ORACLE

ZFS STORAGE APPLIANCE

Protecting Oracle Exadata X8 with ZFS Storage Appliance

Configuration Best Practices

White paper / December 2019

ORACLE

DISCLAIMER

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle software license and service agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

Due to the nature of the product architecture, it may not be possible to safely include all features described in this document without risking significant destabilization of the code.

Table of Contents

EXECUTIVE OVERVIEW

Protecting the mission-critical data that resides on Oracle Exadata Database Machine (Oracle Exadata) is a top priority. The Oracle ZFS Storage Appliance family of products is ideally suited for this task due to superior performance, enhanced reliability, extreme network bandwidth, powerful features, simplified management, and cost-efficient configurations.

This paper describes configuration best practices for the Exadata X8 and previous platforms with the ZS7-2 High-End Storage Appliance. The Exadata X8M platform configuration with ZS7-2 Storage Appliance is covered in a companion white paper.

- Extreme Network Bandwidth – With a highly scalable architecture that can be built around InfiniBand or 10 gigabit Ethernet (GbE) connectivity, Oracle ZFS Storage Appliance systems provide the networking performance and redundancy that is required when connecting to Oracle Exadata.
- ZFS-Enhanced Disk Reliability – Hardened ZFS features, such as copy-on-write, metadata check summing, and background data scrubbing, ensure data integrity, detect the presence of even silent data corruption, and correct errors before it is too late.
- Powerful Features – The DTrace Analytics feature of Oracle ZFS Storage Appliance systems allows customers to quickly and effectively identify performance bottlenecks. Oracle Intelligent Storage Protocol, another feature, enables unique Oracle Database–aware storage that simplifies administration and optimizes performance. Technologies and features, such as Replication, snapshots, Cloning, and encryption, provide solutions for any data protection or development/test (DevTest) provisioning challenge that may arise. These are just a few of the enterprise-class features available with Oracle ZFS Storage Appliance systems.
- Simplified Management – Oracle Enterprise Manager Plug-in for Oracle ZFS Storage Appliance has an easy-to-use web management interface that cuts down on training expenses and reduces administration overhead. The innovative, scalable storage pool and fast file system provisioning of Oracle ZFS Storage Appliance systems make managing storage extremely easy.
- Oracle Optimized Compression – Oracle ZFS Storage Appliance systems include advanced inline-compression algorithms that provide superior data reduction while maintaining maximum throughput. This can offload the burden from Oracle Database servers or provide additional data reduction by pairing with compression at the Oracle Database layer. The Oracle Database feature Hybrid Columnar Compression (HCC) is available only on Oracle storage. Oracle ZFS Storage Appliance systems integrate with Oracle Database to provide a full range of compression options that are optimized for specific data usage and workload patterns.
- Reduced Footprint from Deduplication – Oracle ZFS Storage Appliance systems feature inline data deduplication that expands usable storage capacity by eliminating redundant data blocks. The deduplication architecture has been enhanced to leverage read-optimized flash devices to stage the deduplication tables in persistent cache.
- Superior Performance – Oracle ZFS Storage Appliance systems are capable of Oracle Database backup rates of up to 50 TB per hour and restore rates of up to 62 TB per hour. They previously set world records in both the SPC-2 and SPECsfs industry-standard benchmarks. Superior hardware and tighter integration enable backup-and-restore throughput that is higher than competitor storage products.

The following graph shows maximum sustainable backup and restore performance. Physical throughput rates were measured at the network level for backup and restore workloads between Oracle Exadata and Oracle ZFS Storage Appliance systems. For the detailed rates of Oracle ZFS Storage ZS7-2 with specific storage configurations, please see the performance sizing sections later in this document.

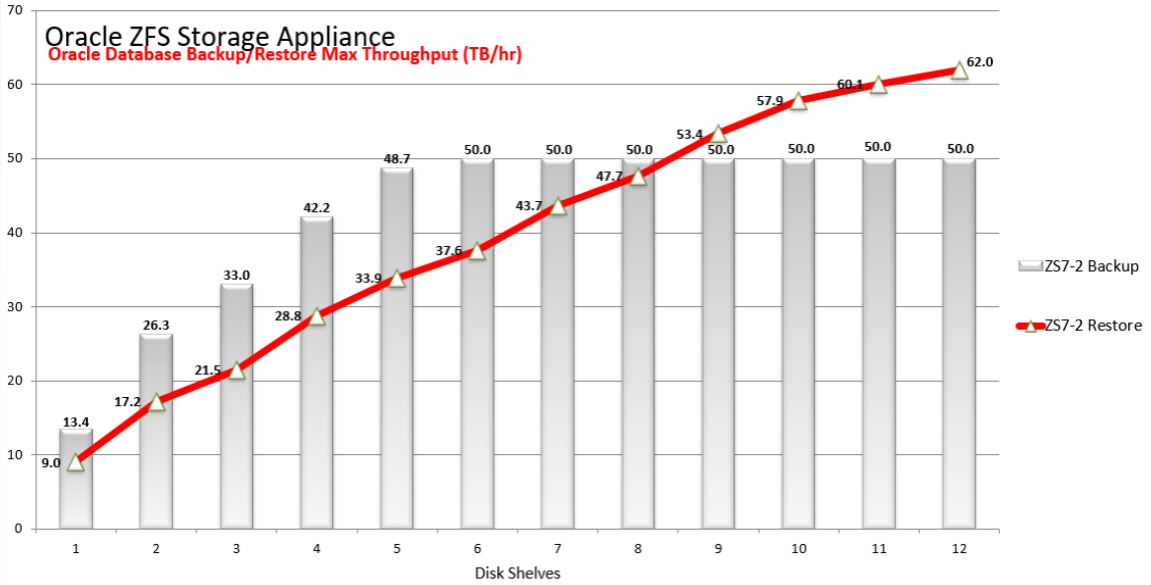

Figure 1. Maximum sustainable backup and restore throughput of Oracle ZFS Storage ZS7-2 High-End System

These are complete, real-world results using Oracle Database 12*c* and a large online transactional processing (OLTP) database that was populated with sample customer data in a sales order-entry schema. Advanced Row Compression, an Oracle Database feature, was used at the database level to align with best-practice recommendations for customers that are running OLTP workloads. These throughput rates were not obtained using a database or input/output (I/O) generator test tool, which can be misleading. Also, they were not projected based on low-level system benchmarks. The backup and restore performance data collected for this document was measured using level 0 backup and restore operations for an otherwise idle Oracle Database.

A full-rack Exadata Storage Expansion X6-2 or later model from Oracle is required to achieve some of these throughput rates, such as 62 TB per hour for a restore operation. Alternatively, the same throughput can be achieved with multiple, smaller Oracle Exadata configurations concurrently utilizing the same Oracle ZFS Storage ZS7-2 appliance. When accounting for database-level compression or incremental backup strategies, effective backup rates that are much higher than the physical rates recorded earlier are routinely observed.

Level 0 full Oracle Database backups to Oracle ZFS Storage ZS7-2 achieved a sustainable throughput rate of up to 50 TB per hour. An effective backup rate of 50 to 120 TB per hour was attained for incremental backups,

with a daily change rate of 5 to 10 percent. Maximum Oracle Database restore rates from Oracle ZFS Storage ZS7-2 measured over 60 TB per hour. To put this into perspective, a 150 TB instance of Oracle Database, which consumes over 300 TB of disk space when mirroring and the `temp` space are accounted for, can be backed up in less than 3 hours using a level 0 backup, or in nearly half that time when performing an incremental backup (assumes a daily change rate of 5 percent). In the event of a failure, that same Oracle Database instance can be restored in under 2.5 hours. Restoring an Oracle Database instance of this size could take days with competitive solutions. When recovery time objective (RTO) requirements are considered, the difference between hours and days is huge. The extreme restore throughput capabilities of Oracle ZFS Storage Appliance systems ensure that critical Oracle Database instances will be recovered and available as quickly as possible.

High performance is an important consideration when choosing a solution to protect Oracle Exadata. The following technologies make it possible for Oracle ZFS Storage Appliance systems to achieve these backup and restore rates:

- InfiniBand Support – Oracle ZFS Storage Appliance systems can be configured with a highly redundant and scalable InfiniBand architecture. This allows for a seamless integration with Oracle Exadata and provides a high-bandwidth, low-latency I/O path that generates relatively little CPU overhead.

- Oracle Recovery Manager Integration – Oracle Recovery Manager (Oracle RMAN) is a highly parallelized application that resides within Oracle Database and optimizes backup and recovery operations. Oracle ZFS Storage Appliance systems are designed to integrate with Oracle RMAN by utilizing up to 3,000 concurrent threads that distribute I/O across many channels spread across multiple controllers. This improves performance dramatically with sequential large-block streaming I/O workloads that are typical for most backup and restore situations.

- Oracle Database's Direct NFS Client feature – The optimized Direct NFS Client feature is an aggressive implementation that allocates individual TCP connections for each Oracle Database process, in addition to reducing CPU and memory overhead, by bypassing the operating system and writing buffers directly to user space.

- 1 MB Record Sizes – Oracle ZFS Storage Appliance systems now enable larger 1 MB record sizes. This reduces the number of input/output operations per second (IOPS) that are required to disk, preserves the I/O size from Oracle RMAN buffers to storage, and improves performance of large-block sequential operations.

- Hybrid Storage Pools – Oracle ZFS Storage Appliance systems have an innovative Hybrid Storage Pool (HSP) architecture that utilizes dynamic storage tiers across memory, flash, and disk. The effective use of dynamic random access memory (DRAM) and enterprise-class software specifically engineered for multilevel storage is a key component that facilitates the superior performance of Oracle ZFS Storage Appliance systems.

The Oracle ZFS Storage Appliance family of products achieves world-record-setting throughput, and has achieved #1 in the price/performance metric of the independently audited SPC-2 industry-standard benchmark. Combine this with the powerful features, simplified management, and Oracle-on-Oracle integrations, and it is easy to see why these systems are a compelling solution for protecting mission-critical data on Oracle Exadata.

INTRODUCTION

Database, system, and storage administrators are faced with a common dilemma when it comes to backup and recovery of Oracle Database instances—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 Oracle ZFS Storage Appliance family of products helps administrators meet these challenges by providing cost-effective and high-bandwidth storage systems that combine the simplicity of the NFS protocol with ZFS-enhanced disk reliability. Through Oracle 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.

Oracle ZFS Storage Appliance systems are easy-to-deploy unified storage systems uniquely suited for protecting data contained in Oracle Exadata. With native InfiniBand and 10/25/40GbE connectivity, they are an ideal match for Oracle Exadata. These high-bandwidth interconnects reduce backup and recovery time, as well as reduce backup application costs and support fees, compared to traditional NAS storage systems. With support for both traditional tiered and incrementally updated backup strategies, Oracle ZFS Storage Appliance systems deliver enhanced storage efficiency that can further reduce recovery time and simplify system administration.

Deploying Oracle ZFS Storage Appliance systems for protecting the mission-critical data that resides on Oracle Database on Oracle Exadata requires that backup window and recovery time objectives (RTOs) be met to ensure timely recovery in the event of a disaster. This paper describes best practices for setting up Oracle ZFS Storage Appliance systems for optimal backup and recovery of Oracle Database and includes specific tuning guidelines for Oracle Exadata.

This paper addresses the following topics:

- Selecting a data protection strategy
- Configuring Oracle ZFS Storage Appliance systems
- Configuring Oracle Exadata
- Best Practices for traditional and incrementally updated backups
- Oracle RMAN backup-and-restore performance sizing for Oracle ZFS Storage ZS7-2 High End (HE) system

SELECTING A DATA PROTECTION STRATEGY

Choosing the best backup strategy for Oracle Database is an important step when building a data protection solution to meet RTOs, recovery point objectives (RPOs), and version retention objectives (VROs). Most importantly, the backups need to be performed quickly and efficiently, with no impact to end-user applications and with minimal resources consumed on production hardware. The type of backup strategy is also relevant when optimizing Oracle RMAN workloads with an Oracle ZFS Storage Appliance system. Specific best practices are presented in the following sections.

Traditional Backup Strategy

A traditional backup strategy is any strategy that uses unmodified full backups or any combination of unmodified level 0, level 1 cumulative incremental, and level 1 differential incremental backups to restore and recover any part of an Oracle Database instance in the event of a physical or logical failure.

The simplest implementation of a traditional backup strategy is periodic, full backup of the entire Oracle Database instance. These backups can be performed while Oracle Database is open and active. Full backups are conducted on a user-defined schedule, which could be weekly or daily. Oracle Database transactional archive logs should be multiplexed, with one copy stored directly on an Oracle ZFS Storage Appliance system. These archive logs are used to recover a restored full backup, and apply all transactions up until the time of the last online redo-log switch before the failure. The RPO of this solution is never more than 20 minutes, assuming that the redo logs are properly sized so that log switches always occur in less than 20 minutes. The VRO would typically dictate that at least two full backups are kept active at all times, with Oracle RMAN automatically expiring and eventually deleting older backups. Daily full backups might be ideal for a small Oracle Database instance with a strict RTO.

A common implementation is a tiered approach that combines incremental level 0 and level 1 backup. Level 0 incremental backups are often taken on a weekly basis, with level 1 differential or cumulative incremental backups performed daily.

Oracle RMAN block-change tracking is used to improve the performance of incremental backup. The level 0 incremental backup scans the entire Oracle Database instance, but level 1 incremental backups use the block-change tracking file to scan only the blocks that have changed since the last backup. This significantly reduces the amount of reads that are required on the Oracle Database instance.

The following figure is an example of an Oracle RMAN traditional backup strategy that utilizes weekly level 0 backups combined with daily level 1 differential backups. Level 0 backups are equivalent in content to a full Oracle Database backup. “Differential” means that each level 1 backup backs up only data that has been changed since the last level 0 or level 1 backup.

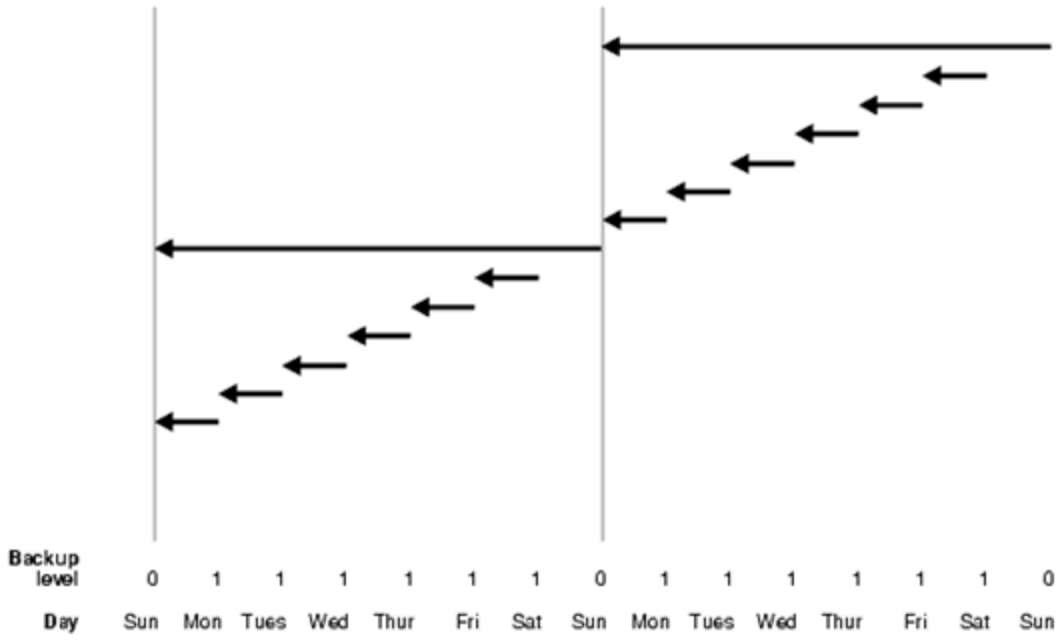

Figure 2. Traditional backup strategy with daily differential incremental backups

The VRO in this example specifies that two weeks of backup data be retained at all times. Level 0 incremental backups are performed every Sunday, with level 1 differential incremental backups performed all other days. The level 1 differential incremental backups are backing up only data that has changed within the last 24 hours. Oracle RMAN block-change tracking is used so the backup operations performed Monday through Saturday are scanning only a small portion of Oracle Database and are sending only a small amount of data over the network for writing to disk. In this example, the archive logs are multiplexed with a copy stored directly on an Oracle ZFS Storage Appliance system.

An entire or partial Oracle Database instance can be restored and recovered to any point within the two-week span. Oracle RMAN restores the most recent level 0 backup before the specified recovery point, restores all subsequent level 1 backups between the level 0 backup and the recovery point, and then applies the archive log transactions that are needed. Restoring data from backup is faster than applying transactions in archive logs. In this example, recovering from a failure on a Monday would be relatively fast and straightforward, while recovering from a failure on a Saturday would be a longer process because five differential backups would need to be restored.

The following figure is an example that utilizes weekly level 0 incremental backups combined with a mix of daily level 1 differential incremental and level 1 cumulative incremental backups. A cumulative incremental backup includes only data that has changed since the last level 0 backup. A cumulative incremental backup consumes more space than a differential backup, but streamlines the restore process.

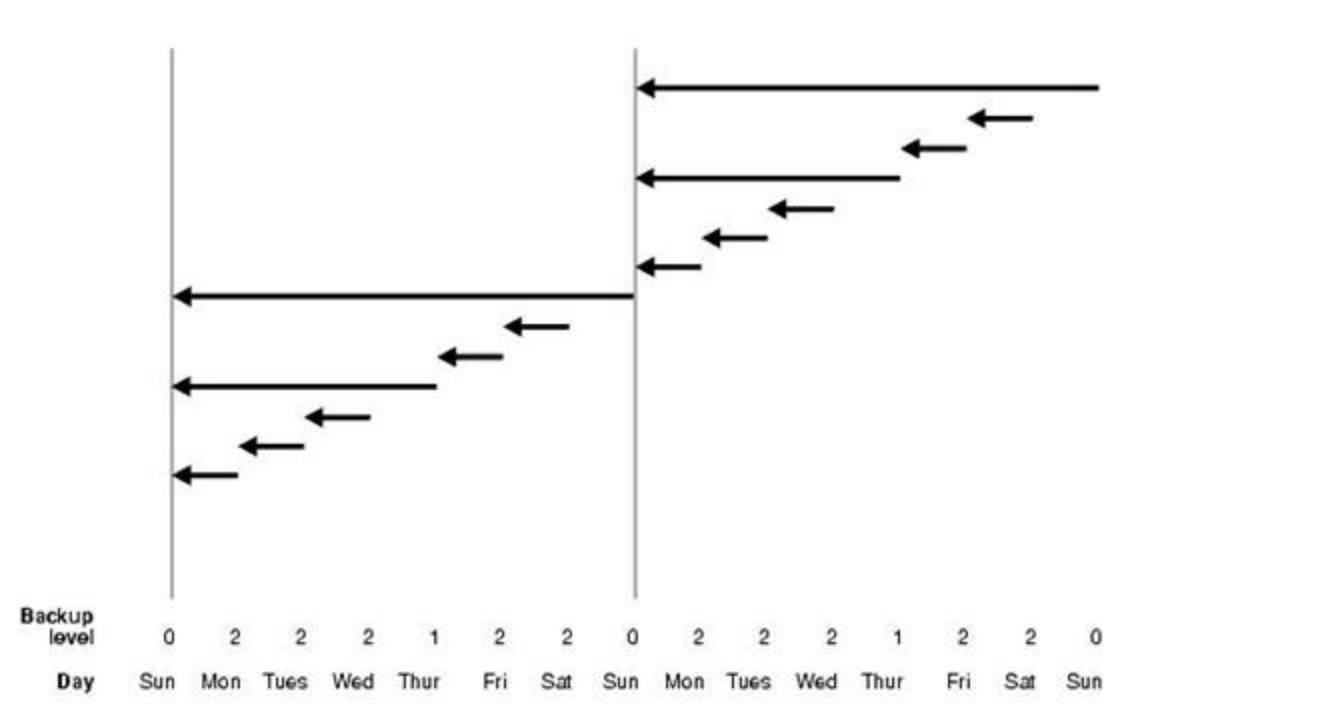

Figure 3. Traditional backup strategy with daily differential and cumulative incremental backups

The difference in this example is that the mid-week backups are now level 1 cumulative incremental backups. This makes for a larger Thursday backup, but streamlines the restore process. For instance, recovering from a failure that happened on a Friday would now require only restoring two backups.

A traditional tiered-incremental backup strategy offers several advantages:

- Increased throughput rates – Up to 50 TB per hour backup and 62 TB per hour restore throughput rates are possible with Oracle ZFS Storage Appliance systems and traditional Oracle RMAN workloads, due the ability to use large record sizes with parallel streaming I/O.
- Faster daily backups with less bandwidth consumption are possible due to the use of block-change tracking.
- Backups consume much less capacity than daily full backups due to the incrementally changed data approach.
- Support for ZFS data deduplication is provided, which provides an excellent opportunity for data reduction on weekly level 0 backups.
- This strategy has synergies with tape archiving and uses native optimization for backup sets.
- This strategy bypasses unused datafile blocks and provides full multisection support.
- The second level 0 backup implicitly validates the previously active level 0 backup, so there is no single point of failure.

Incrementally Updated Backup Strategy

An incrementally updated backup strategy creates an initial level 0 image copy backup. This is an identical image-consistent copy of the datafiles that are stored on an Oracle ZFS Storage Appliance system. All subsequent backups are differential incremental backups that capture only the data that changed since the last backup. These are typically performed on a daily basis with the Oracle RMAN backup set containing only data that was changed within the last 24 hours.

Previous incremental backups are then applied to the image copy backup to roll it forward in time. This streamlines restore operations by providing a level 0 image-consistent copy that trails one or more days behind the active Oracle Database instance. A visual representation of this process is shown in the following figure.

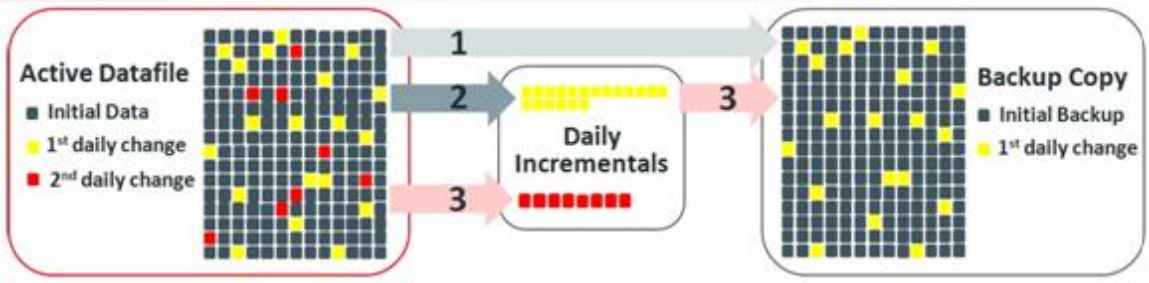

Figure 4. Oracle RMAN incrementally updated backup strategy

In this example, an identical Oracle RMAN job is run every night. On the first night, the job creates a level 0 image copy backup of all the datafiles in Oracle Database. This is represented by the grey blocks. On the second night, it creates a backup set of only the data that was changed within the last 24 hours and stores these in a separate share. This is represented by the yellow blocks. On the third night, it creates a backup set of only the data that was changed within the last 24 hours (red blocks) and then applies the previous night's changed data to the backup copy (yellow blocks). Every subsequent night that the backup is run, it performs the same function of backing up just the data changed within the last 24 hours and applying previously changed data to the backup copy.

Just as with traditional Oracle RMAN strategies, incrementally updated backups can be performed with Oracle Database open and active. Archive logs should be multiplexed with one copy directly stored on an Oracle ZFS Storage Appliance system, ensuring that the RPO of the solution should never be more than 20 minutes.

An incrementally updated backup strategy offers several advantages:

- Limits the number of level 0 backups that are performed
- Simplifies the restore process in most situations
- Provides image copy synergies with provisioning for development and test
- Reduces disk space consumption, if only a single level 0 is maintained

Leveraging Data Deduplication

Oracle ZFS Storage Appliance systems provide inline data deduplication. This capability can significantly expand the usable storage capacity by eliminating redundant data blocks. The deduplication architecture uses an inline, fixed-block implementation that aligns to the record size of the share.

Deduplication requires record sizes of 128 KB or larger. For large sequential I/O workloads, such as backup and restore workloads, it is recommended to use 1 MB record sizes on the share. Deduplication also requires that specialized read flash metadevices be included in the storage pool. These devices are used to store a secondary copy of the deduplication hash table so that reads do not incur a higher latency when they are retrieved from data disks.

There is an opportunity for effective data deduplication on any subsequent incremental level 0 or full Oracle RMAN backup. There is no opportunity for data reduction due to deduplication with incremental level 1 Oracle RMAN backups.

Deduplication should not be used with an incrementally updated backup strategy. This backup strategy typically uses record sizes smaller than the 128 KB restriction. An incrementally updated backup strategy eliminates or significantly reduces the need for subsequent level 0 backups. As a result, it is a poor fit for data deduplication, and deduplication is not beneficial with incremental level 1 backup streams.

Deduplication is recommended with a traditional Oracle RMAN backup strategy, particularly for use cases with long retention policies or frequent level 0 backups. Deduplication provides the most benefit when the opportunity for referenced data is high.

For example, if daily full backups are taken with a two-week retention policy or if weekly level 0 backups are taken with a ten-week retention policy, there is opportunity in both cases to have a high ratio of referenced data to allocated data that can result in significant capacity savings for deduplication. Conversely, a one-week retention policy with weekly level 0 backups would have only a maximum of two level 0 backups active at one time. This provides a theoretical maximum pool deduplication ratio of two times (two referenced copies for one allocated copy), which assumes no changed data. In use cases where the opportunity for referenced data is limited (less than four active level 0 or full backups), high deduplication ratios are unattainable.

Deduplication achieves a synergistic benefit with LZ4 compression. These data services are engineered to work together, and always enabling LZ4 compression when using deduplication increases data reduction benefits.

Deduplication ratios are data-dependent and are influenced by a number of factors, including

- The percentage of the source Oracle Database instance that changes between level 0 (or full) backups
- How the data is ordered; for example, the structure of the schema, partitions, tablespaces, and tables
- Characteristics of the changed workload
- Number of active referenced level 0 or full backups

Combined data reduction due to deduplication and LZ4 compression for Oracle Database backup is typically between 3 times to 6 times. Less-common backup strategies, such as daily full backups, can achieve higher data reduction ratios.

SELECTING TRANSPORTABLE DATA ENCRYPTION (TDE) OR ZFS STORAGE AT-REST ENCRYPTION

Oracle RMAN integrates with the Transportable Data Encryption (TDE) feature in the Oracle Advanced Security option for Oracle Database. During a backup operation, it recognizes if the data is encrypted with TDE and then passes the encrypted data through unchanged. These integrations enhance security and performance.

Because the encrypted state is preserved, the potential exists to match duplicate blocks with previous level 0 or full backups when 1 M sections of an Oracle Database instance remain unchanged. This is true even when Oracle RMAN is configured to run in encrypted mode.

Oracle RMAN compression should not be used with TDE because it causes the data to be unencrypted, compressed, and re-encrypted during the backup session. This creates a major bottleneck for backup performance and places an enormous strain on Oracle Database CPU resources. Not using Oracle RMAN compression is consistent with non-TDE best practices because Oracle RMAN compression, in general, is suitable with Oracle ZFS Storage Appliance systems only when network bandwidth is limited.

There is no opportunity for TDE data reduction due to storage compression. Although Oracle RMAN backups of an Oracle Database instance that uses TDE can still achieve the same benefit from deduplication, the combined data reduction ratio is diminished because the storage compression does not provide a benefit. Tablespaces encrypted with TDE can still benefit from Oracle Database compression, such as that provided by HCC or the Advanced Row Compression feature of Oracle Database.

Using ZFS Storage at-rest data encryption is also an option if you prefer not to encrypt data on the Exadata server itself. If you are using data reduction features described in this paper, you must select AES CCM algorithms over AES GCM algorithms. Otherwise, there is no opportunity for data reduction.

Best Practices for Oracle ZFS Storage Appliance Systems

When protecting Oracle Database on Oracle Exadata with Oracle ZFS Storage Appliance systems, a traditional Oracle RMAN backup strategy is often the preferred option.

Oracle RMAN backup sets benefit from such technologies as unused-block skipping, null-block compression, multisection support, and multiple input files combined into a single backup set. Oracle RMAN traditional backup strategies exclusively utilize large streaming I/O, which allows for 1 M record sizes to be used on ZFS shares and allows for the use of data deduplication. Oracle RMAN uses nonshared buffers, which means that less time is spent in a busy state waiting for another

buffer to clear, and per-channel throughput is higher. Level 0 backups complete faster and consume less space on disk. Throughput rates of restore operations are higher.

In an incrementally updated backup strategy, level 0 backups take longer to complete and consume more space on disk. Level 0 backups and merge operations that apply incremental changes to a level 0 backup utilize shared Oracle RMAN buffers with a reduced per-channel throughput and typically require a smaller record size for ZFS shares. There is limited multisection support. Multisection support is not available with Oracle Database versions prior to Oracle Database 12*c*. Deduplication is not supported in this use case, and restore throughput rates are lower due to the smaller record size.

Additional benefits of traditional backup strategies are more flexibility in designing a backup strategy optimized to specific customer needs and standard integrations to seamlessly archive to tape. Because write-optimized flash is not required, a lower price-point can be achieved when configuring Oracle ZFS Storage Appliance systems for a dedicated, traditional Oracle RMAN backup use case.

An incrementally updated backup strategy has advantages and should be used when the daily change rate is small. Because incremental updates eliminate or reduce the need for level 0 backups on an ongoing basis and only changed data is sent over the network, there are significant advantages to implementing this backup strategy in this situation.

As a general guideline, an incrementally updated backup strategy should be used when both of the following statements are true:

- The source Oracle Database instance is large enough that a weekly level 0 backup would negatively impact network, disk, or server resources
- The daily change rate of the Oracle Database instance is less than 5 percent

An incrementally updated backup strategy offers synergies with ZFS snapshot cloning for DevTest provisioning and can simplify the restore process. However, there is a potential for a single point of failure when maintaining only one level 0 backup, and most implementations provide only a narrow period of time in which restores from a backup can be performed. An n-1 trailing update strategy is standard.

Both incrementally updated and traditional strategies provide inline, optimal source-side data reduction for daily incremental backups due to the use of Oracle RMAN block-change tracking. Only the changed data is transferred over the network. Both strategies satisfy demanding RPOs, and there should never be more than 20 minutes between the failure point and the most recent archive log backup.

Applying archive log transactions is more resource-intensive than restoring backups. For an Oracle Database instance with stringent RTOs, a traditional incremental strategy might struggle to satisfy the RTOs if the failure occurs just before the next level 0 backup. In this case, multiple restore processes would be required, followed by applying redo transactions from the archive logs. In situations such as this, cumulative incremental backups can be substituted for differential backups to streamline the restore process.

The ability to customize backup strategies combined with the superior restore throughput of Oracle ZFS Storage Appliance systems ensures that RTOs are met and exceeded.

CONFIGURING AN ORACLE Z7-2 HIGH END SYSTEM

The following section provides best practices for optimizing an Oracle ZS7-2 High End system to provide Oracle Database protection in an Oracle Exadata environment. The Oracle ZS7-2 HE provides extreme performance and offers maximum levels of scalability, CPU, and DRAM. With the potential to scale up to 3 TB of DRAM, 37.5 TB of write-optimized flash, and 1.4 PB of read-optimized flash, this is a highly scalable platform that can support up to 16 PB of raw storage capacity.

The Oracle ZS7-2 High End system is recommended over the ZS7-2 Mid Range system for optimal RMAN backup performance. In general, Oracle ZS7 MR offers exceptional throughput and redundancy at a low price point. It is well suited for smaller configurations with two to four disk shelve. The ZS7-2 HE is recommended for larger configurations that focus on streaming I/O.

Please refer to oracle.com/storage/nas/index.html for the latest Oracle ZFS Storage Appliance model specifications.

When making configuration decisions for your Exadata/ZS7-2 environment, there are a few factors to consider.

- Large sequential streaming workloads generally do not benefit from the presence of write- or read-optimized flash. Also, while DRAM is critical for achieving superior performance under these conditions, having an excessive amount of DRAM is unnecessary and does not further improve performance. If Oracle ZFS Storage Appliance systems are exclusively used (100 percent) for traditional Oracle RMAN backup and restore workloads with large streaming I/O, a system without write- or read-optimized cache can provide the highest ROI.
- Write- and read-optimized cache is often recommended to achieve good performance and usability with most nonbackup Oracle Database I/O, incrementally updated backup workloads, cloning for DevTest provisioning, and many other mixed I/O scenarios. Having an Oracle ZFS Storage Appliance system with a significant amount of write-optimized flash increases flexibility for effectively using different workload types. This might be important for current or future activity planning.
- If running direct transactional Oracle Database workloads is a primary focus of the system, having a large amount of DRAM and CPU resources is a significant benefit.

Choosing the Correct Disk Shelves

Oracle ZFS Storage Appliance systems offer two options for disk shelves, both with similar price points. Oracle Storage Drive Enclosure DE3-24C features high-capacity 14 TB disks, and Oracle Storage Drive Enclosure DE3-24P features high-performance 1.2 TB disks. Each disk shelf contains 24 disks, and both can be configured with the same write-optimized flash, read-optimized flash, and deduplication metadevice options. (Up to four disks per disk shelf can be replaced with solid-state drive [SSD] write-flash accelerators.) Oracle ZS7-2 HE can be customized based on disk shelf and write-optimized flash requirements. The following table provides disk shelf details.

TABLE 1. DISK SHELF DETAILS

Disk Shelf	Size/Disk	RPM	IOPS/Disk	MB/sec per Disk	Rack Units
Oracle Storage Drive Enclosure DE3-24P	1.2 TB	10 K	160	170	2
Oracle Storage Drive Enclosure DE3-24C	14 TB	7.2 K	120	200	4

Note: All-SSD disk shelves are also an option for creating all-flash storage pools. They are not included here because they do not provide the recommended ROI for data protection use cases.

The high-capacity Oracle Storage Drive Enclosure DE3-24C disk shelf is recommended when protecting Oracle Exadata with Oracle ZFS Storage Appliance systems. Its larger capacity and slightly higher throughput rate provide a significant advantage for most backup use cases. The Oracle Storage Drive Enclosure DE3-24P disk shelf might be considered in situations where higher IOPS and lower latency would be a significant advantage or if rack space would be a limiting factor and the desire would be to maximize performance in a small, partial-rack configuration.

Choosing a Storage Profile

When a storage profile is selected to protect Oracle Exadata, mirrored, single-parity, and double-parity profiles are all worthy of consideration. The following table provides a comparison of the storage profiles.

TABLE 2. STORAGE PROFILE COMPARISON

	Usable Capacity	Advantages	Negatives
Mirrored	42.2%	» *Restore performance* » *Maximum protection* » *Maximum flexibility*	Costly
Single Parity	69.3%	» *Backup performance* » *Moderate flexibility*	Limited redundancy
Double Parity	76.7%	» *Streaming performance* » *Most efficient*	Limited IOPS

Note: Useable capacity accounts for raw capacity lost due to parity, spares, and file system overhead, as well as small amounts of space lost on each disk due to operating system (OS) overhead, drive manufacturer overhead, and scratch space reservations. This will vary slightly depending on the size of the storage pool; this example assumes a configuration with four disk shelves.

MIRRORED

A mirrored profile is a frequently recommended storage profile due to its strong redundancy trait and robust performance, particularly for restore processes. Because it generates twice as many virtual devices (vdevs) as a single-parity implementation, a mirrored storage pool is capable of handling far more IOPS. This gives it the flexibility to perform well with large sequential I/O, such as traditional Oracle RMAN workloads, and also achieve exceptional performance with workloads that generate small random I/O, such as direct Oracle Database OLTP transactions.

The negatives of choosing a mirrored profile are that it consumes more disk space than the other two options, and it generates more internal bandwidth on writes, which would have an impact when Serial Attached SCSI (SAS) or Peripheral Component Interconnect (PCI) bandwidth are limiting factors.

A mirrored profile is recommended when there is an emphasis on achieving optimal performance for restore processes or IOPS-intensive workloads. It can provide the shortest duration RTO possible for Oracle Database running critical business applications.

A mirrored profile is recommended when the focus is on incrementally updated backup strategies, cloning for DevTest provisioning, or direct Oracle Database workloads. With these use cases, if Oracle Database runs an OLTP workload (characterized by mostly small transactions with a focus on changing and reading existing rows of data) or has an element of write transactions dispersed throughout the day, this profile generates small random I/O that places an IOPS load on the storage pool. A mirrored profile is best suited to handle heavy IOPS.

SINGLE PARITY

Single parity is a middle-of-the-road option. It provides optimal backup performance for traditional Oracle RMAN workloads and is an attractive option when usable capacity concerns would make a mirrored profile a poor fit. However, single parity might not provide the desired level of redundancy for certain use cases.

Single parity implements a narrow 3+1 stripe width and utilizes powerful ZFS features to provide exceptional performance with large streaming I/O operations, but also enough flexibility to handle some random or smaller I/O workloads.

Single parity is a good fit when there is an emphasis on backup performance; when usable capacity concerns would make a mirrored profile a poor fit; and when use cases such as incrementally updated backups, DevTest provisioning, and direct

Oracle Database workloads might be used sparingly, but are not a primary focus. Because single parity uses a narrow-stripe width, it still generates a fair amount of vdevs (half as many as with a mirrored profile) and has the flexibility to handle some workloads that generate nonstreaming large I/O operations. However, an IOPS-intensive workload, such as an incrementally updated backup strategy for an OLTP Oracle Database instance, should utilize a mirrored profile for datafile copies.

DOUBLE PARITY

Double parity provides the best usable capacity and performs as well as single parity for large streaming I/O, which is typical for traditional Oracle RMAN workloads. It accomplishes this by utilizing a wide stripe width. The width varies at the time of storage pool creation depending on the number of disks in the configuration, but it ranges up to 14 disks. As a result, the number of vdevs in a double-parity storage pool is far fewer than with mirrored or single parity profiles. The ability to handle IOPS-intensive workloads is severely diminished.

Double parity is the recommended storage profile when deduplication is enabled.

Double parity is recommended when Oracle ZFS Storage Appliance systems are 100 percent dedicated to large sequential workloads, such as traditional Oracle RMAN backup and restore workloads. It is not advisable for use cases such as cloning for DevTest provisioning or utilizing an incrementally updated backup strategy. Mirrored or single-parity profiles are more flexible for handling additional use cases that might result in heavier disk IOPS with lower latencies. The following figure reflects raw disk capacity distribution for different storage profiles.

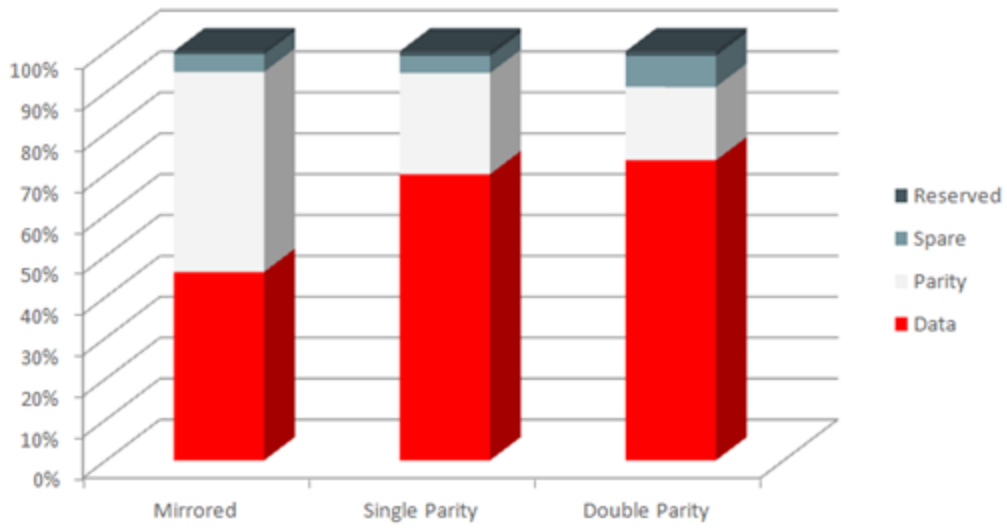

Figure 5. Raw disk capacity distribution

Configuring the Storage Pools

In most situations, it is recommended to configure a single storage pool on each controller. Each storage pool should be configured with half of the available data disk drives in each Oracle Storage Drive Enclosure DE3-24 disk shelf. This allows for maximum performance and redundancy.

It is recommended to select the No Single Point of Failure (NSPF) option when configuring the storage pool. This ensures that the loss of an entire disk shelf does not compromise the availability of data.

To enable NSPF, a minimum of two disk shelves is needed for mirrored profiles and a minimum of three disk shelves is needed for double-parity profiles.

Using Write-Flash Accelerators and Read-Optimized Flash

Oracle ZFS Storage Appliance systems provide a unique, cost-effective, high-performance storage architecture that is built on a flash-first HSP model. A performance-on-demand approach allows optional write-flash accelerators and read-optimized flash devices to be configured into the storage pool.

Utilizing flash-based caching to unlock the power of Oracle ZFS Storage Appliance systems' HSP architecture is critical to achieving optimal performance with transactional or mixed I/O workloads. However, traditional Oracle RMAN backup and restore workloads do not benefit from the presence of flash devices. These workloads generate streaming 1 MB I/O operations, and datasets are very large (often greater than 10 TB). Throughput of the system during backups is typically determined by the rate at which data in memory can be synchronized to the storage pool. Additionally, Oracle RMAN backups are a low priority to populate level 2 read cache. The datasets are large, and the frequency of restore is low.

Because SSDs are significantly more costly than HDDs, flash devices do not provide the best ROI for Oracle ZFS Storage Appliance configurations used exclusively for traditional Oracle RMAN workloads. In these environments, more benefits are realized by adding additional HDD disk shelves, which increases performance and capacity. However, if Oracle Database cloning for DevTest provisioning is used or if an incrementally updated backup strategy is implemented, flash devices might be recommended or even required. The best practices section for traditional Oracle RMAN workloads and incrementally updated backup strategies provides more-detailed guidelines on when to use flash devices.

Selecting Network Connectivity

Oracle ZFS Storage Appliance systems can be configured with either InfiniBand or 10/25/40 GbE connectivity for protecting data.

When connecting an Oracle ZFS Storage Appliance system to the Oracle Exadata InfiniBand fabric, any available ports on the two leaf switches in each Oracle Exadata rack can be utilized. Ports 5B, 6A, 6B, 7A, 7B, 8B, and 12A are available in full-rack Oracle Exadata Storage Expansion X6-2 configurations. In partial-rack configurations, additional ports are available.

Oracle ZFS Storage Appliance systems can integrate seamlessly by connecting to the two InfiniBand leaf switches (Oracle's Sun Datacenter InfiniBand Switch 36) that are preconfigured in the Oracle Exadata rack.

The Oracle RMAN backup and restore throughput rates and performance-sizing data presented in this document were collected in a lab using InfiniBand connectivity.

Oracle RMAN backup and restore throughput rates for 10 GbE configurations are not measurably different for smaller configurations that are disk-limited. Large configurations that are limited by system resources will experience a small decrease in throughput of less than 10 percent when switching to 10 GbE. Configurations that are network bandwidth–limited to begin with and do not add additional active 10 GbE links will experience a more significant decrease in throughput.

To optimize availability and tolerate the loss of an InfiniBand switch in typical configurations, port 1 on each host channel adapter (HCA) should be connected to the lower-leaf switch, and port 2 should be connected to the upper-leaf switch. Detailed information is documented in Oracle's Application Integration Engineering (AIE) white paper "Configuring a Single Oracle ZFS Storage Appliance into an InfiniBand Fabric with Multiple Oracle Exadata Database Machines."

In some environments, 10/40 GbE is a better choice. Some examples of these situations include distance limitations that make InfiniBand deployments prohibitive or backing up five or more isolated Oracle Exadata systems to a single Oracle ZS7 system.

Choosing Direct NFS Client

The Direct NFS Client feature of Oracle Database is highly recommended for all Oracle RMAN workloads between Oracle Exadata and Oracle ZFS Storage Appliance systems, and it is required to achieve optimal performance.

Direct NFS Client is a custom NFS client that resides within the Oracle Database kernel and provides several key advantages:

- Significantly reduces system CPU utilization by bypassing the OS and caching data just once in user space with no second copy in kernel space
- Boosts parallel I/O performance by opening an individual TCP connection for each Oracle Database process
- Distributes throughput across multiple network interfaces by alternating buffers to multiple IP addresses in a round-robin fashion
- Provides high availability (HA) by automatically redirecting failed I/O to an alternate address

These advantages enable increased bandwidth and reduced CPU overhead.

No additional steps are required on Oracle ZFS Storage Appliance systems to enable Direct NFS Client.

ORACLE INTELLIGENT STORAGE PROTOCOL

Oracle Intelligent Storage Protocol, a feature of Oracle ZFS Storage Appliance systems, was introduced to interact with Direct NFS Client in Oracle Database 12*c*. It enables Oracle Database–aware storage by dynamically tuning record size and synchronous write bias on Oracle ZFS Storage Appliance systems. This simplifies the configuration process and reduces the performance impact of configuration errors. Hints are passed from the Oracle Database kernel to the Oracle ZFS Storage Appliance system. These hints are interpreted to construct a workload profile to dynamically optimize storage settings.

Oracle Intelligent Storage Protocol is an optional protocol that requires NFSv4 and SNMP. In the current implementation, a properly configured environment that adheres to the best practices in this document performs equally well without Oracle Intelligent Storage Protocol. For instructions on how to enable Oracle Intelligent Storage Protocol, see My Oracle Support document 1943618.1 "Oracle ZFS Storage Appliance: How to Enable Oracle Intelligent Storage Protocol (OISP)."

Configuring IP Network Multipathing

IP network multipathing (IPMP) groups are recommended to provide full HA redundancy. Direct NFS Client can provide a level of HA, but currently relies on the kernel NFS mount for opening or creating files. IPMP is required to provide full HA in all situations.

The example illustrated in the following figure assumes that two InfiniBand HCAs are installed in each Oracle ZFS Storage Appliance controller and the configuration is set up as follows:

1. Create InfiniBand datalinks for ibp0, ibp1, ibp2, and ibp3; set the partition key to ffff and the link mode to **Connected** mode.
2. Create network interfaces for ibp0, ibp1, ibp2, and ibp3; use the address 0.0.0.0/8 for each network interface.
3. Create the first IPMP network interface (ib-ipmp-controller1) using ibp0 and ibp3 and with both ports set as active; create two different IP addresses for this link for optimized performance with Direct NFS Client; the number of IP addresses should match the number of active InfiniBand interfaces.
4. Create the second IPMP network interface (ib-ipmp-controller2) using ibp1 and ibp2 and with both ports set as active; create two different IP addresses for this link for optimized performance with Direct NFS Client; the number of IP addresses should match the number of active InfiniBand interfaces. This IPMP group is owned by the second controller. It can be created directly using the browser user interface (BUI) on the second controller or, if you are configuring in a takeover mode, it can be created on the first controller and ownership can be changed in BUI **Configuration > Cluster** screen prior to a failback operation.

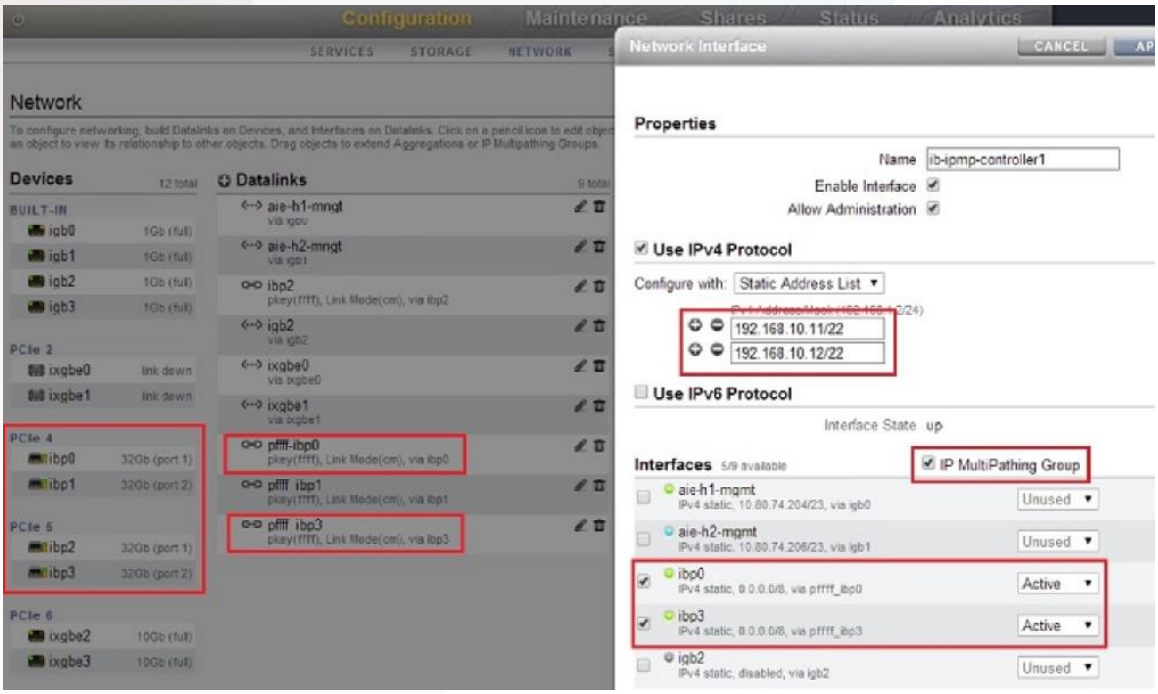

Figure 6. IPMP configuration with two HCAs per controller

Please note that active/active IPMP on Oracle ZFS Storage Appliance systems requires as many IP addresses as active links to process traffic on all of the active links. To apply this in Oracle RMAN backup and restore applications, you should use an oranfstab file to configure Direct NFS Client load spreading over multiple network interfaces.

If four InfiniBand or 10 GbE cards are installed in each controller, it is recommended to configure two IPMP network interfaces on each controller, as shown in the following figure. Each IPMP group includes two active interfaces that are spread across different cards, PCI bridges, and network switches for optimal redundancy. Using two groups of two as opposed to one group of four reduces the overhead while still providing full HA redundancy.

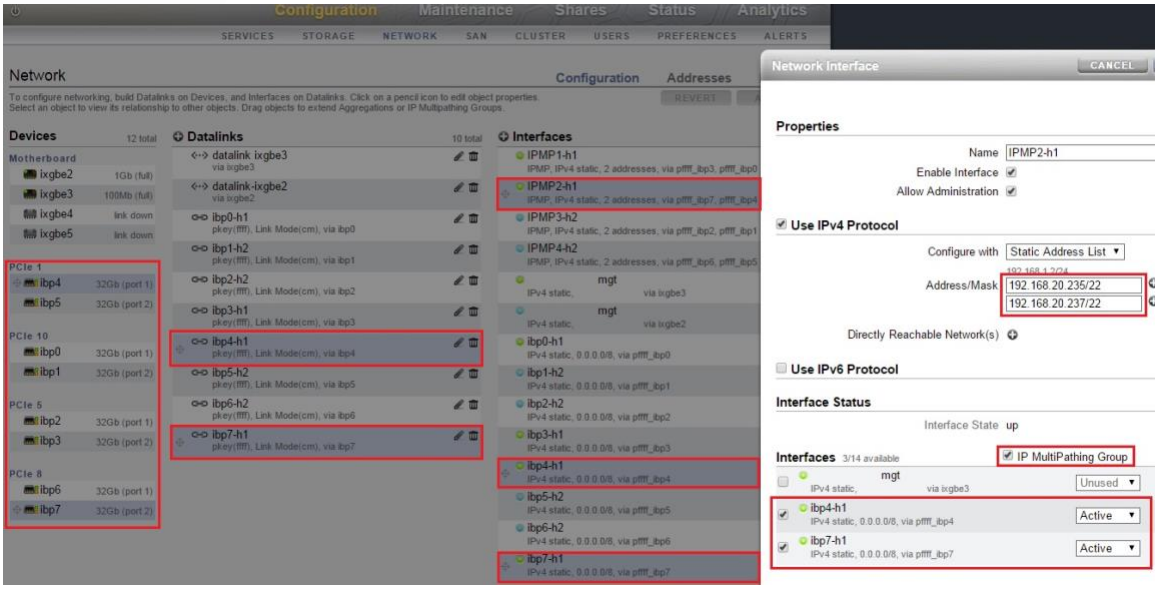

Figure 7. IPMP configuration with four HCAs per controller

Enable adaptive routing for the multihoming policy when using IPMP to ensure that outbound traffic from an Oracle ZFS Storage Appliance system is balanced over the network links and IP addresses. Access the BUI, select **Configuration > Network > Routing**, and then select option **multihoming=adaptive**.

CONFIGURING ORACLE EXADATA

The following section provides best practices for optimizing Oracle Exadata when using Oracle ZFS Storage Appliance systems to perform Oracle Database protection.

Choosing NFSv3 or NFSv4

NFSv3 and NFSv4 are both excellent protocol choices when using Direct NFS Client with Oracle Exadata and Oracle ZFS Storage Appliance systems. NFSv4 implements several enhancements, such as a stronger security model, file locking managed within the core protocol, and delegations that can help improve the accuracy of client-side caching.

NFSv4 incurs a more significant overhead. However, Direct NFS Client workloads utilize a large packet size, and any potential performance impact is negligible in this environment. NFSv4 is required, along with Oracle Database 12*c*, to enable Oracle Intelligent Storage Protocol.

ENABLING NFS ON ORACLE EXADATA

When only NFSv4 is used, no additional steps are necessary prior to configuring and mounting the share(s) on Oracle Exadata for Oracle Database.

If support for NFSv3 or NFSv2 connectivity is desired, additional remote procedure call (RPC) mounting and locking protocols are needed. To enable these services on Oracle Exadata versions based on Oracle Linux 6 (`cat /etc/redhat-release`), `rpcbind`, `nfslock` and `nfs` services should be started and made persistent across reboots. Additionally, `rpcbind` requires read access to `/etc/hosts.allow` and `/etc/hosts.deny`.

The following example uses the Oracle Exadata command `dcli` to allow read access of these files and to enable `rpcbind`, `nfslock` and `nfs` services on all Oracle Database servers.

```
# 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

# dcli -l root -g /home/oracle/dbs_group chkconfig rpcbind on

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

# dcli -l root -g /home/oracle/dbs_group chkconfig nfslock on

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

# dcli -l root -g /home/oracle/dbs_group chkconfig nfs on

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

NFS MOUNT OPTIONS

If shares are dedicated to traditional Oracle RMAN backup, utilize the following mount options:

```
rw,hard,rsize=1048576,wsize=1048576,tcp,vers=4,timeo=600
```

If shares are utilized for incrementally updated backups or there is a potential to utilize Oracle RMAN switch-to-copy for immediate Oracle Database recovery, utilize the following mount options:

```
rw,hard,rsize=32768,wsize=32768,tcp,actimeo=0,vers=3,timeo=600
```

Direct NFS Client does not utilize NFS mount options. However, setting the proper mount options is recommended to be in compliance with Oracle Database requirements and to improve performance and functionality if Direct NFS Client is not available and the system reverts to NFS.

Backup shares are required to be mounted on the Oracle Database nodes running the Oracle RMAN backup and restore jobs. However, it is recommended to mount the shares on all Oracle Database servers so that every node can execute a backup or a restore operation. This is particularly beneficial in failure scenarios when Oracle Database instances and Oracle RMAN backup services might be automatically migrated to other Oracle Database servers that are not normally "active on."

Configuring Direct NFS Client

In Oracle Database 12*c*, Direct NFS Client is enabled by default. In Oracle Database 11*g*, Direct NFS Client is enabled on a single Oracle Database node with the following command:

```
$ make -f $ORACLE_HOME/rdbms/lib/ins_rdbms.mk dnfs_on
```

The Oracle Exadata command `dcli` is used to simultaneously enable Direct NFS Client on all Oracle Database nodes:

```
$ dcli -l oracle -g /home/oracle/dbs_group make -f \
   $ORACLE_HOME/rdbms/lib/ins_rdbms.mk dnfs_on
```

The Oracle Database instance should be restarted after enabling Direct NFS Client.

Confirm that Direct NFS Client is enabled by checking the Oracle Database alert log for an Oracle Disk Manager (ODM) message after Oracle Database startup:

```
Oracle instance running with ODM: Oracle Direct NFS ODM Library Version 3.0
```

Direct NFS Client activity can also be confirmed by SQL query:

```
SQL> select * from v$dnfs_servers;
```

Direct NFS Client can be disabled with the following command:

```
$ make -f $ORACLE_HOME/rdbms/lib/ins_rdbms.mk dnfs_off
```

For a complete list of recommended patches, see My Oracle Support document 1495104.1 "Recommended Patches for Direct NFS Client."

CREATING AN ORANFSTAB FILE

The oranfstab file is required to achieve the published backup and restore rates. The file is created in `$ORACLE_HOME/dbs/oranfstab` and applies to all Oracle Database instances that share `ORACLE_HOME`.

The oranfstab file configures load spreading of Direct NFS Client connections over multiple addresses on an Oracle ZFS Storage Appliance system (represented by "path") or multiple addresses on Oracle Exadata for Oracle Database (represented by "local"). Load balancing over multiple interfaces reduces or eliminates two possible system bottlenecks: network interface bandwidth and TCP/IP buffering.

It is recommended to match the number of path IP addresses defined in the oranfstab file with the number of active interfaces on the Oracle ZFS Storage Appliance controller. Also, it is recommended to match the number of local IP addresses defined in the oranfstab file with the number of active interfaces on Oracle Exadata for Oracle Database.

Older Oracle Exadata models, such as Oracle's Exadata Storage Server X3-2 or Exadata Storage Server X2-2, have just a single active interface for the data path (bondib0) on each compute node.

However, newer models, such as Oracle's Exadata Storage Server X5-2 and Exadata Storage Server X6-2, have the ability to operate in an active/active role with independent IP addresses defined on both ib0 and ib1. Some models, such as Oracle's Exadata Storage Server X4-8, have four active data path interfaces. In these scenarios, multiple local IP addresses need to be defined in the oranfstab file to enable Direct NFS Client to utilize all available paths.

Here is an example of an oranfstab file for use with an Oracle ZFS Storage Appliance system with clustered controllers that is hosting two storage pools, one per controller, and one share per pool. The shares are defined with the NFS export path on the Oracle ZFS Storage Appliance system and the local mount point owned by the `oracle` user. Shares are listed beneath the path addresses with which they are associated. Oracle Exadata is an Exadata Storage Server X3-2 model, and each Oracle ZFS Storage Appliance controller is configured with two active interfaces. Note that the local address is specific to each Oracle Exadata compute node so that oranfstab files on additional compute nodes will define a different local address. In this example, Direct NFS Client utilizes NFSv3, which is the default protocol.

```
server: zfs-storage-a

local: 192.168.10.1 path: 192.168.10.50

local: 192.168.10.1 path: 192.168.10.51

export: /export/backup1 mount: /zfs/backup1

server: zfs-storage-b

local: 192.168.10.1 path: 192.168.10.52

local: 192.168.10.1 path: 192.168.10.53

export: /export/backup2 mount: /zfs/backup2
```

Here is a similar example that uses an Exadata Storage Server X6-2 model with active/active InfiniBand interfaces. Note that a different local address is defined so that the workload is spread across all available paths. A second share named `copy` is created on each pool and presented to Direct NFS Client. In this example, Direct NFS Client utilizes NFSv4 by defining an `nfs_version` parameter for each server.

```
server: zfs-storage-a

local: 192.168.10.1 path: 192.168.10.50

local: 192.168.10.2 path: 192.168.10.51

nfs_version: nfsv4

export: /export/backup1 mount: /zfs/backup1

export: /export/copy1 mount: /zfs/copy1

server: zfs-storage-b

local: 192.168.10.1 path: 192.168.10.52

local: 192.168.10.2 path: 192.168.10.53

nfs_version: nfsv4

export: /export/backup2 mount: /zfs/backup2

export: /export/copy2 mount: /zfs/copy2
```

Here is an example of an oranfstab file on an Exadata Storage Server X4-8 model connected to a clustered Oracle ZFS Storage Appliance system with only one active network interface configured on each controller. This oranfstab configuration spreads the load across all four active interfaces on the Oracle Exadata compute node going to the single active path address on each controller.

```
server: zfs-storage-a

local: 192.168.10.1 path: 192.168.10.50
```

```
local: 192.168.10.2 path: 192.168.10.50
local: 192.168.10.3 path: 192.168.10.50
local: 192.168.10.4 path: 192.168.10.50
export: /export/backup1 mount: /zfs/backup1
server: zfs-storage-b
local: 192.168.10.5 path: 192.168.10.51
local: 192.168.10.6 path: 192.168.10.51
local: 192.168.10.7 path: 192.168.10.51
local: 192.168.10.8 path: 192.168.10.51
export: /export/backup2 mount: /zfs/backup2
```

Finally, here is an example of an oranfstab file on an Exadata Storage Server X4-8 model connected to a clustered Oracle ZFS Storage Appliance system with four active network interfaces configured on each controller.

```
server: zfs-storage-a
local: 192.168.10.1 path: 192.168.10.50
local: 192.168.10.2 path: 192.168.10.51
local: 192.168.10.3 path: 192.168.10.52
local: 192.168.10.4 path: 192.168.10.53
export: /export/backup1 mount: /zfs/backup1
export: /export/stage1 mount: /zfs/stage1
server: zfs-storage-b
local: 192.168.10.5 path: 192.168.10.54
local: 192.168.10.6 path: 192.168.10.55
local: 192.168.10.7 path: 192.168.10.56
local: 192.168.10.8 path: 192.168.10.57
export: /export/backup2 mount: /zfs/backup2
export: /export/stage2 mount: /zfs/stage2
```

Note that if the local address is not specified in the oranfstab file, Direct NFS Client utilizes the first routable address/interface it discovers.

Examples in this section are geared toward InfiniBand due to synergies connecting to the Oracle Exadata native InfiniBand infrastructure, but the oranfstab syntax is agnostic and is applicable to 10 GbE as well. For a complete reference to the options available in the oranfstab file, consult the administration guide for your specific version of Oracle Database software.

Configuring Oracle RMAN Backup Services

Oracle RMAN backup services should be created and used to balance Oracle RMAN workloads across all Oracle Exadata compute nodes. Spreading a backup across multiple Oracle Real Application Clusters (Oracle RAC) nodes improves performance, increases parallel tasks, and reduces utilization load on any single component. Oracle RMAN backup services

are automatically migrated to other Oracle Exadata servers for Oracle Database in the Oracle RAC cluster when the preferred instance is unavailable.

The following configuration example assumes a half-rack Exadata Storage Expansion X6-2 with four Oracle RAC nodes. The Oracle Database name is `hulk`.

The syntax is as follows:

```
srvctl add service -d <db_name> -r <preferred instance> -a <alternate instance(s)> -s <name for newly created service>
```

```
[oracle@ex01db01 ~]$ srvctl add service -d hulk -r hulk1 -a hulk2,hulk3,hulk4 -s hulk_bkup1
[oracle@ex01db01 ~]$ srvctl add service -d hulk -r hulk2 -a hulk1,hulk3,hulk4 -s hulk_bkup2
[oracle@ex01db01 ~]$ srvctl add service -d hulk -r hulk3 -a hulk1,hulk2,hulk4 -s hulk_bkup3
[oracle@ex01db01 ~]$ srvctl add service -d hulk -r hulk4 -a hulk1,hulk2,hulk3 -s hulk_bkup4


[oracle@ex01db01 ~]$ srvctl start service -d hulk -s hulk_bkup1
[oracle@ex01db01 ~]$ srvctl start service -d hulk -s hulk_bkup2
[oracle@ex01db01 ~]$ srvctl start service -d hulk -s hulk_bkup3
[oracle@ex01db01 ~]$ srvctl start service -d hulk -s hulk_bkup4
[oracle@ex01db01 ~]$ srvctl status service -d hulk
Service hulk_bkup1 is running on instance(s) hulk1
Service hulk_bkup2 is running on instance(s) hulk2
Service hulk_bkup3 is running on instance(s) hulk3
Service hulk_bkup4 is running on instance(s) hulk4
```

When Oracle Database is restarted, the Oracle RMAN backup services should be rebalanced. This can be accomplished with the following command:

```
[oracle@ex01db01 ~]$ srvctl stop service -d hulk; srvctl start service -d hulk
```

InfiniBand Best Practices

For optimal IP over InfiniBand (IPoIB) performance and stability, ensure that the latest fixes and best practices are in place. For a complete list and description of InfiniBand-related fixes, see My Oracle Support document 2087231.1 "Guidelines when Using ZFS Storage in an Exadata Environment."

Preparing Oracle Database for Backup

ARCHIVELOG MODE

Archiving of the online redo logs is enabled when Oracle Database is configured to operate in "archivelog" mode. Benefits of using archivelog mode include

- Protection is provided in the event of media failure.
- Oracle Database transactions that occurred after the most recent backup can be recovered.
- Backups can be performed while Oracle Database is open and active.
- Inconsistent backups can be used to restore Oracle Database.

It is recommended that Oracle Database run in archivelog mode, and the archivelogs should be multiplexed with one copy on Oracle Exadata storage and one copy on the Oracle ZFS Storage Appliance system.

BLOCK-CHANGE TRACKING

Block-change tracking is an Oracle RMAN feature that records changed blocks within a datafile. The level 0 backup scans the entire datafile, but subsequent incremental backups rely on the block-change tracking file to scan just the blocks that have been marked as changed since the last backup.

It is recommended to enable block-change tracking to improve performance for incremental backups. If the chosen backup strategy includes only full or level 0 backups, block-change tracking should not be enabled.

BEST PRACTICES FOR TRADITIONAL ORACLE RMAN BACKUP

This section details the recommended configuration steps necessary to achieve optimal performance and functionality when using an Oracle RMAN traditional backup strategy to protect Oracle Exadata with an Oracle ZFS Storage Appliance system. Best practices presented in this section specifically correspond to traditional Oracle RMAN backup strategies, while recommendations in previous sections correspond to general applications.

Configuring the Network Shares

A single share per storage pool is recommended when backing up Oracle Database using a traditional Oracle RMAN strategy. A storage pool is typically configured on each controller for maximum performance and to fully utilize hardware resources on both controllers. Oracle Database can then be backed up using two shares, with one owned by each controller.

Alternatively, when backing up multiple Oracle Database instances concurrently or even multiple Oracle Exadata systems, an individual Oracle RMAN backup operation can be configured to use only a single share owned by one controller, with other Oracle RMAN backup workloads effectively utilizing the other controller. Full HA redundancy is still provided in this configuration because of the ability to fail over storage resources to the other controller.

Share access permissions should be aligned to match the user identity (ID) of the `oracle` user and the group ID of the `dba` group. A standard Oracle Exadata configuration is 1001 and 1002, respectively. In most configurations, the directory permissions are set to rwxr-x---.

RECORD SIZE

The ZFS record size is a setting that is configured at the share level and influences the size of back-end disk I/O. Optimal settings depend on the network I/O sizes used by the application—in this case, Oracle RMAN. Traditional Oracle RMAN workloads with Direct NFS Client generate large 1 MB writes and reads at the network layer. In this case, a 1 MB record-size setting should be used. The ability to use large record sizes has significant advantages, such as increased throughput performance, which is critical for bandwidth-intensive workloads. Other benefits include reduced utilization of controller CPU resources.

In recent years, HDD capacities have grown as quickly as ever, yet the IOPS these disks are capable of delivering has leveled off. Oracle RMAN workloads often generate datasets on the TB scale, with only a small frequency of read-backs. As such, caching is not an optimal solution for handling IOPS. Maximizing the throughput and limiting the IOPS to disk are important factors for achieving the best performance from the backup solution. Oracle RMAN traditional backup strategies enable this by delivering large, multichannel network I/O that greatly benefits from large record sizes on the ZFS share.

SYNCHRONOUS WRITE BIAS

Synchronous write bias is a share setting that controls behavior for servicing synchronous writes. It can be optimized for latency or throughput.

All writes are initially written to the ZFS adaptive replacement cache (ARC), regardless of whether they are asynchronous, synchronous, latency-optimized, or throughput-optimized. Also, all writes are copied from the ARC to the storage pool. An asynchronous write returns an acknowledgement to the client after the write to ARC is complete. When synchronous writes are optimized for throughput, an acknowledgement is not returned until the write is copied to the storage pool. When synchronous writes are optimized for latency, an additional copy is written to persistent storage so that acknowledgements can be returned to the client faster. When write-optimized flash is configured in the storage pool, it is used as the persistent storage for latency-sensitive synchronous writes.

The synchronous write bias share setting should be configured for throughput. There are two reasons:

- Write-optimized flash is a limited resource and is much more expensive than HDDs. It provides a major boost to latency-sensitive writes, but traditional Oracle RMAN backups are bandwidth-sensitive, large 1 M streaming writes. If flash is not present in the storage pool configuration, a lower price point can be achieved by not adding it. HDDs can be used to fill the slots that flash can occupy and provide additional throughput and capacity in the process. If write-optimized flash is configured in the storage pool, it is a shared resource that other network shares and LUNs can access, and it should be reserved for latency-sensitive workloads where it can provide benefit.
- Setting the synchronous write bias to the latency setting generates additional data transfer and reduces performance when bandwidth-sensitive workloads are processed. When the synchronous write bias is optimized for latency, an additional copy is read from ARC and written to flash. When write-optimized log devices are mirrored, an additional two copies are written to flash. Write-optimized flash devices are designed to service a lot of IOPS, but can easily become bandwidth-saturated with high-throughput workloads. Even if there are many idle flash devices configured in the storage pool and there is an adequate amount of flash bandwidth available, SAS bandwidth becomes a limiting factor. In a configuration with mirrored storage pools with mirrored log devices and a latency synchronous write bias setting, synchronous writes are written first to ARC, and then four more copies are written. SAS bandwidth would be four times larger than network bandwidth. Throughput-optimized writes generate better performance for bandwidth-sensitive workloads.

READ CACHE

Read-optimized flash should not be used for caching traditional Oracle RMAN workloads because there is little benefit from storing Oracle RMAN backup sets in cache. Moreover, the level 2 ARC is not intended for streaming workloads. The cache device usage share setting should be configured to not use cache devices.

DATA COMPRESSION

The data compression setting determines whether compression algorithms are applied by the Oracle ZFS Storage Appliance system at the share level. Compression is most effective at the Oracle Database level, and the best practice is to use HCC for read-focused transactional workloads and Advanced Row Compression for write-focused transactional workloads.

For traditional Oracle RMAN workloads, LZ4 compression should be enabled at the share level. It provides additional benefit when combined with Oracle Database compression by reducing the bandwidth to back-end disk with only a minimal impact to CPU utilization. Physical network throughput is actually increased when using LZ4 because SAS bandwidth and HDD utilization are typically limiting factors for a traditional Oracle RMAN workload. With Advanced Row Compression enabled for an OLTP Oracle Database, LZ4 often provides additional space savings in the range of 1.8 to 2.4 times. Gzip-based compression algorithms are costly on CPU overhead and should not be considered in this context.

DEDUPLICATION

It is recommended to use data deduplication with backup strategies when four or more incremental level 0 or full backups will be active within the retention policy. Deduplication should use a double-parity storage profile and be combined with LZ4 compression to optimize useable capacity. A 1 MB record size should be used with deduplication.

Best practices require a pair of metadevices per storage pool for every 88 data disks. Typically, backup use cases utilize two storage pools, with one active on each controller. Each storage pool is allocated half of the data disks in each disk shelf.

Any configuration that uses deduplication has a minimum of four metadevices, with two in each storage pool. The metadevice pairs should not be placed in the same Oracle Storage Drive Enclosure DE3-24 disk shelf, although they can share one disk shelf if one metadevice pair is assigned to the other storage pool. More metadevices are not recommended until the configuration exceeds eight Oracle Storage Drive Enclosure DE3-24 disk shelves, which is 88 data disks per storage pool. At that time, an additional four metadevices would be added to the configuration, with, again, two in each storage pool.

For an Oracle Storage Drive Enclosure DE3-24 configuration with 12 disk shelves that use data deduplication, it is recommended to start with eight metadevices, with four in each storage pool. It is also recommended to grow the configuration four disk shelves at a time to preserve an NSPF double-parity profile, and to avoid performance impacts from vdev hotspots. When growing from 12 to 16 disk shelves, no additional metadevices are needed. Then, when growing from 16 to 20 disk shelves, two of the disk shelves should be ordered with all data disks (24), the other two disk shelves should be ordered with 20 data disks and 2 metadevices, and the remaining two slots should be filled with log devices. A pair of metadevices from separate disk shelves should be added to one storage pool, and the remaining two metadevices should be added to the other storage pool.

Metadevices should be configured in a striped format. The data stored there is a secondary copy, similar to L2ARC read cache. As such, redundancy is not an important factor.

If deduplication is enabled, an individual backup job should always utilize the same controller to maximize data reduction. The deduplication domain is global to the storage pool and will not spread to the other storage pool that is primary on the second controller in the cluster. A backup job that writes data to both controllers has a reduced opportunity for duplicate block matches. Some duplicate blocks are written to one storage pool, while the matching allocated block resides on the other storage pool.

When an Oracle ZFS Storage Appliance system is used to back up multiple Oracle Database instances, it is recommended to configure 50 percent of the Oracle Database instances to always use one controller, and 50 percent to always use the other controller. Furthermore, optimal resource utilization can be achieved by staggering the backups so that two are running at one time, rather than initiating all backups at the same time each day.

The following figure shows a backup share that was created using the settings discussed in this section.

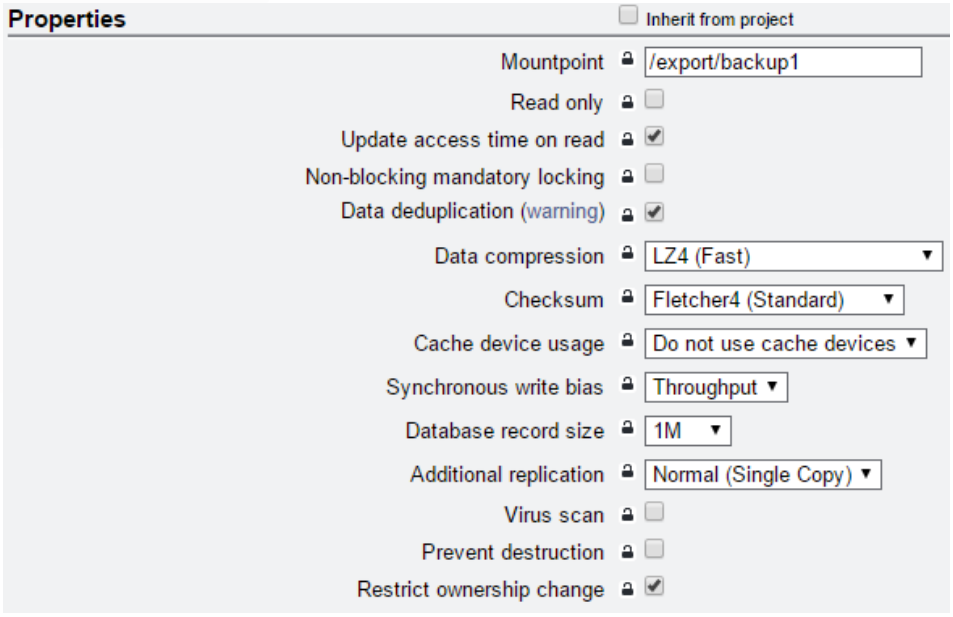

Figure 8. Example share for traditional Oracle RMAN backup

Oracle RMAN Configuration

BACKUP FORMAT

Level 0 backups can technically be either the backup-set or image-copy format; however, the backup-set format should be used. The backup-set format is required to achieve many of the benefits highlighted in the data protection strategy section of this document.

OPTIMIZING CHANNELS

Determining the number of Oracle RMAN channels to use is an important aspect of tuning a backup solution. When Oracle RMAN opens a new channel, it allocates a new set of input and output buffers. Each channel has the ability to take a datafile or a section of a datafile and process the backup or restore job in parallel to work being done by other channels. Channels can be assigned to different nodes in the Oracle RAC cluster, and can have different backup destinations, with shares potentially owned by different Oracle ZFS Storage Appliance controllers.

Additional channels increase scalability and can provide significantly improved performance, more-efficient resource utilization, load balancing across Oracle Database nodes, a more robust HA architecture, and workload spreading between storage controllers.

As hardware limits are approached, allocating additional Oracle RMAN channels provides diminishing returns. It is not recommended to over-allocate channels because there is little to no performance gain, despite additional memory and CPU resources allocated for more Oracle RMAN buffers, and added complexity in the form of more backup pieces being created.

Determining the recommended number of channels for a particular configuration depends on the hardware factor that will limit overall performance in an optimally configured solution. Performance limiting components could be many things, including Oracle Exadata or the network, HDD, CPU, or SAS resources. Thorough testing is always recommended when implementing major changes in a production environment. However, the following table provides guidance for how many Oracle RMAN channels to configure in a traditional Oracle RMAN backup strategy for each hardware configuration.

This table assumes recent Exadata storage server (X6 and later) and ZS7-2 HE with storage balanced across both controllers. It assumes that network and SAS bandwidth are not limiting factors, that the best practices in this document are implemented, and that there are no other significant concurrent workloads during the backup window.

TABLE 3. SUGGESTED ORACLE RMAN CHANNELS PER CONFIGURATION FOR A TRADITIONAL ORACLE RMAN BACKUP STRATEGY

	Channels per Oracle Exadata Eighth Rack	Channels per Oracle Exadata Quarter Rack	Channels per Oracle Exadata Half Rack	Channels per Oracle Exadata Full Rack
1 Disk Shelf	8	8	8	8
2 Disk Shelves	8	12	12	12
3–4 Disk Shelves	8	16	16	16
5–6 Disk Shelves	8	16	24	24
7–8 Disk Shelves	8	16	32	32
9+ Disk Shelves	8	16	32	40

When Oracle RMAN channels are configured or allocated, they should be alternated across the Oracle RAC nodes and storage shares.

SECTION SIZE

Enabling highly parallel Oracle RMAN workloads is critical for achieving optimal performance and resource utilization from the backup solution. One challenge is when a very large datafile is encountered. If it is processed by a single Oracle RMAN channel, throughput slows significantly, and other hardware resources in the environment sit idle while waiting for the outlier datafile processing to be completed.

Oracle RMAN's solution to this problem lies in its ability to break up large files into smaller pieces that can be processed in parallel by multiple channels. This is called multisection support and is determined by the section size parameter. It is recommended to set the section size to 100 gigabytes (`100G`).

FILESPERSET PARAMETER

The `filesperset` parameter determines how many datafiles or sections of datafiles are included in each backup set. When multiple input files are read to create a single backup set, it can improve performance, particularly when the read or copy phases are limiting factors. The default `filesperset` setting is 64; however, this is detrimental for single-file or partial Oracle Database restore operations because the entire backup set will be read back, even though only a small section is used. Also, an excessively large `filesperset` setting can impact the load balancing and performance scaling properties of Oracle RMAN. The objective is to have all Oracle RMAN channels effectively utilized throughout the backup. If there is a limited number of datafiles or data sections, it might not be possible to create full backup sets on every channel.

As a general practice, it is recommended to set the `filesperset` parameter to 1. Testing has shown that this provides excellent performance while load balancing across all channels. If deduplication is enabled, it is a requirement to set `filesperset` to 1. Including multiple files or sections in the same backup piece diminishes deduplication benefits.

SAMPLE RUN BLOCK

Here is a sample run block for a weekly level 0 backup that can be included as part of an incremental backup strategy. This example assumes a half-rack Exadata Storage Expansion X6-2 backing up to both controllers of an Oracle ZFS Storage Appliance system configured with four disk shelves. Oracle RMAN backup services are used to evenly spread channels across all four Oracle RAC nodes. Channels are alternated between the two storage shares, with one owned by each controller.

```
run

{

allocate channel ch1 device type disk connect 'sys/passwd@exa-scan/bkup_db1' format
'/zfs/bkup1/%U';

allocate channel ch2 device type disk connect 'sys/passwd@exa-scan/bkup_db2' format
'/zfs/bkup2/%U';

allocate channel ch3 device type disk connect 'sys/passwd@exa-scan/bkup_db3' format
'/zfs/bkup1/%U';

allocate channel ch4 device type disk connect 'sys/passwd@exa-scan/bkup_db4' format
'/zfs/bkup2/%U';

allocate channel ch5 device type disk connect 'sys/passwd@exa-scan/bkup_db1' format
'/zfs/bkup1/%U';

allocate channel ch6 device type disk connect 'sys/passwd@exa-scan/bkup_db2' format
'/zfs/bkup2/%U';

allocate channel ch7 device type disk connect 'sys/passwd@exa-scan/bkup_db3' format
'/zfs/bkup1/%U';
```

```
allocate channel ch8 device type disk connect 'sys/passwd@exa-scan/bkup_db4' format
'/zfs/bkup2/%U';

allocate channel ch9 device type disk connect 'sys/passwd@exa-scan/bkup_db1' format
'/zfs/bkup1/%U';

allocate channel ch10 device type disk connect 'sys/passwd@exa-scan/bkup_db2' format
'/zfs/bkup2/%U';

allocate channel ch11 device type disk connect 'sys/passwd@exa-scan/bkup_db3' format
'/zfs/bkup1/%U';

allocate channel ch12 device type disk connect 'sys/passwd@exa-scan/bkup_db4' format
'/zfs/bkup2/%U';

allocate channel ch13 device type disk connect 'sys/passwd@exa-scan/bkup_db1' format
'/zfs/bkup1/%U';

allocate channel ch14 device type disk connect 'sys/passwd@exa-scan/bkup_db2' format
'/zfs/bkup2/%U';

allocate channel ch15 device type disk connect 'sys/passwd@exa-scan/bkup_db3' format
'/zfs/bkup1/%U';

allocate channel ch16 device type disk connect 'sys/passwd@exa-scan/bkup_db4' format
'/zfs/bkup2/%U';

configure snapshot controlfile name to '+DATA/dbname/snapcf_dbname.f';

BACKUP AS BACKUPSET

  SECTION SIZE 100G

  INCREMENTAL LEVEL 0

  DATABASE

  FILESPERSET 1

  TAG 'BKUP_SUNDAY_L0';

}
```

BEST PRACTICES FOR AN INCREMENTALLY UPDATED BACKUP STRATEGY

This section details the recommended configuration steps necessary to achieve optimal performance, reliability, and usability when utilizing an Oracle RMAN incrementally updated backup strategy to protect Oracle Exadata with Oracle ZFS Storage Appliance systems. Best practices in this section specifically address an incrementally updated backup strategy. This section is intended to be used as supplementation for the technological details and general best practices that are fully described in previous sections.

Configuring an Oracle ZFS Storage Appliance System

It is recommended to use mirrored storage profiles and SSD write-flash accelerators for an incrementally updated backup strategy. When the previous incremental level 1 backup set is merged into the image copy, it creates a large amount of small-block, random I/O for Oracle Database instances running OLTP workloads. Mirrored storage pools are well suited for handling I/O-intensive workloads. Small I/O updates to the backup copy are latency-sensitive and require write-optimized flash in the storage pool for optimal performance.

CREATING SHARES

Two shares are recommended when backing up Oracle Database with an incrementally updated backup strategy. The initial image copy backup should be placed in one share and the daily incremental backup should be placed in the other.

Share access permissions should be aligned to match the user ID of the `oracle` user and the group ID of the `dba` group. A standard Oracle Exadata configuration is 1001 and 1002, respectively. In most configurations, the directory permissions are set to rwxr-x---.

RECORD SIZE

Reads and writes for the daily incremental share generate large, 1 MB I/O operations. Therefore, it is recommended to configure the ZFS share's record size to 1 MB.

Updates to the backup copy files generate smaller writes because they are updating only blocks within the backup datafile that have changed on the active datafile. Determining the optimal ZFS record size for the copy share depends on the characteristics of the transactional workload to the source Oracle Database instance.

If the source Oracle Database instance is running an OLTP workload, the ZFS record size should be set to 32 KB. An OLTP workload is characterized by mostly small transactions, with a focus on changing and reading existing rows of data. This generates relatively small write I/O operations that can be dispersed throughout the datafile. Setting the ZFS record size to 32 KB minimizes read-modify-write overhead while still providing good performance for streaming workloads, such as the initial backup and subsequent restore or restore-validate operations.

If the source Oracle Database instance is running an online application processing (OLAP) workload, the ZFS record size should be set to 128 KB. An OLAP workload is characterized by large queries and batch appends that generate large write I/O operations when updating a backup copy.

SYNCHRONOUS WRITE BIAS

Synchronous write bias is a share setting that controls behavior for servicing synchronous writes. It can be optimized for latency or throughput.

The synchronous write bias should be configured for throughput for the daily incremental share and configured for latency for the copy share. Applying changes to the backup copy is a latency-sensitive workload that requires copying I/O to write-optimized flash so that a faster acknowledgement is returned to the client. This significantly improves the IOPS capabilities of the share and the overall performance of the solution.

READ CACHE

Read-optimized flash is not required, but is recommended to optimize performance during the update phase in an incrementally updated backup strategy. The cache-device-usage share setting on the copy share should be configured for all data and metadata.

DATA COMPRESSION

LZ4 compression is recommended.

Configuring the Oracle RMAN Environment

BACKUP FORMAT

In an incrementally updated backup strategy, the level 0 backup has to be in the image-copy format, and daily incremental backups are always the backup-set format. There are no other choices.

OPTIMIZING CHANNELS

Channels allocated for an image-copy operation have a lower per-channel throughput capacity than channels allocated for a backup-set operation, because they use shared buffers for I/O functions. Oracle RMAN spends more time in a wait state because the shared buffer has not cleared its busy status. This does not impact the overall performance of the backup solution in environments that can properly scale to use many channels spread across multiple hardware platforms, but more channels might be required to achieve the same amount of throughput as a backup-set operation.

SECTION SIZE

Multisection support for image copies was introduced in Oracle Database 12*c*. When running a previous version of Oracle Database software, it is not recommended to use the image-copy format to back up big file tablespaces because Oracle RMAN performance scaling and load balancing cannot be ensured. This is not a concern with standard tablespaces because multiple datafiles guarantee optimal performance scaling and load balancing across all Oracle RMAN channels.

DEDUPLICATION

Deduplication should not be enabled with an incrementally updated backup use case. An incrementally updated backup strategy typically requires record sizes smaller than 128 KB, which is not recommended with the deduplication architecture. There is limited opportunity for duplicate matches because subsequent level 0 or full backups are not included in this use case.

SAMPLE RUN BLOCK

Here is an example run block that is designed to be run repeatedly every time a backup of Oracle Database is performed. It uses a two-share incrementally updated backup strategy and places level 0 image copies of each file in bkup1, and it places subsequent level 1 incremental backups in bkup2. The same run block is executed every night, but it performs different tasks, depending on the existence of prior level 0 and level 1 backups.

After initial execution, it creates a level 0 image copy backup of all the datafiles in Oracle Database. On the second night, it creates a backup set of only the data that was changed within the last 24 hours and stores this in a separate share. On the third night, it creates a backup set of only the data that was changed within the last 24 hours and then applies the previous night's changed data into the backup copy. Every subsequent night that the backup is run, it performs the same function of backing up just the data changed within the last 24 hours and applying previously changed data to the backup copy. This is tracked with the Oracle RMAN command `tag`.

This example assumes an Oracle Exadata half-rack model with four Oracle Database servers. It also assumes a standard file tablespace with no need to break large datafiles into small sections for backup. Oracle RMAN backup services are used in the connect string to spread the channels across all available nodes in the Oracle RAC cluster.

```
RUN

{

allocate channel ch1 device type disk connect 'sys/passwd@exa-scan/bkup_db1' format
'/zfs/bkup1/%U';

allocate channel ch2 device type disk connect 'sys/passwd@exa-scan/bkup_db2' format
'/zfs/bkup1/%U';

allocate channel ch3 device type disk connect 'sys/passwd@exa-scan/bkup_db3' format
'/zfs/bkup1/%U';

allocate channel ch4 device type disk connect 'sys/passwd@exa-scan/bkup_db4' format
'/zfs/bkup1/%U';

allocate channel ch5 device type disk connect 'sys/passwd@exa-scan/bkup_db1' format
'/zfs/bkup1/%U';
```

```
allocate channel ch6 device type disk connect 'sys/passwd@exa-scan/bkup_db2' format
'/zfs/bkup1/%U';

allocate channel ch7 device type disk connect 'sys/passwd@exa-scan/bkup_db3' format
'/zfs/bkup1/%U';

allocate channel ch8 device type disk connect 'sys/passwd@exa-scan/bkup_db4' format
'/zfs/bkup1/%U';

allocate channel ch9 device type disk connect 'sys/passwd@exa-scan/bkup_db1' format
'/zfs/bkup1/%U';

allocate channel ch10 device type disk connect 'sys/passwd@exa-scan/bkup_db2' format
'/zfs/bkup1/%U';

allocate channel ch11 device type disk connect 'sys/passwd@exa-scan/bkup_db3' format
'/zfs/bkup1/%U';

allocate channel ch12 device type disk connect 'sys/passwd@exa-scan/bkup_db4' format
'/zfs/bkup1/%U';

allocate channel ch13 device type disk connect 'sys/passwd@exa-scan/bkup_db1' format
'/zfs/bkup1/%U';

allocate channel ch14 device type disk connect 'sys/passwd@exa-scan/bkup_db2' format
'/zfs/bkup1/%U';

allocate channel ch15 device type disk connect 'sys/passwd@exa-scan/bkup_db3' format
'/zfs/bkup1/%U';

allocate channel ch16 device type disk connect 'sys/passwd@exa-scan/bkup_db4' format
'/zfs/bkup1/%U';

configure device type disk parallelism 16;

RECOVER COPY OF DATABASE

  WITH TAG 'incr_zfs';

BACKUP

   INCREMENTAL LEVEL 1

  FOR RECOVER OF COPY WITH TAG 'incr_zfs'

  DATABASE format '/zfs/bkup2/%U';

}
```

ORACLE RMAN BACKUP AND RESTORE THROUGHPUT PERFORMANCE SIZING FOR ORACLE ZS7

The following figure shows the maximum sustainable throughput rates attained by Oracle's AIE group during Oracle RMAN backup and restore operations to Oracle ZS7-2. These rates were collected running level 0 backup sets. The backup workload was configured to use both controllers. A double-parity storage profile was used with LZ4 compression, and deduplication was not enabled.

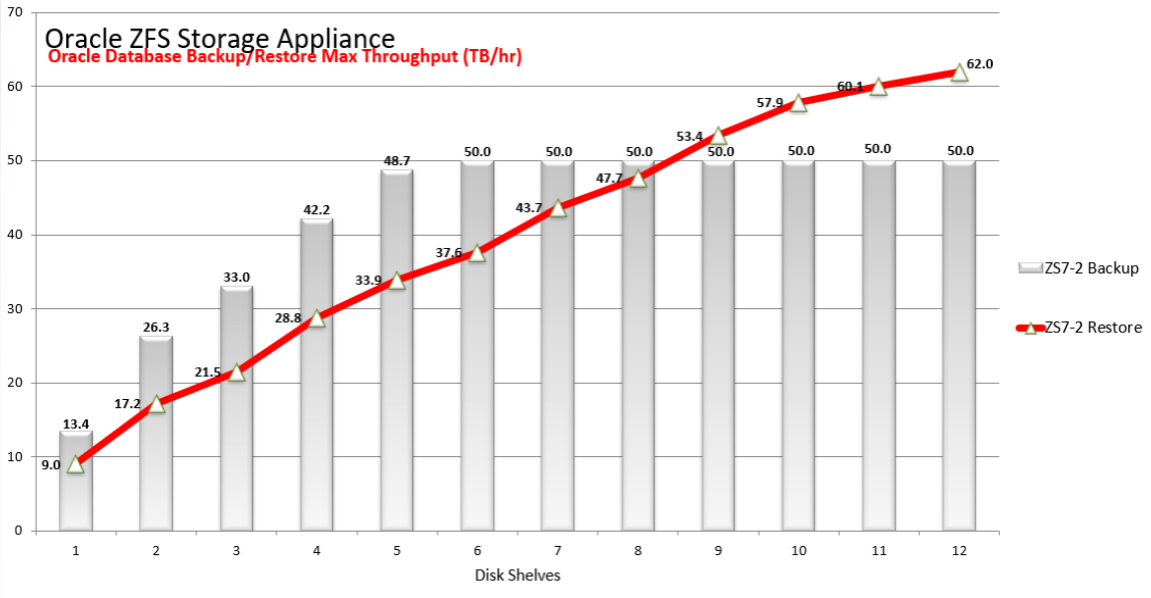

Figure 9. Oracle RMAN throughput for Oracle ZS7-2 High End System

These are real-world results using Oracle Database 12*c* and a large OLTP Oracle Database instance populated with sample customer data in a sales order-entry schema. The Oracle Database instance used in these performance characterizations contained 15 TB of user data. Advanced Row Compression was enabled at the Oracle Database level because this is the best practices recommendation when running OLTP workloads. This was a fully functional Oracle Database instance with the ability to run live transactional workloads before, during, or after an Oracle RMAN backup operation. These throughput rates were not obtained using a database or I/O-generator test tool because those can be misleading and only indirectly applicable to real-world use cases. Also, the throughput rates were not projected based on low-level system benchmarks.

In all scenarios, sustainable throughput was determined by measuring physical I/O at the network layer. An average was collected over an extended period. The figure demonstrates that the maximum Oracle Database backup and restore rates for Oracle ZS7-2 HE have been proven capable of sustaining a real-world Oracle Exadata backup environment. Maximum sustainable backup rates of 50 TB per hour and 62 TB per hour were demonstrated. Performance was collected using Oracle Exadata and Oracle ZS7-2 HE system that was otherwise idle during the Oracle RMAN operation. Test environments were configured following the best practices presented in this white paper.

The focus of this evaluation was Oracle ZFS Storage Appliance systems. A reasonable effort was made to ensure that they were always the limiting factor. It would not be meaningful for effective characterization of the storage if the runs were limited by external factors, such as network infrastructure, database contention, source-side hardware resources, or Oracle RMAN configuration shortcomings.

Each configuration under test included a sufficient number of networking and SAS cards to provide the bandwidth that the configuration was capable of supporting. A configuration with fewer networking or SAS cards could encounter a storage-side bottleneck at a lower level.

In the restore configurations, the primary bottleneck was storage-pool (data vdevs) saturation from large sequential reads and prefetch I/O.

For the backup characterization, the primary bottleneck was also storage-pool (data vdevs) saturation, but not in smaller configurations because performance was still ramping up as additional disk shelves were added. However, after the backup rate plateaus, the primary bottleneck was controller CPU resources.

Compression was the largest consumer of CPU cycles in these runs. However, compression reduces the amount of data written to the storage pool. Even with it enabled, storage-pool saturation was the primary bottleneck for small and medium

configurations. With no compression, backup throughput scaled linearly, which avoided a CPU bottleneck, but at a shallower angle because there was more data being written to the storage pool. Uncompressed backups to Oracle ZS7-2 can reach 50 TB per hour for 11 to 12 disk shelves.

Despite consuming more CPU resources, the use of LZ4 compression is an easy win for Oracle RMAN backup use cases. Not only does it enable higher throughput rates in small and medium configurations, but the 1.8 to 2.8 times storage-footprint savings is a critical benefit in even large configurations.

Oracle Storage Drive Enclosure DE3-24P and Oracle Storage Drive Enclosure DE3-24C disk shelves can both support these throughput rates. The two drive types have a similar throughput rating. The 10 K drives have an advantage in IOPS, which is not a factor for sequential 1 MB workloads, such as Oracle RMAN level 0 backup and restore workloads. All-flash disk shelves were not included in this performance sizing effort because they generally do not provide optimal ROI for a backup use case.

Performance Sizing with Deduplication

Disk storage space consumed by traditional RMAN full backups can be reduced by combining deduplication with LZ4 compression. Keep in mind that enabling deduplication consumes more CPU resources.

Level 0 backup workloads can be configured to use a single controller with half of the data disks in each Oracle Storage Drive Enclosure DE3-24 disk shelf. Backups can be configured to use a single storage pool to maximize deduplication benefits, per the best practices presented in this document.

When running an active workload on just one storage pool, only half of the data disks from each Oracle Storage Drive Enclosure DE3-24 disk shelf are used. This assumes that separate workloads are using the remaining disks in a storage pool that is active on the other controller. A double-parity storage profile was used with LZ4 compression.

Our testing is based on customer-emulated OLTP workload was run to generate 10 to 15 percent changed data between each incremental level 0 backup. The Oracle Database schema used in this characterization contained 15 TB of user data.

The deduplication architecture consumes more CPU resources compared to the baseline architecture. This is observable in larger configurations. Oracle ZFS Storage Appliance systems were able to sustain very high throughput rates, even with deduplication enabled. This is a major advantage over competitor products when total data reduction ratios fall between 3 to 6 times.

With deduplication is enabled in our testing, multiple Oracle Database workloads are running; one Oracle Database instance ran an Oracle RMAN workload to one controller, and another Oracle Database instance ran a concurrent Oracle RMAN workload to the other controller.

For more information about configuring Oracle ZFS Storage data reduction features with Oracle RMAN full backups, see Oracle Support Document 2087231.1 (Guidelines When Using ZFS Storage in an Exadata Environment.

CONCLUSION

Finding the right backup solution for Oracle Exadata is a challenging problem. Costly alternatives provide poor ROI and cannot support high-performance environments. Competitive offerings are inflexible and do not address all of the customer's needs.

Oracle ZFS Storage Appliance systems have proven to be an ideal solution for protecting the mission-critical data that resides on Oracle Exadata. Powerful features combined with custom Oracle-on-Oracle integrations enable a wide range of Oracle RMAN backup strategies. These provide outstanding performance and flexibility unmatched by third-party solutions.

Extreme restore throughput helps satisfy even the most stringent RTOs. Archive log multiplexing delivers recovery times of 20 minutes or less. Oracle Intelligent Storage Protocol, HCC, LZ4 storage compression, large 1 MB record sizes, data deduplication, and Direct NFS Client provide unique advantages when protecting Oracle Database.

In addition to data protection benefits, an Oracle Exadata backup solution using Oracle ZFS Storage Appliance systems provides many other advantages, such as low-cost, high-performance storage for unstructured data that resides outside of Oracle Database, and an ideal snapshot-cloning solution for provisioning development and test environments. It is easy to see why Oracle ZFS Storage Appliance systems are a preferred solution for protecting Oracle Exadata.

REFERENCES

- "Networking Best Practices with Oracle ZFS Storage Appliance"
- "Realizing the Superior Value of Oracle ZFS Storage Appliance"
- "Configuring Oracle ZFS Storage Appliance for Oracle RAC Database RMAN Backup and Restore"
- Oracle Database 12*c* Hybrid Columnar Compression web page
- Oracle Database 12*c* Advanced Compression web page
- Oracle Intelligent Storage Protocol data sheet
- Oracle ZFS Storage Appliance data sheet
- Oracle Enterprise Manager web page

ORACLE CORPORATION

Worldwide Headquarters
500 Oracle Parkway, Redwood Shores, CA 94065 USA

Worldwide Inquiries
TELE + 1.650.506.7000 + 1.800.ORACLE1
FAX + 1.650.506.7200
oracle.com

CONNECT WITH US

Call +1.800.ORACLE1 or visit oracle.com. Outside North America, find your local office at oracle.com/contact.

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

Integrated Cloud Applications & Platform Services

White Paper **Protecting Oracle Exadata X8 with ZFS Storage Appliance**
December 2019
Author: Greg Drobish
Contributing Authors: Oracle ZFS Storage Team, Oracle Exadata Team, Oracle MAA Team

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

ORACLE