. For Oracle Database 11*g* Release 2, the following parameters can be tuned:

- For backup set and backup and restore: `_backup_disk_bufcnt=64` and `_backup_disk_bufsz=1048576`
- For image copy backup and restore: `_backup_file_bufcnt=64` and `_backup_file_bufsz=1048576`

For additional information about tuning these parameters and tuning equivalent parameters for earlier versions of the Oracle Database software, see Article ID 1072545.1: *RMAN Performance Tuning Using Buffer Memory Parameters*) at http://support.oracle.com.

Oracle Direct NFS (dNFS) is a high-performance NFS client that delivers exceptional performance for Oracle RMAN backup and restore operations. dNFS should be configured for customers seeking maximum throughput for backup and restore operations.

Detailed Implementation Guidelines

This section provides detailed instructions for implementing a Sun ZFS Storage Appliance for backup and recovery of Oracle database systems, including Oracle Exadata Database Machines.

Configuring the Sun ZFS Storage Appliance Network

Sun ZFS Storage Appliance network configuration steps include assigning IP addresses and, optionally, IPMP groups to the physical network interface cards (NICs). For maximum throughput, set the Link Mode to **Connected Mode** for IB interfaces and select **Use Jumbo Frames** for 10 Gb Ethernet. See the screens for these settings in Figure 1 and Figure 2 in *Appendix A. Configuring the Sun ZFS Storage Appliance*.

Configure a minimum of two NICs in each head. Depending on the specific requirements for fault tolerance, IB ports can be configured with fixed IP addresses that can be failed over between cluster controllers, or IPMP, which allows IP addresses to be failed over between ports on a single controller. In practice, the bandwidth of the PCI-2 slot is only able to support the data rate of a single QDR IB port, so a practical IPMP configuration for InfiniBand ports is active/standby access between the ports of a specific card. Figure 3, Figure 4, Figure 5, Figure 6, and Figure 7 in *Appendix A* show the screens used for this aspect of the network configuration.

Configure adaptive routing to allow for data to be returned on the same interface from which it was requested, as shown in Figure 8 in *Appendix A*.

NOTE: Depending on how the Sun ZFS Storage Appliance is installed and configured, the administrator may also need to add a static route to the 1 GB management interface used by the cluster peer.

Configuring the Sun ZFS Storage Appliance Storage Pool

Pool configuration assigns physical disk drive resources to logical storage pools for backup data storage. To maximize system throughput, configure two equally sized storage pools by assigning half of the physical drives in each drive tray to each storage pool, as shown in Figure 9 in *Appendix A. Configuring the Sun ZFS Storage Appliance*.

NOTE: The Sun ZFS Storage Appliance management software presents a warning message about efficiency when two pools with the same RAID protection profile are configured. This message can be safely ignored when configuring for a high-performance Oracle RMAN backup solution.

Configuring the Sun ZFS Storage Appliance Cluster

Cluster configuration assigns physical network and pool resources to specific controllers in the Sun ZFS Storage Appliance cluster. An important concept in Sun ZFS Storage Appliance clustering is that a single head owns a specific physical resource at any given time. Consequently, ownership of IP addresses used to access a specific head must be correlated with the pool and network interfaces that are also owned by that controller.

For example, a Sun ZFS Storage Appliance cluster with two IP addresses on an IPMP group, four IB ports, and two pools should be configured such that each cluster node owns one of the two IPMP groups, the requisite interface ports to support the group, and one of the two pools. Figure 10 in *Appendix A. Configuring the Sun ZFS Storage Appliance* shows a cluster configuration on the Sun ZFS Storage Appliance. Client access to a specific share on a specific pool must be executed using the IP address associated with the pool.

Configuring the Sun ZFS Storage Appliance Share

Share configuration is the process of setting up and tuning NFS mount points for client access. For implementations with two pools (for example, `pool-1` and `pool-2`), create two separate projects, `dbname-1` and `dbname-2`, in `pool-1` and `pool-2`, respectively.

A project is a Sun ZFS Storage Appliance entity that provides a higher-level management interface point for a collection of shares. To optimize share management, update the default mount point for shares contained in the project to reference the database name, such as `/export/dbname`. If two projects support the same database, the names of the shares contained in each project must be different. For example, if `dbname-1` contains two shares, they could be configured as `backup1` and `backup3`, while the corresponding two shares in `dbname-2` could be configured as `backup2` and `backup4`.

For a performance-optimized system, create eight shares for each project. For shares supporting streaming backup and restore operations, set the Database record size property to **128 kB** and the Synchronous write bias property to **Throughput**. Figure 11 in *Appendix A. Configuring the Sun ZFS Storage Appliance* shows a sample share configuration that meets these specifications.

Root access can be granted for all database nodes in an Oracle Exadata rack by specifying a specific NFS access rule to allow root access for any server on the Oracle Exadata private network, for example, 192.168.36.0/22. Figure 12 in *Appendix A* shows an example of NFS root-access exceptions configured for a specific project.

Configuring the Sun ZFS Storage Appliance DTrace Analytics

The Sun ZFS Storage Appliance includes a comprehensive performance analysis tool called DTrace Analytics. DTrace Analytics is a framework that monitors important subsystem performance accounting statistics. A subset of the available accounting statistics should be monitored to provide comprehensive data on the effectiveness and performance of Oracle RMAN backup and restore workloads.

The following analytics are available when advanced analytics are configured on the Sun ZFS Storage Appliance (Configuration > Preferences > Enable Advanced Analytics):

- *CPU:* Percent utilization broken down by CPU mode
- *Disk:* Average number of I/O operations broken down by state of operation
- *Disk:* I/O bytes per second broken down by type of operation
- *Disk:* I/O operations per second broken down by latency
- *Disk:* Disks with utilization of at least 95 percent broken down by disk
- *Network:* Interface bytes per second broken down by direction
- *Network:* Interface bytes per second broken down by interface
- *Protocol:* NFSv3 operations per second broken down by size

- *Protocol:* NFSv3 operations per second broken down by type of operation
- *Protocol:* NFSv3 operations per second of type read broken down by latency
- *Protocol:* NFSv3 operations per second of type write broken down by latency
- *Protocol:* NFSv3 operations per second of type read broken down by size
- *Protocol:* NFSv3 operations per second of type write broken down by size

Implementing these accounting statistics helps end users gain a quantitative understanding of the instantaneous and historical resource consumption and quality of service (QoS) for their specific implementation.

Configuring the Client NFS Mount

When configuring the Sun ZFS Storage Appliance, any server that accesses the appliance, including Oracle Exadata servers, is considered a client. Configuring the client NFS mount includes creating the target directory structure for access to the Sun ZFS Storage Appliance as well as the specific NFS mount options necessary for optimal system performance. Mount options for Oracle Solaris and Linux clients are as follows:

- Linux:

```
rw,bg,hard,nointr,rsize=1048576,wsize=1048576,tcp,vers=3,timeo=600
```

- Oracle Solaris:

```
rw,bg,hard,nointr,rsize=1048576,wsize=1048576,proto=tcp,vers=3,forcedirectio
```

Tuning the Linux Network and Kernel

Depending on the specific Linux installation, the NFS client software and necessary supporting software subsystems may or may not be enabled. Three Linux services required to run NFS are `portmap`, `nfs`, and `nfslock`. The services can be configured to run after reboot using the `chkconfig` command and enabled dynamically using the service command, as follows:

```
# chkconfig portmap on
# service portmap start
# chkconfig nfs on
# service nfs start
# chkconfig nfslock on
# service nfslock start
```

To ensure the `portmap` service has access to the `/etc/hosts.allow` and `/etc/hosts.deny` files, open up group and world read permissions on these files after verifying with local system administration officials that read permissions may be granted for these files:

```
# ls -l /etc/host*
```

```
-rw-r--r-- 1 root root   17 Jul 23  2000 /etc/host.conf
-rw-r--r-- 1 root root 1394 Mar  4 10:36 /etc/hosts
-rw------- 1 root root  161 Jan 12  2000 /etc/hosts.allow
-rw-r--r-- 1 root root  147 Mar  3 14:03 /etc/hosts.backupbyExadata
-rw------- 1 root root  347 Jan 12  2000 /etc/hosts.deny
-rw-r--r-- 1 root root  273 Mar  3 14:03 /etc/hosts.orig

# dcli -l root -g /home/oracle/dbs_group chmod 644 /etc/hosts.allow
# dcli -l root -g /home/oracle/dbs_group chmod 644 /etc/hosts.deny

# ls -l /etc/host*
-rw-r--r-- 1 root root   17 Jul 23  2000 /etc/host.conf
-rw-r--r-- 1 root root 1394 Mar  4 10:36 /etc/hosts
-rw-r--r-- 1 root root  161 Jan 12  2000 /etc/hosts.allow
-rw-r--r-- 1 root root  147 Mar  3 14:03 /etc/hosts.backupbyExadata
-rw-r--r-- 1 root root  347 Jan 12  2000 /etc/hosts.deny
-rw-r--r-- 1 root root  273 Mar  3 14:03 /etc/hosts.orig
```

High-bandwidth TCP protocol processing may require additional tuning beyond default operating system configurations. Verify factory installed Exdata settings in the `/etc/sysctl.conf` are set to the following:

```
net.core.wmem_max=4194304
net.core.rmem_max=2097152
net.ipv4.tcp_wmem=4194304
net.ipv4.tcp_rmem=2097152
```

If updates are necessary, run `sysctl -p` to reconfigure the system:

```
# dcli -l root -g /home/oracle/dbs_group sysctl -p
```

Further client, operating system, network, and kernel tuning may be needed, including software updates, to maximize device driver, networking, and kernel throughput related to network I/O processing. These tuning procedures are system-specific and beyond the scope of this paper. Consult with your operating system and NIC vendors for evaluation and implementation details.

Configuring Oracle Direct NFS (dNFS)

Configuring Oracle Direct NFS (dNFS) is an optional step that may improve throughput for Oracle RMAN backup and restore operations. A complete description of dNFS configuration is available for each specific release of the Oracle Database software from http://support.oracle.com.

NOTE: Prior to configuring dNFS, apply Oracle Database patch 8808984 to ensure optimal dNFS operation. Patch 8808984 is available from http://support.oracle.com, and is included in Oracle Exadata Database 11.2.0.1 (BP 8).

A summary of how to configure dNFS is as follows:

1. Shut down the running instance of the Oracle Database software.
2. Enable dNFS using one of the options below:
   - For *version 11.2.0.2 or greater* of the Oracle Database software, enter:
     ```
     $ make -f \
          $ORACLE_HOME/rdbms/lib/ins_rdbms.mk
     dnfs_on
     ```
   - For a *version prior to 11.2.0.2*, enter:
     ```
     $ ln -sf \
          $ORACLE_HOME/lib/libnfsodm11.so \
          $ORACLE_HOME/lib/libodm11.so
     ```
3. Update the `oranfstab` (`/etc/oranfstab`) file with entries showing the channels and shares accessed on the Sun ZFS Storage Appliance. The following example shows how to access the `backup00` share on `aie-7420a-h1` over two separate IP addresses, 192.168.36.200 and 192.168.36.201.
   ```
   server: aie-7420a-h1
   path: 192.168.36.200
   export: /export/dbname/backup1 mount: /zfssa/dbname/backup1
   export: /export/dbname/backup3 mount: /zfssa/dbname/backup3
   ...
   server: aie-7420a-h2
   path: 192.168.36.201
   export: /export/dbname/backup2 mount: /zfssa/dbname/backup2
   export: /export/dbname/backup4 mount: /zfssa/dbname/backup4
   ```
4. Restart the Oracle Database software instance.

Tuning the Oracle Database Instance for Oracle RMAN Backup and Restore

Optimizing high-bandwidth backup and restore operations using Oracle RMAN and the Sun ZFS Storage Appliance requires adjusting the instance parameters that control I/O buffering. For information about how to tune these parameters on different versions of the Oracle Database software, see Article ID 1072545.1: *RMAN Performance Tuning Using Buffer Memory Parameters*) at http://support.oracle.com.

For Oracle Exadata, tuning the following four parameters should be considered:

- `_backup_disk_bufcnt` – Number of buffers used to process backup sets
- `_backup_disk_bufsz` – Size of the buffers used to process backup sets
- `_backup_file_bufcnt` – Number of buffers used to process image copies
- `_backup_file_bufsz` – Size of the buffers used to process image copies

For backup and restore operations on backup sets and image copies, set the number of buffers to `64`:

```
SQL> alter system set "_backup_disk_bufcnt"=64;
SQL> alter system set "_backup_file_bufcnt"=64;
```

For backup operations, set the buffer size to 1 MB for backup sets:

```
SQL> alter system set "_backup_disk_bufsz"=1048576;
SQL> alter system set "_backup_file_bufsz"=1048576;
```

For restore operations, set the buffer size to 128 kB:

```
SQL> alter system set "_backup_disk_bufsz"=1048576;
SQL> alter system set "_backup_file_bufsz"=1048576;
```

These commands may be configured persistently by adding them to the `SPFILE`, or they may be set dynamically in the Oracle RMAN run block used to execute the backup or restore operations.

The following code fragments show how to dynamically tune the buffer sizes and counts for backup and restore operations.

- Backup set backup:

```
run
{
   sql 'alter system set "_backup_disk_bufcnt"=64';
   sql 'alter system set "_backup_disk_bufsz"=1048576';
   allocate channel...
...
   backup as backupset database;
}
```

- Backup set restore:

```
run
{
   sql 'alter system set "_backup_disk_bufcnt"=64';
   sql 'alter system set "_backup_disk_bufsz"=1048576’;
   allocate channel...
...
   restore database;
}
```

- Image copy backup:

```
run
{
   sql 'alter system set "_backup_file_bufcnt"=64';
   sql 'alter system set "_backup_file_bufsz"=1048576';
   allocate channel...
...
   backup as copy database;
}
```

- Image copy restore:

```
run
{
   sql 'alter system set "_backup_file_bufcnt"=64';
   sql 'alter system set "_backup_file_bufsz"=1048576';
   allocate channel...
...
   restore database;
}
```

Performing an incrementally applied backup requires reading an incremental backup set and writing to an image copy. To tune buffers for incrementally applied backups, run the following:

```
run
{
   sql 'alter system set "_backup_disk_bufcnt"=64';
   sql 'alter system set "_backup_disk_bufsz"=1048576';
   sql 'alter system set "_backup_file_bufcnt"=64';
   sql 'alter system set "_backup_file_bufsz"=1048576';
   allocate channel...
...
   recover copy of database;
}
```

Creating Dedicated Services for Oracle RMAN Operations

Eight services dedicated to Oracle RMAN processing can be configured to optimize management of load balancing, high availability, and upgrades. These services can be evenly load balanced over all the nodes of an Oracle Exadata system. Availability and performance can be optimized by configuring the services to run on a preferred instance while preparing them to fail over to any instance in the cluster. If these services are configured, upgrading a ¼ or ½ rack Oracle Exadata system does not require changing the connect string of the Oracle RMAN run block.

The `srvctl` utility is used to install services for Oracle RMAN processing. The following code fragment shows how to create eight services evenly distributed over a four-node cluster that are set up to fail over to any other node in the cluster. In this example, the services are installed for a database named `dbname` and they are named `dbname_bkup[1-8]`.

```
srvctl add service -d dbname -r dbname1 -a dbname2,dbname3,dbname4 \
                   -s dbname_bkup1
srvctl start service -d dbname -s dbname_bkup1
srvctl add service -d dbname -r dbname2 -a dbname1,dbname3,dbname4 \
                   -s dbname_bkup2
srvctl start service -d dbname -s dbname_bkup2
srvctl add service -d dbname -r dbname3 -a dbname1,dbname2,dbname4 \
                   -s dbname_bkup3
srvctl start service -d dbname -s dbname_bkup3
srvctl add service -d dbname -r dbname4 -a dbname1,dbname2,dbname3 \
                   -s dbname_bkup4
srvctl start service -d dbname -s dbname_bkup4
srvctl add service -d dbname -r dbname1 -a dbname2,dbname3,dbname4 \
                   -s dbname_bkup5
srvctl start service -d dbname -s dbname_bkup5
srvctl add service -d dbname -r dbname2 -a dbname1,dbname3,dbname4 \
                   -s dbname_bkup6
srvctl start service -d dbname -s dbname_bkup6
srvctl add service -d dbname -r dbname3 -a dbname1,dbname2,dbname4 \
                   -s dbname_bkup7
srvctl start service -d dbname -s dbname_bkup7
srvctl add service -d dbname -r dbname4 -a dbname1,dbname2,dbname3 \
                   -s dbname_bkup8
srvctl start service -d dbname -s dbname_bkup8
```

Configuring Oracle RMAN

Configuring Oracle RMAN channels and parallelism includes specifying the file system targets for the Oracle RMAN backup channels and the total number of channels used for backup and restore operations. Performance benefits can be realized by configuring sixteen Oracle RMAN channels spanning the available Sun ZFS Storage Appliance shares. Configure Oracle RMAN channels such that they are evenly distributed over the Oracle Database instances and nodes in the RAC cluster and evenly distributed over the shares exported from the Sun ZFS Storage Appliance.

The following code fragments show sample Oracle RMAN run blocks for performing backup and restore operations for backup sets and image copies as well as applying incremental merges to image copies. The sample code is based on the following database configuration:

- Database name: `dbname`
- SYSDBA login: `sys/welcome`
- Scan address: `ad01-scan`
- Service names for the backup: `dbname_bkup[1-8]`

For performance optimized systems, the Sun ZFS Storage Appliance can be configured for two different use cases:

- A one-pool configuration in which the Sun ZFS Storage Appliance exports eight shares used as eight mount points
- A two-pool configuration in which the sun ZFS Storage Appliance exports sixteen shares as sixteen mount points

The Oracle RMAN run blocks for backup and restore using backup sets and image copies are shown in the examples in the sections below. In these examples, the mount points for the 8- and 16-share configurations are accessed as `/zfssa/dbname/backup1` through `/zfssa/dbname/backup16`.

Conclusion

The Sun ZFS Storage Appliance provides a simple, high-performance, and cost-effective platform to ensure data protection for Oracle databases. It is easily integrated with existing database systems with file-based access provided through NFS using high-bandwidth network protocols, including 10 Gb Ethernet and InfiniBand (IB). Capital and operational costs are driven down by the elimination of backup application license and support fees associated with disk-based backup, as well as license-free data services, including snapshot, compression, and replication.

Engineered to tolerate a wide range of failures, the Sun ZFS Storage Appliance offers single, double, and triple parity RAID to protect against one, two, or three simultaneous physical drive failures. For high-performance requirements, double and triple mirroring can also be used to protect against one-drive and two-drive failure. To further protect against bit errors in disk drives, individual data blocks are checksummed to ensure integrity, while advanced features of the Sun ZFS Storage Appliance dynamically repair compromised data before it reaches the client.

The Sun ZFS Storage Appliance offers capabilities well beyond simply storing database backups over a direct connection to an Oracle Exadata Database Machine. File-based NFS protocol enables Oracle RMAN incrementally updated backups, employed in the Oracle recommended backup strategy (see http://www.oracle.com/technetwork/database/features/availability/maa-tech-wp-sundbm-backup-final-129256.pdf), leading to reductions in backup windows to more effectively meet recovery time objectives (RTO).

For customers demanding the highest levels of system efficiency, Sun ZFS Storage Appliance snapshots of image copy backups can be deployed throughout the enterprise to ensure that every part of the business has access to the most current data.

Access to the Sun ZFS Storage Appliance can be made available to database administrators through the use of role-based user IDs, enabling them to perform specific functions when needed without the assistance of a storage administrator. An ID and role can easily be set up to restrict a database administrator to a limited set of functions, such as creating a snapshot, cloning a snapshot, and adding a share to a predefined project.

Testing has shown that an Oracle Exadata Database Machine half-rack configuration attached to a small Sun ZFS Storage 74x0 cluster with four disk shelves demonstrates NFS-based backup and restore rates suitable for even the most performance-oriented environments.

Like the Oracle Database, the Sun ZFS Storage Appliance is a scalable architecture. By adding disk drives and CPU processing, the Sun ZFS Storage Appliance can currently support raw capacities exceeding 1 PB while maintaining a balanced system with respect to performance.

As shown in this paper, the Sun ZFS Storage Appliance is the preferred general purpose data protection solution for Oracle RMAN backup and restore of Oracle databases, including the Oracle Exadata Database Machine.

Appendix A. Configuring the Sun ZFS Storage Appliance

The Sun ZFS Storage Appliance screens below show details for configuring a Sun ZFS Storage 7420 appliance to support backup and recovery operations using Oracle RMAN. This clustered system has two controllers and four disk shelves. Access to the system is through active/standby IPMP. Two storage pools were created from half the disks (1 TB) and half the write log SSDs from the four disk shelves. Figure 1 and Figure 2 show example datalink configurations for both 10 Gb Ethernet and InfiniBand.

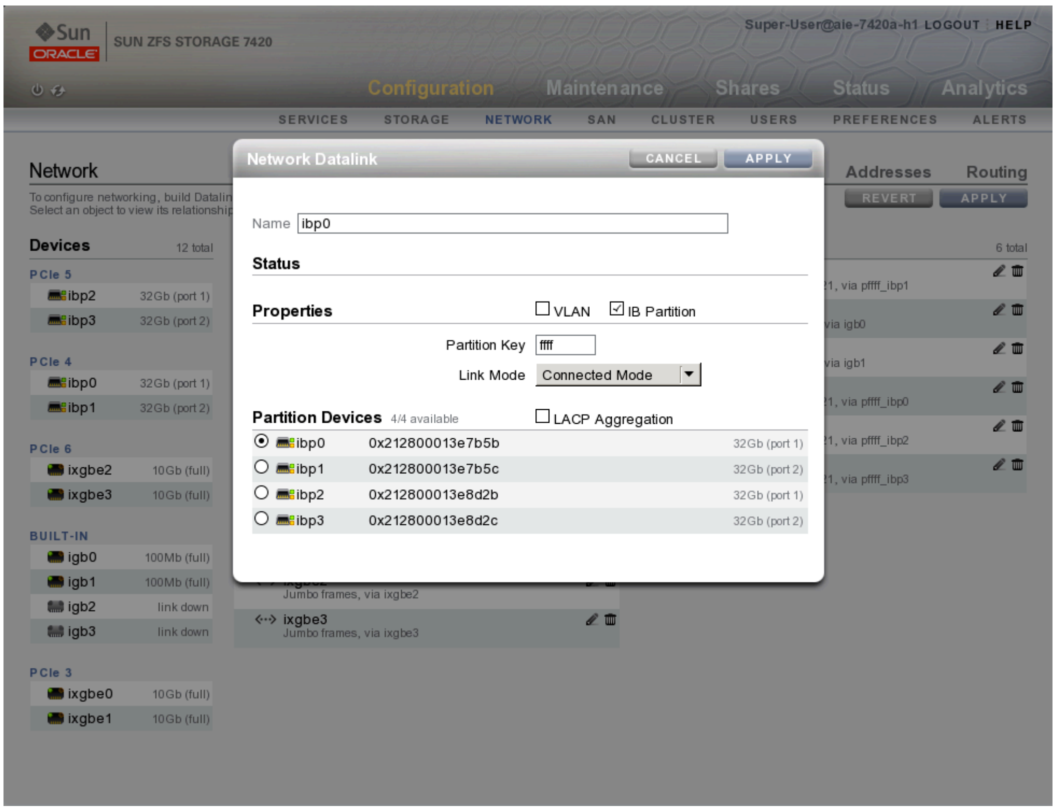

Figure1. Configuring Sun ZFS Storage Appliance datalink for an IB interface

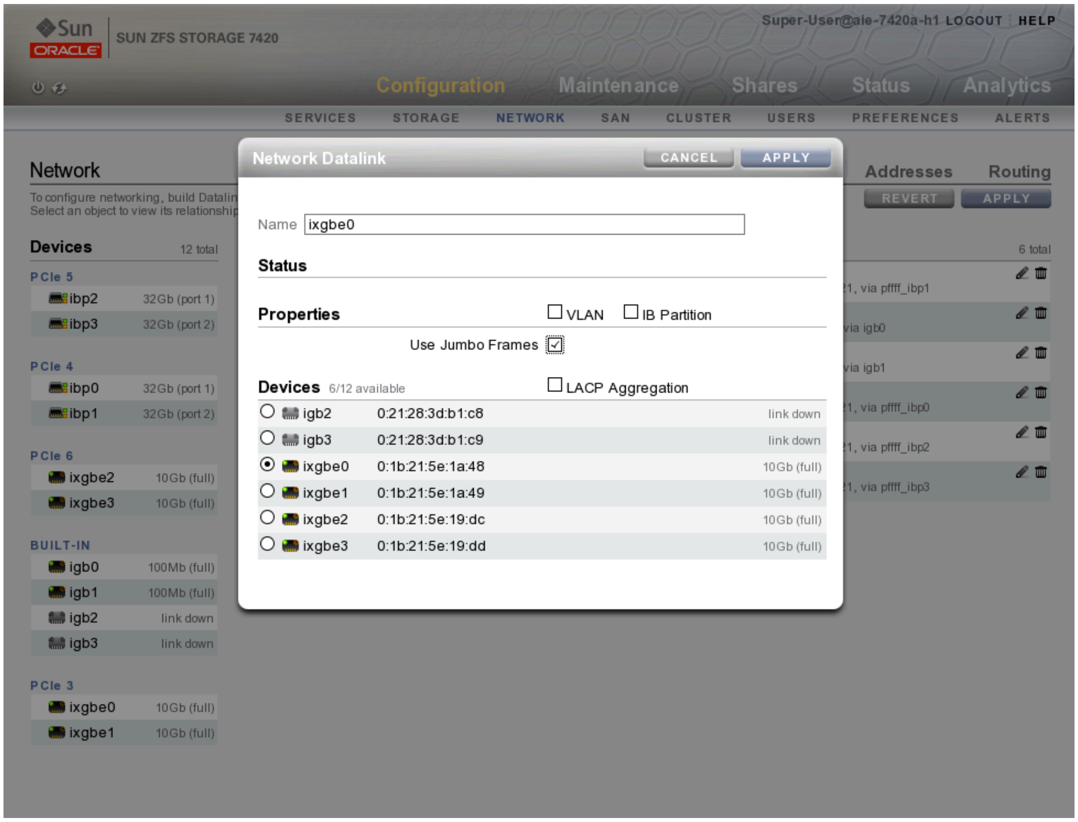

Figure 2. Configuring Sun ZFS Storage Appliance datalink for a 10 Gb Ethernet interface

Figure 3 shows the completed datalink configuration for IB and 10 Gb Ethernet devices.

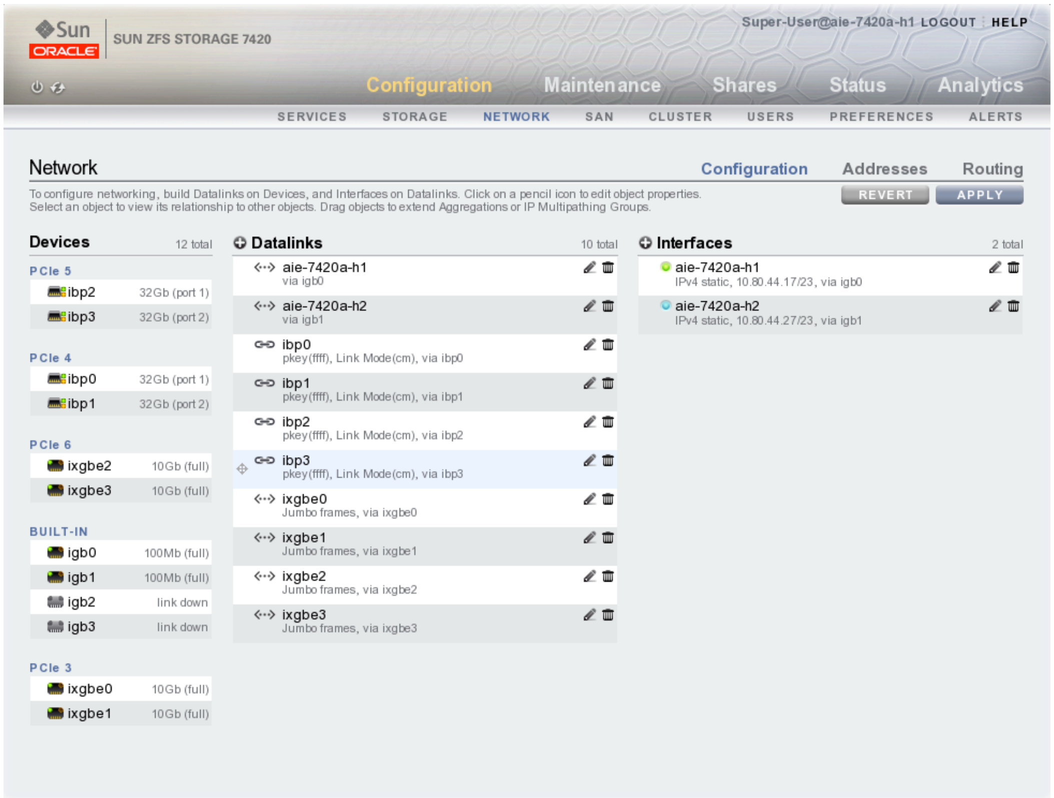

Figure 3. Completed datalink configuration

Figure 4 shows the base interface configuration for IP multipathing (IPMP). When implementing IPMP on the Sun ZFS Storage Appliance, the base interfaces are assigned the commonly used static IP address 0.0.0.0/8. Following the completion of the base configuration, an IPMP group will be built over the base interfaces.

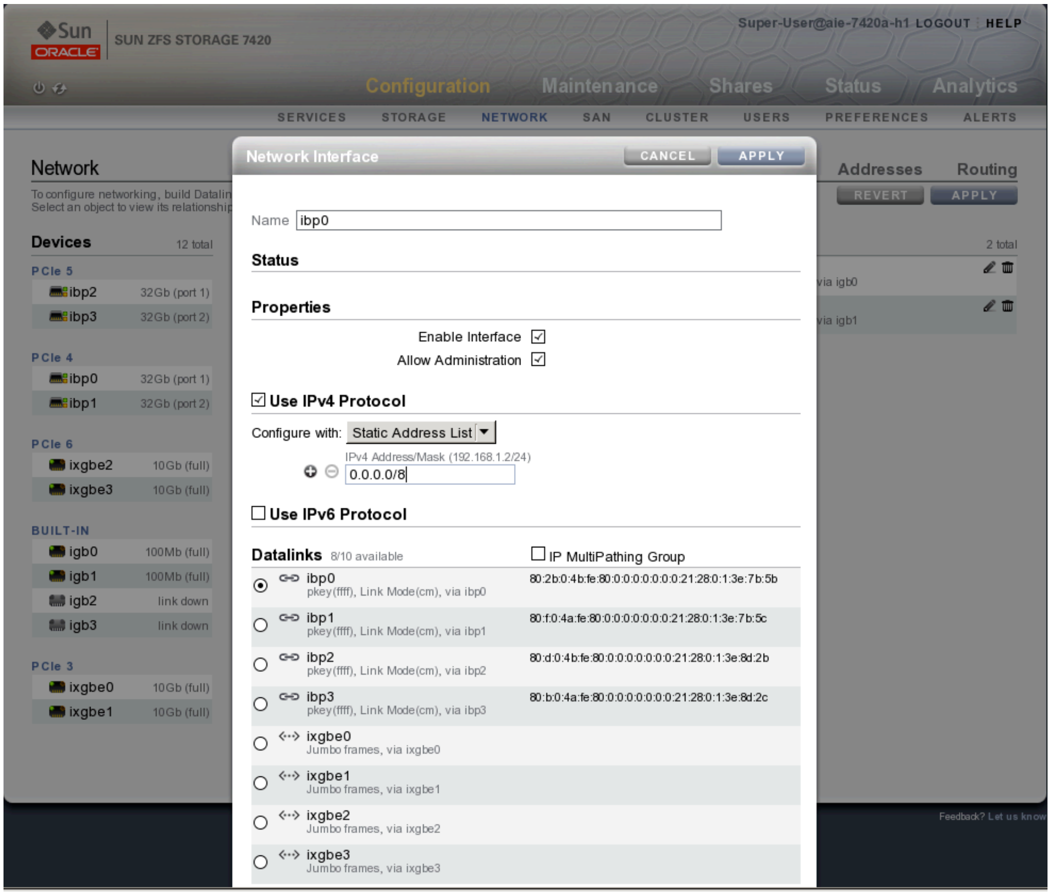

Figure 4. Base IPMP interface configured with IB datalink

Figures 5 and Figure 6 show how to create active/standby IPMP interfaces on a Sun ZFS Storage Appliance that are designed to meet no-single-point-of-failure requirements. The configuration assumes that the upper ports of each IB HCA are cabled into the same switch, and the lower ports are cabled to an alternate switch. The IPMP group is built to tolerate switch failure, link failure, and HCA failure. To maximize throughput in the case of a head node fault, the active ports of each IPMP group are assigned to alternate HCAs, which eliminates the HCA and the PCI slot as a bottleneck.

The first IPMP group is implemented on `ibp0` and `ibp3`, as shown in Figure 5, to ensure redundancy between switches and cards. `ibp0` is set as active and `ibp3` is set as passive. The second IPMP group is implemented on `ipb1` and `ibp2` with `ibp1` passive and `ibp2` active, as shown in Figure 6, again to ensure redundancy between switches and cards. In the event of a head failover, all IPMP resources will be active on the same head. With `ibp0` and `ibp2` configured as active, the load will be distributed over both cards and both slots in the head for optimal performance.

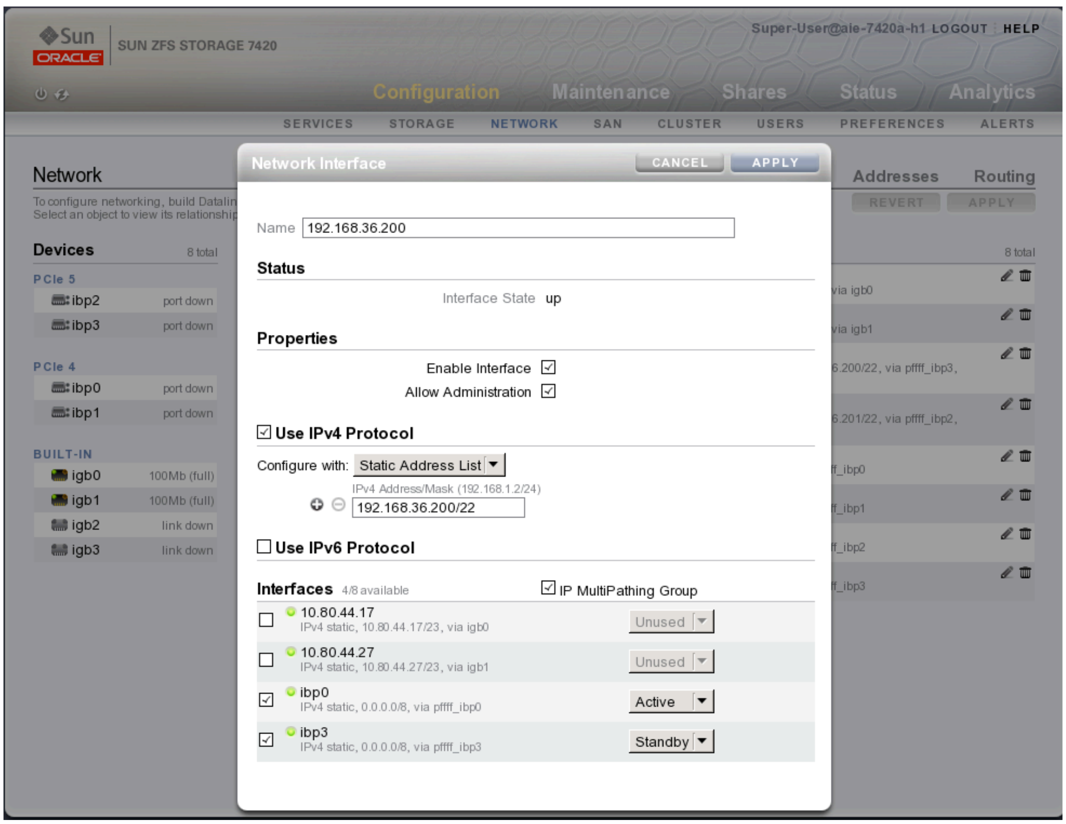

Figure 5. First active/standby IPMP interface over IB configured

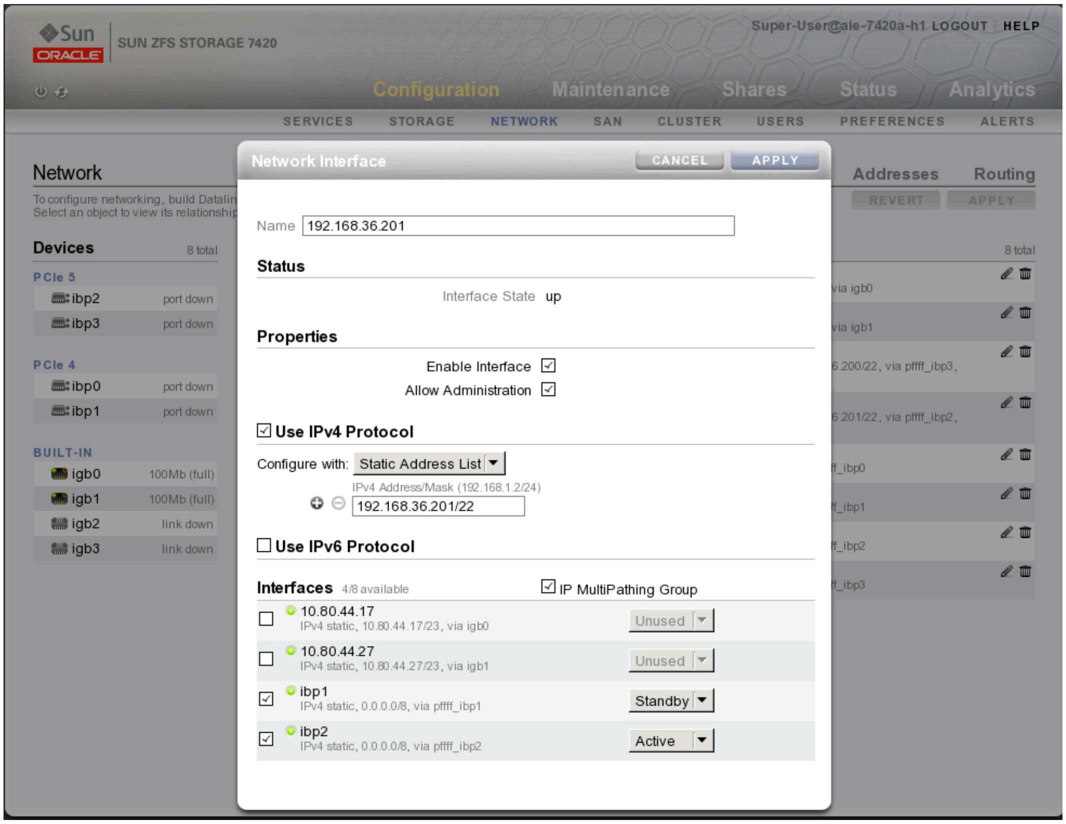

Figure 6. Second active/standby IPMP interface over IB configured

Figure 7 shows how to configure adaptive routing to allow for data to be returned on the same interface from which it was requested. Selecting the **Adaptive** multihoming mode is sufficient to enable the adaptive routing feature. No commit is required.

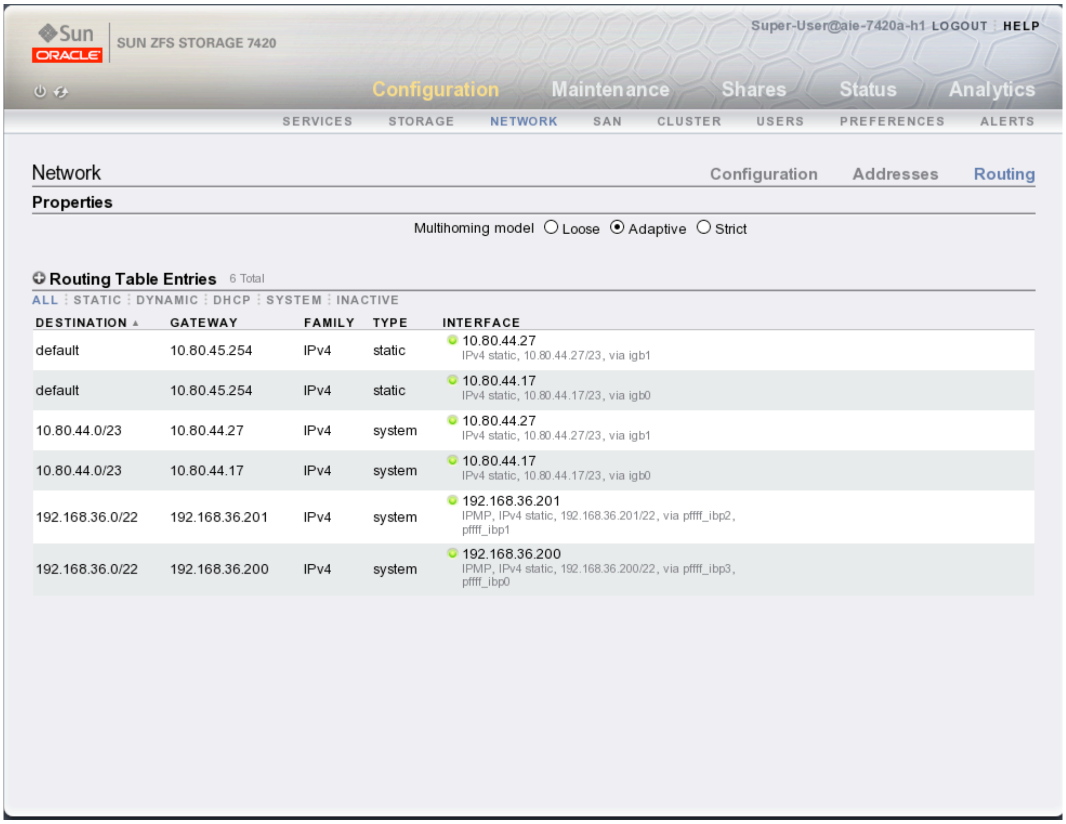

Figure 7. Configuring adaptive routing on the Sun ZFS Storage Appliance

To maximize system throughput, two equally sized storage pools are configured by assigning half of the physical drives in each drive tray to each storage pool, as shown in Figure 8. Note that each drive tray contains 24 drives. In the case of systems implemented with write-optimized flash devices, all four write-optimized flash devices are placed in the same tray, leaving 20 spinning disk drives.

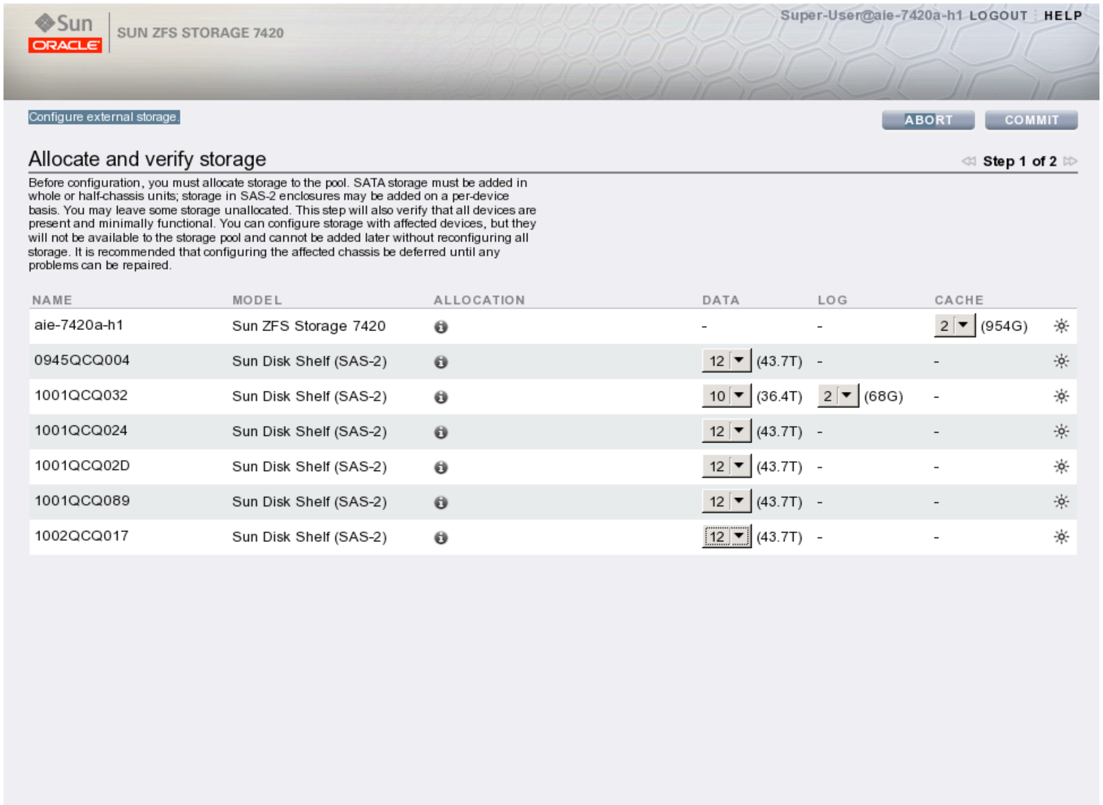

Figure 8. Storage pool configured based on half of drives in each tray

In Figure 9, a Sun ZFS Storage cluster with two IP addresses on an IPMP group, four IB ports, and two pools, is configured such that each cluster node owns one of the two IPMP groups, the requisite interface ports to support the group, and one of the two pools.

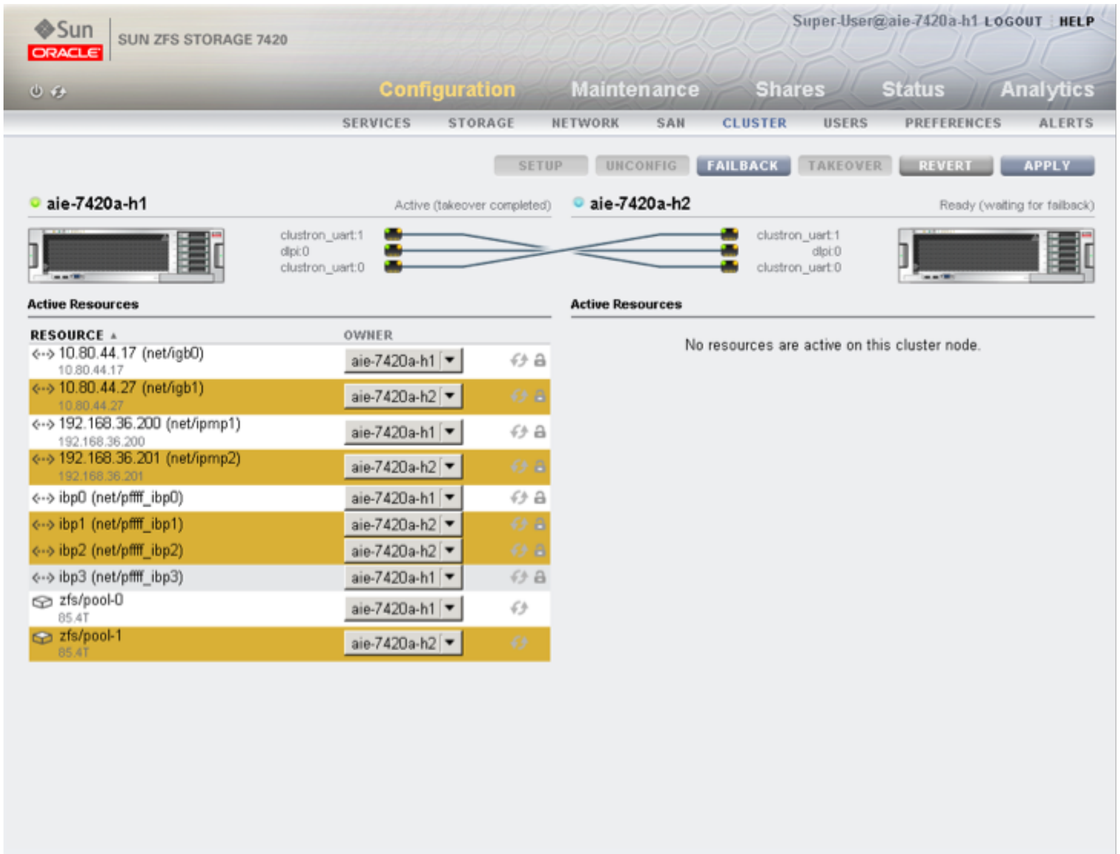

Figure 9. Cluster configuration for the Sun ZFS Storage Appliance

Two implementation options are available for deploying the Sun ZFS Storage Appliance to support Oracle RMAN backups: a one-pool configuration and a two-pool configuration. In the one-pool configuration, the Sun ZFS Storage Appliance is configured with a single project containing eight shares. The project has the same name as the database. Thus, in the example, the project is named `dbname`. The database share mount points in the example are `/export/dbname/backup1` through `/export/dbname/backup8`.

In the two-pool configuration, two projects are configured, one per pool. The project name must be unique within the Sun ZFS Storage Appliance cluster. Therefore, the example uses the project names `dbname-1` and `dbname-2` for simpler management.

To simplify administration of Oracle RMAN and the client operating system, the mount points for the shares reflect the database name `/export/dbname`. To optimize performance, eight shares are configured in each project, and the export names of the eight shares are assigned such that one project supports the even shares, (`/export/dbname/backup2`, … `/export/dbname/backup16`) and the other project supports the odd shares (`/export/dbname/backup1`, `/export/dbname/backup3`, … `/export/dbname/backup15`).

Figure 10 shows an example of the general share configuration setting for either pool configuration.

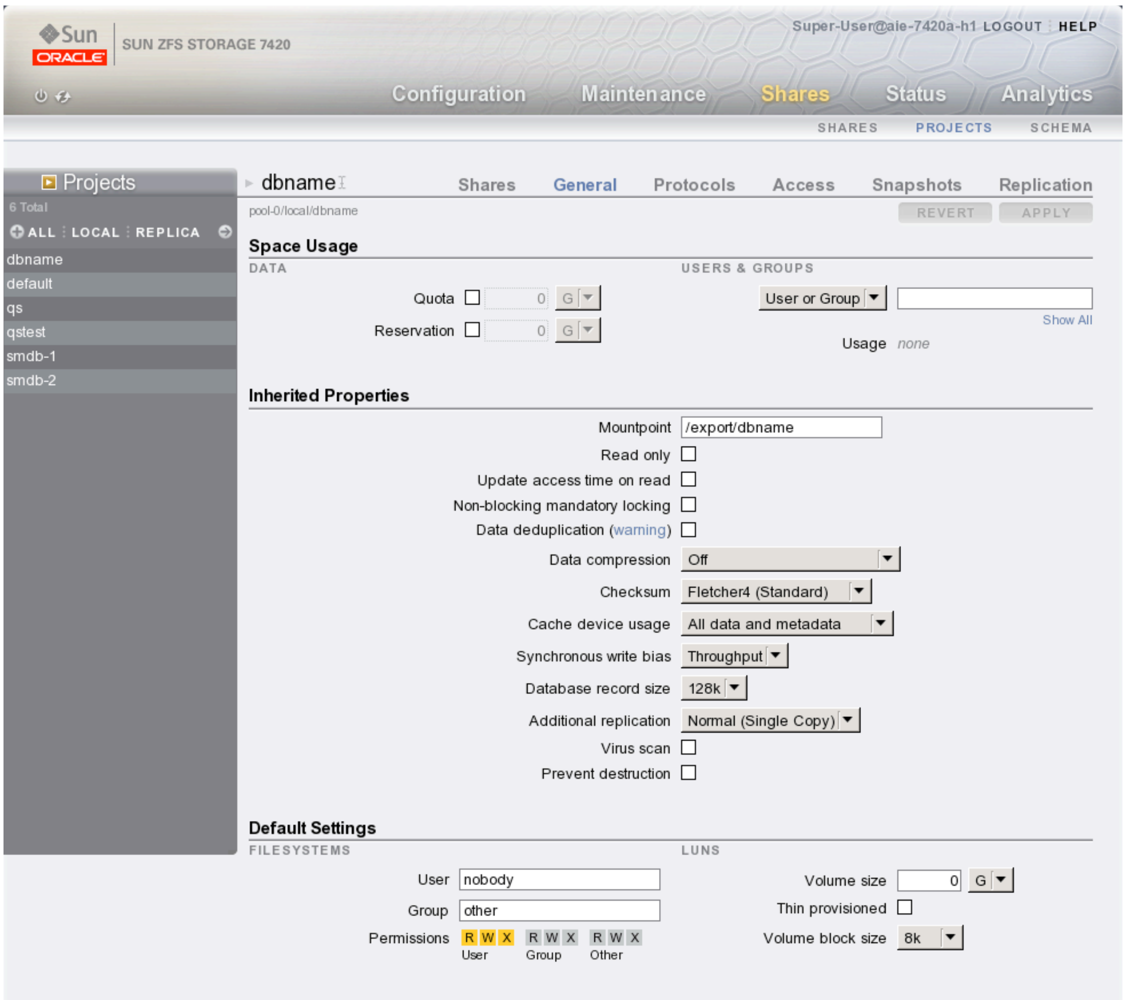

Figure 10. Share configuration setting appropriate for either a one-pool or two-pool configuration

Figure 11 shows the share layout for a one-pool configuration in a project named `dbname`.

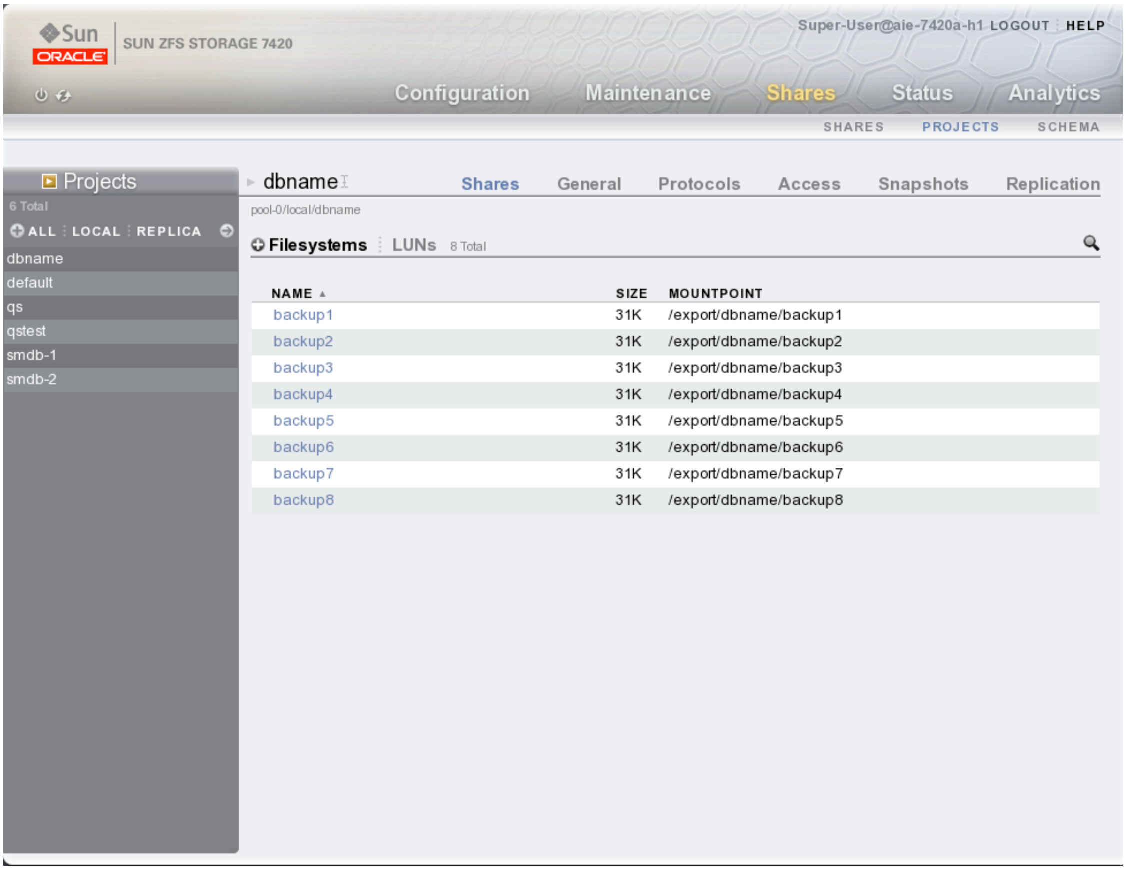

Figure 11. Share layout for a one-pool configuration

Figure 12 shows how root access can be granted for all database nodes in an Oracle Exadata rack by specifying a specific NFS access rule to allow root access for any server on the Oracle Exadata private network, in this case, 192.168.36.0/22. Additional exceptions for other networks are shown as an example.

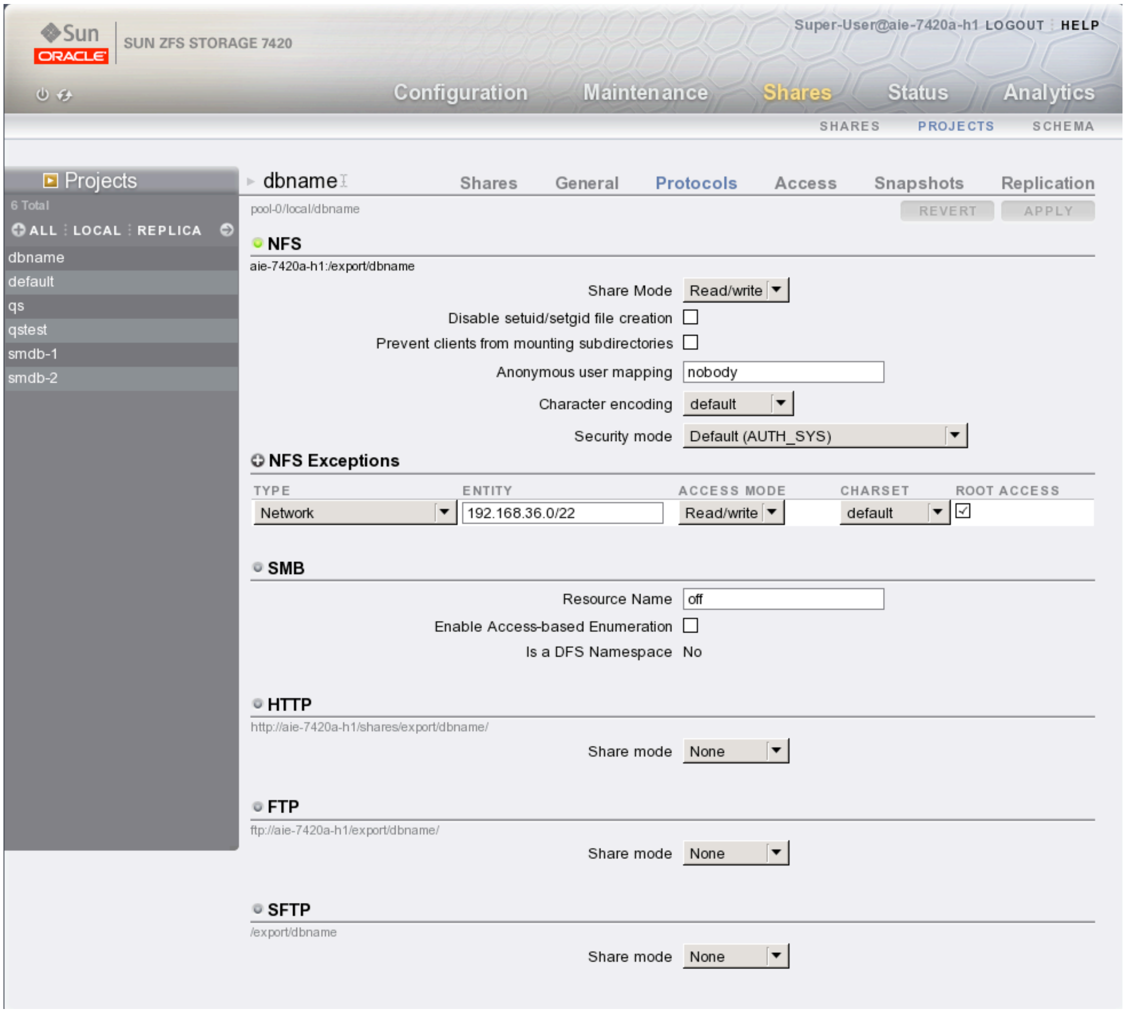

Figure 12. Granting root access for all database nodes in an Oracle Exadata rack

Figure 13 and Figure 14 show the configuration for two pools with the odd-numbered shares in project `dbname-1` and the even-numbered shares in project `dbname-2`.

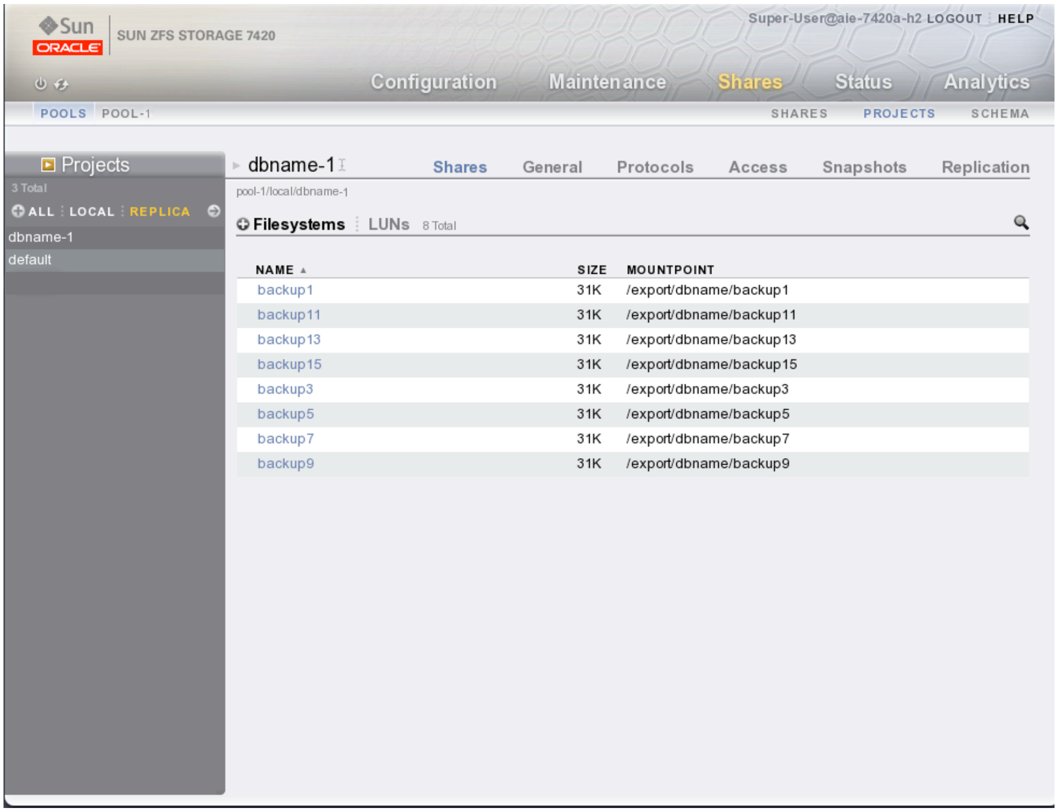

Figure 13. Configuring a pool with the odd-numbered shares in project `dbname-1`

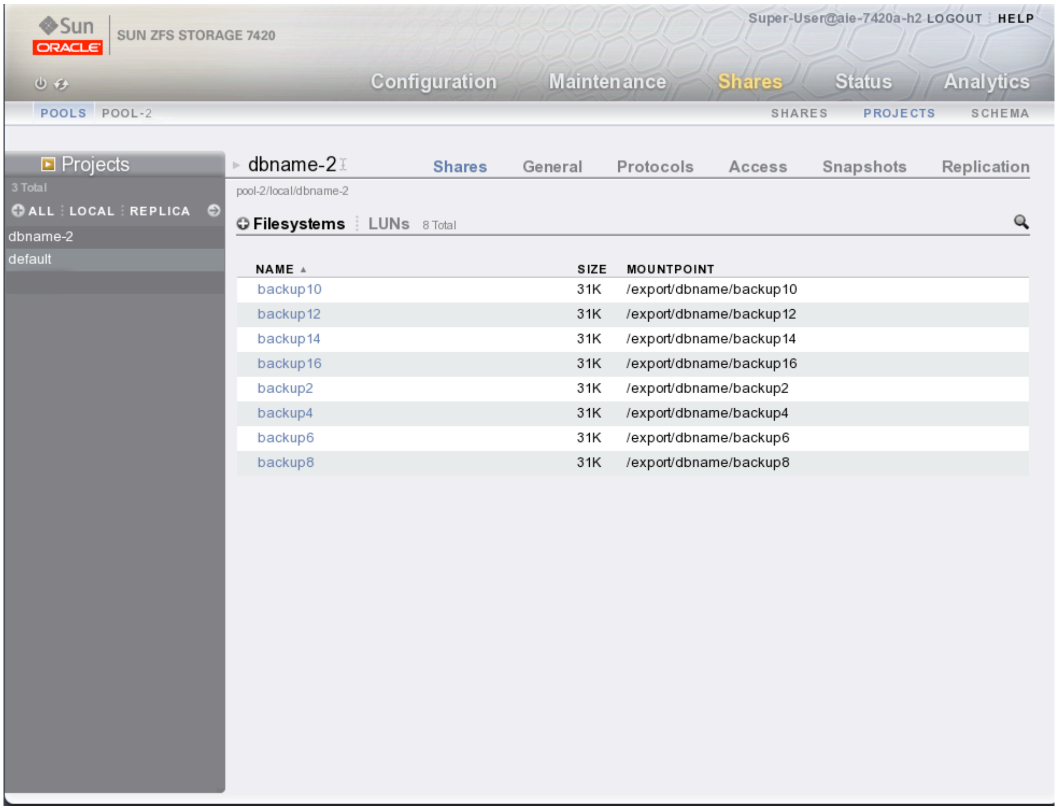

Figure 14. Configuring a pool with the even-numbered shares in project `dbname-2`

Appendix B. Sample Scripts to Connect a Linux RAC Cluster to a Sun ZFS Storage Appliance

This section contains sample scripts showing how to attach a Sun ZFS Storage Appliance to a Linux RAC cluster, such as an Oracle Exadata system. These scripts are designed to support a database named `dbname` in a one-pool and a two-pool Sun ZFS Storage Appliance configuration (see Appendix A. Configuring the Sun ZFS Storage Appliance).

The implementation steps are as follows:

1. Set up the directory structure (mount points) to mount the shares on the host.
2. Update `/etc/fstab` to mount the shares exported from the Sun ZFS Storage Appliance to the appropriate mount points.
3. Create an `init.d` service to automate the process of mounting and unmounting the shares.
4. Update the `oranfstab` file to access the Sun ZFS Storage Appliance exported shares.
5. Mount the shares on the host.
6. Change the permissions of the mounted shares to match the permission settings of `ORACLE_HOME`.
7. Restart the Oracle Database instance to pick up the changes to the `oranfstab` file.

These steps are described in more detail in the sections below.

Setting Up the Directory Structure to Mount the Shares on the Host

Set up mount points for the shares on the host as shown below.

For a one-pool configuration:

```
mkdir -p /zfssa/dbname/backup1
mkdir -p /zfssa/dbname/backup2
mkdir -p /zfssa/dbname/backup3
mkdir -p /zfssa/dbname/backup4
mkdir -p /zfssa/dbname/backup5
mkdir -p /zfssa/dbname/backup6
mkdir -p /zfssa/dbname/backup7
mkdir -p /zfssa/dbname/backup8
```

For a two-pool configuration:

```
mkdir -p /zfssa/dbname/backup1
mkdir -p /zfssa/dbname/backup2
mkdir -p /zfssa/dbname/backup3
mkdir -p /zfssa/dbname/backup4
mkdir -p /zfssa/dbname/backup5
mkdir -p /zfssa/dbname/backup6
```

```
mkdir -p /zfssa/dbname/backup7
mkdir -p /zfssa/dbname/backup8
mkdir -p /zfssa/dbname/backup9
mkdir -p /zfssa/dbname/backup10
mkdir -p /zfssa/dbname/backup11
mkdir -p /zfssa/dbname/backup12
mkdir -p /zfssa/dbname/backup13
mkdir -p /zfssa/dbname/backup14
mkdir -p /zfssa/dbname/backup15
mkdir -p /zfssa/dbname/backup16
```

Updating the `/etc/fstab` File

To update the `/etc/fstab` file, use the appropriate option below.

NOTE: The UNIX newline escape character (\) indicates a single line of code has been wrapped to a second line in the listing below. When entering a wrapped line into `fstab`, removed the \ character and combine the two line segments, separated by a space, into a single line.

For a one-pool configuration:

```
192.168.36.200:/export/dbname/backup1 /zfssa/dbname/backup1 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup2 /zfssa/dbname/backup2 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup3 /zfssa/dbname/backup3 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup4 /zfssa/dbname/backup4 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup5 /zfssa/dbname/backup5 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup6 /zfssa/dbname/backup6 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup7 /zfssa/dbname/backup7 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup8 /zfssa/dbname/backup8 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
```

For a two-pool configuration:

```
192.168.36.200:/export/dbname/backup1 /zfssa/dbname/backup1 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.201:/export/dbname/backup2 /zfssa/dbname/backup2 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup3 /zfssa/dbname/backup3 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.201:/export/dbname/backup4 /zfssa/dbname/backup4 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
```

```
192.168.36.200:/export/dbname/backup5 /zfssa/dbname/backup5 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.201:/export/dbname/backup6 /zfssa/dbname/backup6 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup7 /zfssa/dbname/backup7 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.201:/export/dbname/backup8 /zfssa/dbname/backup8 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup9 /zfssa/dbname/backup9 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.201:/export/dbname/backup10 /zfssa/dbname/backup10 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup11 /zfssa/dbname/backup11 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.201:/export/dbname/backup12 /zfssa/dbname/backup12 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup13 /zfssa/dbname/backup13 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.201:/export/dbname/backup14 /zfssa/dbname/backup14 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.200:/export/dbname/backup15 /zfssa/dbname/backup15 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
192.168.36.201:/export/dbname/backup16 /zfssa/dbname/backup16 nfs \
    rw,bg,hard,nointr,rsize=131072,wsize=1048576,tcp,nfsvers=3,timeo=600 0 0
```

Creating an `init.d` Service

Create an `init.d` service using the appropriate option below.

For a one-pool configuration:

```
#!/bin/sh
#
# zfssa_dbname: Mount ZFSSA project dbname for database dbname
#
# chkconfig: 345 61 19
# description: mounts ZFS Storage Appliance shares
#

start()
{
   mount /zfssa/dbname/backup1
   mount /zfssa/dbname/backup2
   mount /zfssa/dbname/backup3
   mount /zfssa/dbname/backup4
   mount /zfssa/dbname/backup5
   mount /zfssa/dbname/backup6
   mount /zfssa/dbname/backup7
   mount /zfssa/dbname/backup8
```

```
   echo "Starting $prog: "
}

stop()
{
   umount /zfssa/dbname/backup1
   umount /zfssa/dbname/backup2
   umount /zfssa/dbname/backup3
   umount /zfssa/dbname/backup4
   umount /zfssa/dbname/backup5
   umount /zfssa/dbname/backup6
   umount /zfssa/dbname/backup7
   umount /zfssa/dbname/backup8
   echo "Stopping $prog: "
}

case "$1" in
  start)
        start
        ;;
  stop)
        stop
        ;;
  restart)
        stop
        start
        ;;
  status)
        mount
        ;;
  *)
        echo "Usage: $0 {start|stop|restart|status}"
        exit 1
esac
```

For a two-pool configuration:

```
#!/bin/sh
#
# zfssa_dbname: Mount ZFSSA projects dbname for database dbname
#
# chkconfig: 345 61 19
# description: mounts ZFS Storage Appliance shares
#

start()
```

```
{
   mount /zfssa/dbname/backup1
   mount /zfssa/dbname/backup2
   mount /zfssa/dbname/backup3
   mount /zfssa/dbname/backup4
   mount /zfssa/dbname/backup5
   mount /zfssa/dbname/backup6
   mount /zfssa/dbname/backup7
   mount /zfssa/dbname/backup8
   mount /zfssa/dbname/backup9
   mount /zfssa/dbname/backup10
   mount /zfssa/dbname/backup11
   mount /zfssa/dbname/backup12
   mount /zfssa/dbname/backup13
   mount /zfssa/dbname/backup14
   mount /zfssa/dbname/backup15
   mount /zfssa/dbname/backup16

   echo "Starting $prog: "
}

stop()
{
   umount /zfssa/dbname/backup1
   umount /zfssa/dbname/backup2
   umount /zfssa/dbname/backup3
   umount /zfssa/dbname/backup4
   umount /zfssa/dbname/backup5
   umount /zfssa/dbname/backup6
   umount /zfssa/dbname/backup7
   umount /zfssa/dbname/backup8
   umount /zfssa/dbname/backup9
   umount /zfssa/dbname/backup10
   umount /zfssa/dbname/backup11
   umount /zfssa/dbname/backup12
   umount /zfssa/dbname/backup13
   umount /zfssa/dbname/backup14
   umount /zfssa/dbname/backup15
   umount /zfssa/dbname/backup16
   echo "Stopping $prog: "
}

case "$1" in
  start)
        start
```

```
        ;;
  stop)
        stop
        ;;
  restart)
        stop
        start
        ;;
  status)
        mount
        ;;
  *)
        echo "Usage: $0 {start|stop|restart|status}"
        exit 1
esac
```

(Optional) Enable the `init.d` service for start-on-boot by entering:

```
# chkconfig zfssa_dbname on
```

(Optional) Start and stop the service manually using the service command shown below:

```
# service zfssa_dbname start
# service zfssa_dbname stop
```

Updating `oranfstab` to Access Sun ZFS Storage Appliance Exports

To update the `oranfstab` file to access Sun ZFS Storage Appliance exports, use the appropriate option below.

For one-pool configuration:

```
server: 192.168.36.200
path: 192.168.36.200
path: 192.168.36.200
export: /export/dbname/backup1 mount: /zfssa/dbname/backup1
export: /export/dbname/backup2 mount: /zfssa/dbname/backup2
export: /export/dbname/backup3 mount: /zfssa/dbname/backup3
export: /export/dbname/backup4 mount: /zfssa/dbname/backup4
export: /export/dbname/backup5 mount: /zfssa/dbname/backup5
export: /export/dbname/backup6 mount: /zfssa/dbname/backup6
export: /export/dbname/backup7 mount: /zfssa/dbname/backup7
export: /export/dbname/backup8 mount: /zfssa/dbname/backup8
```

For two-pool configuration:

```
server: 192.168.36.200
path: 192.168.36.200
export: /export/dbname/backup1 mount: /zfssa/dbname-2pool/backup1
export: /export/dbname/backup3 mount: /zfssa/dbname-2pool/backup3
export: /export/dbname/backup5 mount: /zfssa/dbname-2pool/backup5
export: /export/dbname/backup7 mount: /zfssa/dbname-2pool/backup7
export: /export/dbname/backup9 mount: /zfssa/dbname-2pool/backup9
export: /export/dbname/backup11 mount: /zfssa/dbname-2pool/backup11
export: /export/dbname/backup13 mount: /zfssa/dbname-2pool/backup13
export: /export/dbname/backup15 mount: /zfssa/dbname-2pool/backup15
server: 192.168.36.201
path: 192.168.36.201
export: /export/dbname/backup2 mount: /zfssa/dbname-2pool/backup2
export: /export/dbname/backup4 mount: /zfssa/dbname-2pool/backup4
export: /export/dbname/backup6 mount: /zfssa/dbname-2pool/backup6
export: /export/dbname/backup8 mount: /zfssa/dbname-2pool/backup8
export: /export/dbname/backup10 mount: /zfssa/dbname-2pool/backup10
export: /export/dbname/backup12 mount: /zfssa/dbname-2pool/backup12
export: /export/dbname/backup14 mount: /zfssa/dbname-2pool/backup14
export: /export/dbname/backup16 mount: /zfssa/dbname-2pool/backup16
```

Mounting the Shares on the Host

To mount the shares on the host, enter one of the following options:

```
# service mount_dbname start
```

or

```
# dcli -l root -g /home/oracle/dbs_group service mount_dbname start
```

Setting the Ownership of the Mounted Shares

Change the permission settings of the mounted shares to match the permission settings of `ORACLE_HOME`. In this example, the user and group ownerships are set to `oracle:dba`.

1. Enter one of the following two options:

```
# chown oracle:dba /zfssa/dbname/*
```

or

```
# dcli -l root -g /home/oracle/dbs_group chown oracle:dba
/zfssa/dbname/*
```

2. Restart the Oracle Database instance to pick up the changes that were made to the `oranfstab` file using one of the options below:

- Restart one instance at a time (rolling upgrade), for example:

```
$ srvctl stop instance -d dbname -i dbname1
$ srvctl start instance -d dbname -i dbname1
$ srvctl stop instance -d dbname -i dbname2
$ srvctl start instance -d dbname -i dbname2
$ srvctl stop instance -d dbname -i dbname3
$ srvctl start instance -d dbname -i dbname3
$ srvctl stop instance -d dbname -i dbname4
$ srvctl start instance -d dbname -i dbname4
$ srvctl stop instance -d dbname -i dbname5
$ srvctl start instance -d dbname -i dbname5
$ srvctl stop instance -d dbname -i dbname6
$ srvctl start instance -d dbname -i dbname6
$ srvctl stop instance -d dbname -i dbname7
$ srvctl start instance -d dbname -i dbname7
$ srvctl stop instance -d dbname -i dbname8
$ srvctl start instance -d dbname -i dbname8
```

- Restart the entire database, for example:

```
$ srvctl stop database -d dbname
$ srvctl start database -d dbname
```

ORACLE®

Protecting Oracle Exadata with the Sun ZFS Storage Appliance: Configuration Best Practices
September 2011, Version 1.2
Updated March 2013, Version 1.3
Author: Jeff Wright
Update: Joe Pichette

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