ORACLE

EMC² where information lives®

White Paper

EMC Symmetrix VMAX Using EMC SRDF/TimeFinder and Oracle Database 10g/11g

Applied Technology

Abstract

This white paper introduces EMC® Symmetrix VMAX™ software and hardware capabilities, and provides a comprehensive set of best practices and procedures for high availability and business continuity when deploying Oracle Database 10g and Oracle Database 11g with Symmetrix® VMAX. This includes EMC TimeFinder® and Symmetrix Remote Data Facility (SRDF®), which have been widely deployed with Oracle databases.

January 2011

Part Number h6210.4

Table of Contents

Executive summary

The EMC® Symmetrix VMAX™ Series with Enginuity™ is a new offering in the Symmetrix® family. Built on the strategy of simple, intelligent, modular storage, it incorporates a new Virtual Matrix interface that connects and shares resources across all nodes, allowing the storage array to seamlessly grow from an entry-level configuration into the world's largest storage system. Symmetrix VMAX provides improved performance and scalability for demanding enterprise database environments while maintaining support for EMC's broad portfolio of software offerings. With the release of Enginuity 5874, Symmetrix VMAX systems now deliver new software capabilities that improve ease of use, business continuity, Information Life Management (ILM), virtualization of small to large environments, and security.

Symmetrix VMAX arrays are well integrated with Oracle databases and applications to support their performance needs, scalability, availability, ease of management, and future growth. This white paper introduces Symmetrix VMAX software and hardware capabilities, and provides a comprehensive set of best practices and procedures for high availability and business continuity when deploying Oracle Database 10g and Oracle Database 11g with EMC Symmetrix VMAX. This includes EMC TimeFinder® and Symmetrix Remote Data Facility (SRDF®), which have been widely deployed with Oracle databases.

Introduction

New Symmetrix VMAX ease of use, scalability and virtualization features

In addition to Symmetrix VMAX enhanced performance, scalability, and availability, Enginuity 5874 introduces new ease of use, virtualization, and ILM functionalities. With Symmetrix VMAX *Auto-provisioning Groups*, mapping devices to small or large Oracle database environments becomes faster and easier. Devices, HBA WWNs, or storage ports can be easily added or removed, and automatically these changes are propagated through the Auto-provisioning Group, thus improving and simplifying complex storage provisioning for any physical or virtual environment. With Symmetrix VMAX *Enhanced Virtual LUN Technology*, Oracle applications data can be migrated between storage tiers seamlessly, while the database is active, thus allowing the placement of data on the storage tier that best matches its performance and cost requirements. As database performance requirements change, it is easy and efficient to move the appropriate LUNs to their new storage tier. Symmetrix VMAX Virtual LUN migration doesn't consume host or SAN resources; it improves return on investment (ROI) by using the correct storage tiering strategy, and it reduces complexity as there is no need to change backup or DR plans since the host devices don't change. Additional enhancements to availability, scalability, and ease of use are introduced later in the paper and are fully described in the VMAX product guide.

Oracle mission-critical applications require protection strategy

The demand for database protection and availability increases as data grows in size and becomes more interconnected, and the organization infrastructure expands. It is essential to have continuous access to the database and applications and efficient use of available system resources. Data centers face disasters caused by human errors, hardware and software failures, and natural disasters. When disaster strikes, the organization is measured by its ability to resume operations quickly, seamlessly, and with the minimum amount of data loss. Having a valid backup and restartable image of the entire information infrastructure greatly helps achieve the desired level of recovery point objective (RPO), recovery time objective (RTO), and service level agreement (SLA).

Enterprise protection and compliance using SRDF

Data consistency refers to the accuracy and integrity of the data and the copies of the data. Symmetrix VMAX offers several solutions for local and remote replication of Oracle databases and applications data. With SRDF software, single or multiple database mirrors can be created, together with their external data, application files and/or message queues – all sharing a consistency group. Replicating data this way creates the point of consistency across business units and applications before any disaster takes place. Failover to the DR site is merely a series of application restart operations that reduce overall complexity and downtime. SRDF provides two- or three-site solutions, and synchronous and asynchronous replication, as well as a no data loss solution over any distance using SRDF/Star, cascaded or concurrent SRDF, and the new SRDF/Extended Distance Protection (EDP). With SRDF/Star, for example, compliance requirements

such as not operating the business without a disaster recovery site can be met, even when the production array is unavailable.

Oracle database clones and snapshots with TimeFinder

Every mission-critical system has a need for multiple copies, such as for development, test, backup offload, reporting, data publishing, and more. With Symmetrix VMAX using TimeFinder software, multiple Oracle database copies can be created or restored in a matter of seconds (either full volume clones or virtual snapshots), regardless of the database size. Such operations are incremental and only changes are copied over. As soon as TimeFinder creates (or restores) a replica, the target devices (or source) will immediately show the final image as if the copy has already finished, even if data copy operations continue incrementally in the background. This functionality shortens business operation times tremendously. For example, rather than performing backup directly on production, it can be offloaded in seconds to a standalone replica. In another example, if an Oracle database restore is required, as soon as TimeFinder restore starts, database recovery operations can start, and there is no need to wait for the storage restore to complete. This ability, also referred to as parallel restore, provides a huge reduction in RTO and increases business availability.

Oracle database recovery using storage consistent replications

In some cases there is a need for extremely fast database recovery, even without failing over to a DR site (especially when only one database out of many sustains a logical or physical corruption). By implementing TimeFinder consistency technology, periodic database replicas can be taken (for example, every few hours) without placing the Oracle database in hot backup mode. Oracle now supports database recovery on a consistent storage replica, applying archive and redo logs to recover it (Oracle support is based on Metalink note 604683.1).

Best practices for local and remote Oracle database replications

This white paper provides an overview of the Symmetrix VMAX system, Auto-provisioning Groups, and Virtual LUN technology with Oracle-related samples. It also details the procedures and best practices for the following use cases:

- Use Case 1 — Offloading database backups from production to a local TimeFinder/Clone, then using Oracle Recovery Manager (RMAN) for farther backup
- Use Case 2 — Facilitating parallel production database recovery by restoring a local TimeFinder/Clone backup image and applying logs to it
- Use Case 3 — Creating local restartable clones (or snaps) of production for database repurposing (such as creating test, development, and reporting copies)
- Use Case 4 — Creating remote mirrors of the production database for disaster protection (synchronous and asynchronous)
- Use Case 5 — Creating remote restartable and writeable database clones (or snaps) for repurposing
- Use Case 6 — Creating remote database valid backup and recovery clones (or snaps)
- Use Case 7 — Facilitating parallel production database recovery by restoring remote TimeFinder/Clone backup images simultaneously with SRDF restore, and then applying Oracle logs to the production database in parallel
- Use Case 8 — Demonstrating fast database recovery using a restartable TimeFinder replica

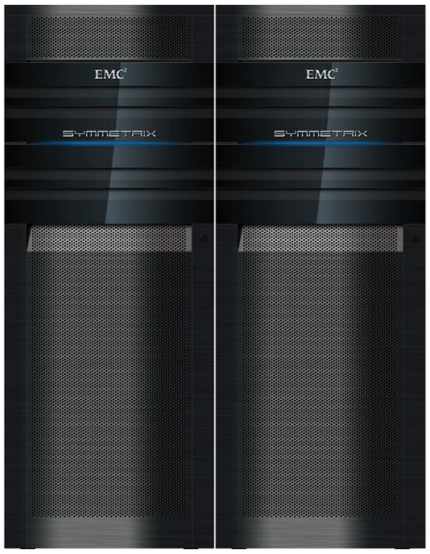
Audience

The primary audience of this white paper is database and system administrators, storage administrators, and system architects who are responsible for implementing, maintaining, and protecting robust databases and storage systems. It is assumed that the readers have some familiarity with Oracle database backup aspects and EMC software, and are interested in achieving higher database availability and protection.

Products and features overview

Symmetrix VMAX

Symmetrix VMAX is built on the strategy of simple, intelligent, modular storage, and incorporates a new Virtual Matrix interface that connects and shares resources across all nodes, allowing the storage array to seamlessly grow from an entry-level configuration into the world's largest storage system. It provides the highest levels of performance and availability featuring new hardware capabilities as seen in Figure 1.



- 2 – 16 director boards
- Up to 2.1 PB usable capacity
- Up to 128 FC FE ports
- Up to 64 FICON FE ports
- Up to 64 Gig-E / iSCSI FE ports
- Up to 1 TB global memory (512 GB usable)
- 48 – 2,400 disk drives
- Enterprise Flash drives 200/400 GB
- FC drives 146/300/450 GB 15k rpm (or 400 GB 10k rpm)
- SATA II drives 1 TB 7.2k rpm

Figure 1. The Symmetrix VMAX platform

Symmetrix VMAX provides the ultimate scale-out platform. It includes the ability to incrementally scale front-end and back-end performance by adding processing modules (nodes) and storage bays. Each processing module provides additional front-end, memory, and back-end connectivity.

Symmetrix VMAX also increases the maximum hyper size to 240 GB (64 GB on Symmetrix DMX™). This allows ease of storage planning and device allocation, especially when using Virtual Provisioning™ where the thin storage pool is already striped and large hypers can be easily used.

Symmetrix VMAX Auto-provisioning Groups

The Auto-provisioning Groups feature facilitates ease and simplicity of storage provisioning for standalone and clustered Oracle databases. It simplifies and shortens storage provisioning tasks for small- and large-scale environments. The storage provisioning that used to take many steps in prior releases can now be accomplished with just a few simple and intuitive operations.

The Auto-provisioning Groups feature is built on the notion of "storage groups," "initiator groups," "port groups," and the views that combine the groups together. Storage groups are populated with Symmetrix devices. Port groups are populated with the array front-end adapter (FA) port numbers. Initiator groups are populated with HBA WWN information. Then by simply combining storage, initiator, and port groups into views, the device masking operations take place automatically across the view. Any modification necessary to available storage devices, storage array ports, or HBAs would simply require changing the appropriate group and will automatically be incorporated throughout the view. For example, if additional database devices are necessary, simply adding those devices to the appropriate storage group will automatically initiate all the necessary mapping and masking operations across the entire view (note that if the devices are already mapped, the operation will complete faster, otherwise the Symmetrix config change will first map the devices appropriately before they are masked, making the task take a little longer). Initiator groups can be cascaded as shown in the next example.

Figure 2 shows an example of using Auto-provisioning Groups to mask Oracle Real Application Cluster (RAC) database devices. A storage group is created with the database devices and a port group with the Symmetrix ports. An initiator group is created for each host's HBAs (for long-term ease of management); however, they are then *cascaded* into a single initiator group for the entire cluster. The Auto-provisioning Groups view simply includes the storage group, port group, and the cascaded initiator group. If any hosts are added or removed from the cluster they will simply be added or removed from the cascaded initiator group. In a similar way, devices or Symmetrix ports can be added or removed from their groups and the view will automate the device provisioning for the cluster.

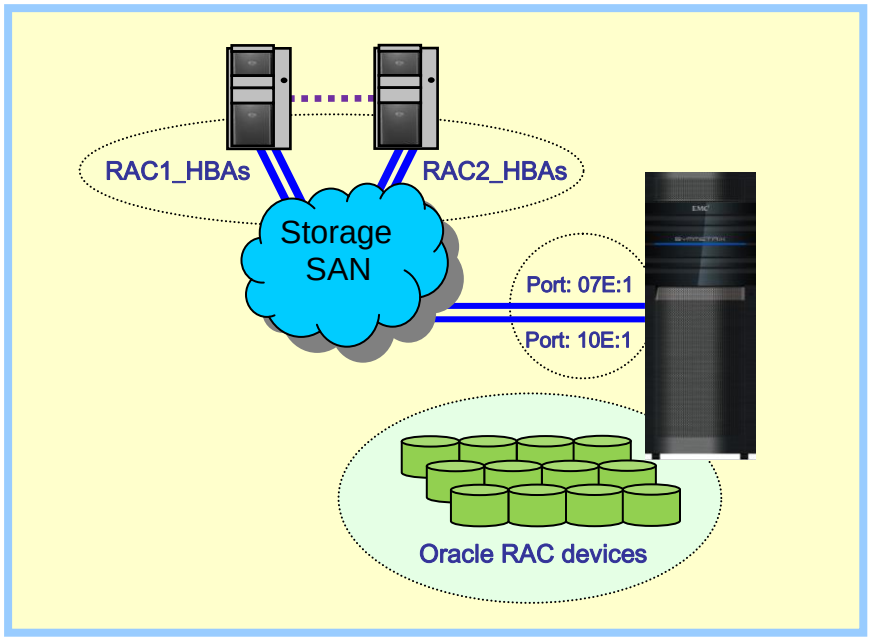


Figure 2. Oracle RAC and Auto-provisioning Groups

The following steps demonstrate the use of Auto-provisioning Groups, based on the example in Figure 2.

1. **Create** a **storage group for RAC devices**
   symaccess -name RAC_devs -type storage devs 790:7AF create
2. **Create a port group with storage ports** 7E:1 and 10E:1
   symaccess -name RAC_ports -type port –dirport 7E:1,10E:1 create
3. **Create an initiator group for each cluster node's HBAs**
   symaccess -name RAC1_hbas -type initiator -file ./RAC1_hbas.txt create
   The file RAC1_hbas.txt contains: WWN:10000000c975c2e4
   WWN:10000000c975c336
   symaccess -name RAC2_hbas -type initiator -file ./RAC2_hbas.txt create
   The file RAC2_hbas.txt contains: WWN:10000000c975c31a
   WWN:10000000c975c3ab
4. **Cascade the cluster nodes' initiator groups into a single one for the entire cluster**
   symaccess -name RAC_hbas -type initiator create
   symaccess -name RAC_hbas -type initiator add –ig RAC1_hbas
   symaccess -name RAC_hbas -type initiator add –ig RAC2_hbas
5. **Create the view for the entire RAC cluster storage provisioning**
   symaccess create view -name RAC_view
   -storgrp RAC_devs -portgrp RAC_ports -initgrp RAC_hbas

Symmetrix VMAX Enhanced Virtual LUN migration technology

Enginuity 5874 provides an enhanced version of Symmetrix Virtual LUN software to enable transparent, nondisruptive data mobility of devices between storage tiers and/or RAID protections. Virtual LUN migration technology provides users with the ability to move Symmetrix logical devices between disk types, such as high-performance enterprise Flash drives (EFDs), Fibre Channel drives, or high-capacity low-cost SATA drives. As devices are migrated they can change their RAID protection.

Virtual LUN migration occurs independent of host operating systems or applications, and during the migration the devices remain fully accessible to database transactions. While the back-end device characteristics change (RAID protection and/or physical disk type) the migrated devices' identities remain the same, allowing seamless online migration. Virtual LUN is fully integrated with Symmetrix replication technology and the source devices can participate in replications such as SRDF, TimeFinder/Clone, TimeFinder/Snap, or Open Replicator.

The advantages of migrating data using storage technology are ease of use, efficiency, and simplicity. Data is migrated in the Symmetrix back end without needing any SAN or host resources increasing migration efficiency. The migration is a safe operation as the target is treated internally as just another "mirror" of the logical device, although with its own RAID protection and storage tier. At the end of the migration the original "mirror" of the logical device is simply removed. Finally, since the identity of source devices doesn't change, moving between storage tiers is made easy and doesn't require additional change control of business operations such as remote/local replications and backup. The migration pace can be controlled using Symmetrix Quality of Service (symqos) commands.

Virtual LUN migration helps customers to implement an Information Life Management (ILM) strategy for their databases, such as the move of the entire database, tablespaces, partitions, or ASM diskgroups between storage tiers. It also allows adjustments in service levels and performance requirements to application data. For example often application storage is provisioned before clear performance requirements are known. At a later time once the requirements are better understood it is easy to make any adjustment to increase user experience and ROI using the correct storage tier.

Figure 3 shows an example of performing a Virtual LUN migration of an ASM diskgroup "+Sales" with 20 x 50 GB devices (ASM members). The migration source devices are spread across 40 x 300 GB hard disk drives and protected with RAID 1. The migration target devices are spread across only 4 x 400 GB EFDs and protected with RAID 5.

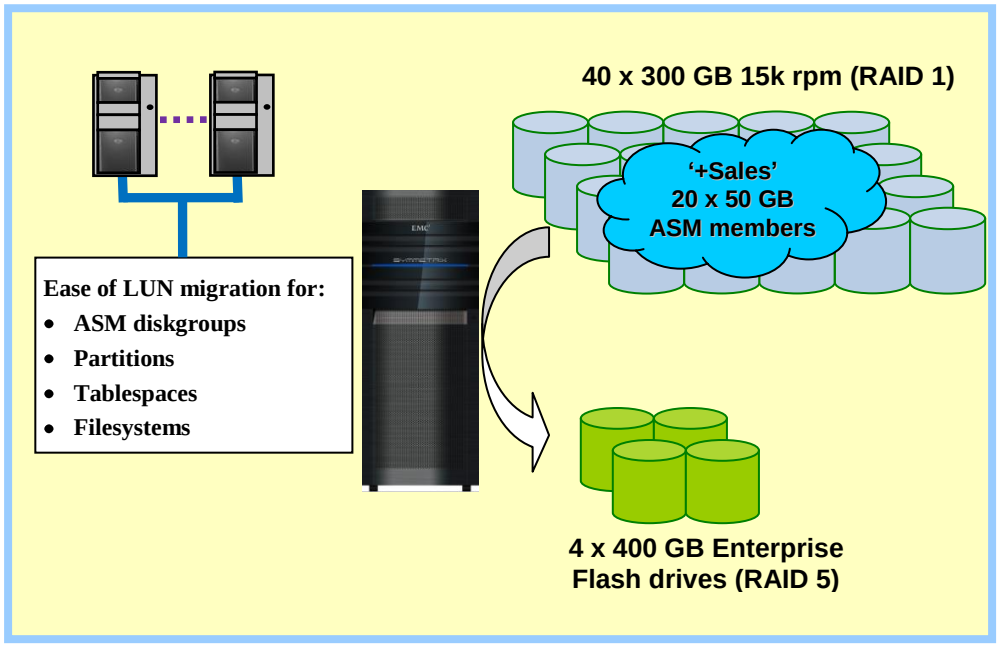


Figure 3. Migration example using Virtual LUN technology

The following steps demonstrate the use of Virtual LUN, based on the example in Figure 3.

1. **Optional: Verify information for a migration session called *Sales_mig***
   symmigrate -name Sales_mig –file Sales_ASM.txt validate
   The file Sales_ASM.txt contains the list of source and target migration devices:
   ```
   0100 0C00
   ...  ...
   0113 0C13
   ```
2. **Perform the migration**
   symmigrate -name Sales_mig –file Sales_ASM.txt establish
3. **Follow the migration progress and rate at 60-second intervals**
   symmigrate -name Sales_mig –file Sales_ASM.txt query –i 60
4. **Terminate the migration session after completion**
   symmigrate -name Sales_mig –file Sales_ASM.txt terminate
5. **Optional: Control migration pace**
   Create a Symmetrix DG with the source devices
   symdg create Sales_dg
   symld –g Sales_dg –range 0100:0113 addall
   Control the copy pace using the DG
   symqos –g Sales_dg set MIR pace 8

Virtual LUN can utilize configured or unconfigured disk space for the target devices. Migration to unconfigured disk space means that devices will move to occupy available free space in a target storage diskgroup. After the migration, the original storage space of the source devices will be unconfigured. In either case the source devices' identity doesn't change, making the migration seamless to the host; no changes to DR, backup, or high availability configuration aspects are necessary. When specifying configured disk space for the migration, in essence the source and target devices simply swap their storage characteristics. However, after the data was migrated to the target devices, the original source drive storage space will be reformatted, to prevent exposure of the data that once belonged to it.

With Enginuity 5874, migration of logical devices and metavolumes is supported. (Only the metahead volumes needs to be specified. The metamembers will be automatically selected.) Virtual LUN migration does not support migration of thin devices (or thin pool devices), virtual devices (or save pool devices), and internal Symmetrix devices such as VCM, SFS, or Vault.

Migration to configured space

This option is useful when most of the space in the target diskgroup is already configured (and therefore not enough free space is available). It is also useful when it is expected that the migration is temporary and a reverse migration will take place at a later time to the same target devices. One example of this is migrating the SALES ASM diskgroup to a Flash drive tier before the end-of-the-month closing report. That way when the time comes to migrate back, the source devices return to occupy their previous storage space. When migrating to a configured space both source and target devices are specified. The target devices should match in size to the source devices and they should be at least unmasked to any host, and optionally unmapped from any Symmetrix FA port. These requirements ensure that the target devices of the migration do not contain currently active customer data. Likewise, the target devices cannot be involved in any other Symmetrix copy operation such as SRDF, Clone, Snap, or Open Replicator. After the migration, the target devices occupy the original storage location and protection of the source devices, and the original source device storage space is formatted to prevent exposure of its old data by the target.

Migration to unconfigured space

This option is useful when enough free space is available in the target storage diskgroup. When migrating to an unconfigured space only the source devices are specified. For the migration target, a storage

diskgroup number is provided along with the RAID protection type of the new LUN. At the completion of this migration the old source LUN is unconfigured so no reformat of the LUN is required.

Symmetrix VMAX TimeFinder product family

The EMC TimeFinder family of local replication technology allows for creating multiple, nondisruptive, read/writeable storage-based replicas of database and application data. It satisfies a broad range of customers' data replication needs with speed, scalability, efficient storage utilization, and minimal to no impact on the applications – regardless of the database size. TimeFinder provides a solution for backup, restart, and recovery of production databases and applications, even when they span Symmetrix arrays. TimeFinder is well integrated with other EMC products such as SRDF and allows the creation of replicas on a remote target without interrupting the synchronous or asynchronous replication. If a restore from a remote replica is needed, TimeFinder and SRDF will restore data incrementally and in parallel, to achieve a maximum level of availability and protection. The TimeFinder product family supports the creation of dependent write-consistent replicas using EMC consistency technology, and replicas that are valid for Oracle backup/recovery operations, as described later in the use cases.

TimeFinder/Clone and the new cascaded clones

TimeFinder/Clone provides the ability to create, refresh, or restore multiple full volume copies of the source volumes where after the first full synchronization, only incremental changes are passed between source and target devices. TimeFinder/Clone operations can have any combination of standard (STD) and/or business continuance volumes (BCV) for source and/or target devices, making it extremely flexible. TimeFinder/Clone can work in emulation mode, simulating TimeFinder/Mirror commands (symmir) for legacy reasons; however, it is recommended to use the native TimeFinder/Clone command syntax (symmclone) when creating new scripts.

TimeFinder/Clone can scale to thousands of devices and can create up to 16 targets to each source device. It also provides the flexibility of synchronizing the target volumes before the clone session (replica) is activated, also referred to as precopy, after the clone session is activated, also referred to as background copy, or let the clone devices synchronize only when data is accessed, also referred to as no-copy, which can be used, for example, for short-term gold copies.

TimeFinder always presents the final copied image immediately on its target devices (when creating a replica) or source devices (when restoring it), even if background copy operations are still in progress. This allows the application to immediately use the TimeFinder devices. For example, during TimeFinder restore of a valid database backup image, Oracle roll forward recovery can start in parallel, reducing RTO.

Cascaded clones is a new feature in Enginuity 5874 that provides the ability to perform one additional clone operation on a clone target without losing the incremental nature of the relationships. This can become useful when the first clone is a gold copy (backup image, for example) that should not be used, but additional replicas are required off it for purposes such as backup, reporting, publishing, test/dev, and so on. Another option is to do it by using multiple TimeFinder/Snaps. However when a full volume replica is required instead, starting with Enginuity 5874 it is also possible to create an additional clone and deploy it for such purposes.

TimeFinder/Snap and the new TimeFinder/Snap Recreate

TimeFinder/Snap software allows users to create, refresh, or restore multiple read/writeable, space-saving copies of data. TimeFinder/Snap allows data to be copied from each source device to as many as 128 target devices where the source devices can be either a STD device or a BCV. The target devices are Symmetrix *virtual devices* (VDEV) that consume negligible physical storage through the use of pointers to track changed data.

Any update to source target devices after the snap session was activated causes the pre-updated data to be copied in the background to a designated shared storage pool called a *save device* pool. The virtual device's pointer is then updated to that location. Any subsequent updates after the first data modification won't require any further background copy. Since copy operations happen in the background, performance

overhead of using TimeFinder/Snap is minimal, and the process is known as Avoid Copy on First Write (ACOFW).

TimeFinder/Snap Recreate is new in Enginuity 5874. It provides the ability to very quickly refresh TimeFinder snapshots. Previously it was necessary to terminate an older snap session in order to create a new one. The TimeFinder *recreate* command simplifies the process to refresh old snaps without having to describe the source and target devices relationships again.

TimeFinder Consistent Split

With TimeFinder you can use the Enginuity Consistency Assist (ECA) feature to perform consistent splits between source and target device pairs across multiple, heterogeneous hosts. Consistent split (which is an implementation of instant split) helps to avoid inconsistencies and restart problems that can occur if you split database-related devices without first quiescing the database. The difference between a normal instant split and a consistent split is that when using consistent split on a group of devices, the database writes are held at the storage level momentarily while the foreground split occurs, maintaining dependent-write order consistency on the target devices comprising the group. Since the foreground instant split completes in just a few seconds, Oracle needs to be in hot backup mode only for this short time when hot backup is used. When consistent split alone is used to create a restartable replica, interference with business operations is minimal.

TimeFinder target devices, after performing a consistent split, are in a state that is equivalent to the state a database would be in after a power failure, or if all database instances were aborted simultaneously. This is a state that is well known to Oracle and it can recover easily from it by performing a crash recovery the next time the database instance is started.

TimeFinder and SRDF

TimeFinder and SRDF products are closely integrated. In fact, it is always recommended to use SRDF in conjunction with remote TimeFinder to allow remote copies utilizing the target hardware resources without interrupting the SRDF replications. Also the remote copies can serve as a gold copy whenever an SRDF target needs to be refreshed. As an example, a remote TimeFinder/Clone can be created from the SRDF R2 devices, and many additional snaps can be created out of that clone for test, development, and reporting instances. When SRDF/A is used any remote TimeFinder operation should use the consistent split feature to coordinate the replica with SRDF/A cycle switching. The use cases in this paper illustrate some of the basic Oracle business continuity operations that TimeFinder and SRDF can perform together.

Symmetrix VMAX SRDF product family

Symmetrix Remote Data Facility (SRDF) is a Symmetrix-based business continuance and disaster restart solution. In simplest terms, SRDF is a configuration of multiple Symmetrix units whose purpose is to maintain real-time copies of host devices in more than one location. The Symmetrix units can be in the same room, in different buildings within the same campus, or hundreds of miles apart. SRDF provides data mobility and disaster restart spanning multiple host platforms, operating systems, and applications. It can scale to thousands of devices, can replicate while maintaining write-order consistency from multiple source arrays to multiple target arrays, and can support a variety of topologies and configurations.

The local SRDF device, known as the source (R1) device, is configured in a pairing relationship with a remote target (R2) device, forming an SRDF pair. When the R2 devices are mirrored with R1 devices, the R2 devices are write-disabled to the remote host. After the R2 devices are synchronized with its R1 devices, they can be split at any time, making the R2 devices fully accessible to their hosts. The R2 device can be either used directly by hosts (once they are split), can be restored incrementally to the R1 devices, or can be used in conjunction with TimeFinder to create additional replicas. TimeFinder replicas can be taken from the R2 devices even while SRDF is replicating, without disturbing the replication.

Many other new performance and scalability features were added to SRDF with Enginuity release 5874, including a new protection mode called SRDF/Extended Distance Protection (SRDF/EDP). Please refer to the SRDF product guide for a full description.

SRDF modes of operation

SRDF/Synchronous (SRDF/S), SRDF/Asynchronous (SRDF/A), and SRDF Adaptive Copy are the basic operation modes of SRDF. The first two are valid for Oracle database protection and maintain dependent write-order consistency. The third is useful for bulk data transfers or in combination with more complex SRDF solutions such as SRDF/Automated Replication (SRDF/AR)

SRDF/Synchronous mode

SRDF/S is used to create a no data loss solution of committed transactions. It provides the ability to replicate multiple databases and applications data remotely while guaranteeing the data on both the source and target devices is exactly the same. SRDF/S can protect single or multiple source Symmetrix storage arrays with synchronous replication.

With SRDF/S Synchronous replication, shown in Figure 4, each I/O from the local host to the source R1 devices is first written to the local Symmetrix cache (1) and then it is sent over the SRDF links to the remote Symmetrix unit (2). Once the remote Symmetrix unit acknowledged it received the I/O in its cache successfully (3), the I/O is acknowledged to the local host (4). Synchronous mode guarantees that the remote image is an exact duplication of the source R1 device's data.

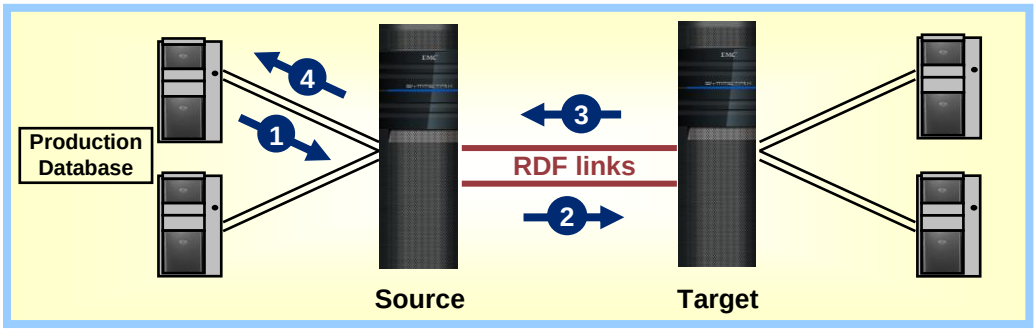


Figure 4. SRDF/Synchronous replication

Single Roundtrip and Concurrent Write SRDF performance enhancements

Starting with the Enginuity 5772 Service Release, SRDF/S provides a few performance enhancements. The first, *Single Roundtrip*, allows faster SRDF/S response time when long distances increase write latency. Where previously a transfer-ready condition state was required from the SRDF target before sending the actual data, now both transfer ready and data are sent in parallel and acknowledged once. The second, *Concurrent Write*, allows SRDF/S to send up to eight I/Os in parallel per each source device if the I/O arrives from different FA ports. This allows SRDF/S to perform much faster, for example, during Oracle checkpoints and when host multipathing tools like EMC PowerPath® are used.

SRDF/Asynchronous replication mode

SRDF/Asynchronous (SRDF/A) provides a consistent point-in-time image on the target (R2) devices that is only slightly behind the source (R1) devices. SRDF/A allows replication over unlimited distance, with minimum to no effect on the performance of the local production database(s). SRDF/A can "ride" through workload peaks by utilizing the local Symmetrix cache and optionally spilling data to a disk pool (also called *delta set extension*, or DSE) and reducing the link bandwidth requirements.

SRDF/A session data is transferred to the remote Symmetrix array in timed cycles, also called *delta sets*, as illustrated in Figure 5. There are three cycles that work in unison – the *capture* cycle receives all new I/O from the hosts, the *transmit/receive* cycles on the R1 and R2, respectively, send and receive the previous captured cycle until it is fully received, and the *apply* cycle applies a previously fully received cycle to the R2 devices.

The SRDF/A cycle switching process is very efficient and scalable. Within a capture cycle if a piece of data is updated multiple times only the most recent update to the data is transmitted once. This process is called

write folding. Also, there is no need to maintain write consistency of each I/O. Instead, consistency is maintained *between* cycles. If replication stops for any reason SRDF will make sure to either apply a fully received cycle to the target R2 devices, or discard the last incomplete cycle. This leaves the remote R2 devices always only one or two cycles behind the R1 devices. While the default minimum cycle switching time is 30 seconds, it can grow during peak workload, and shrink back to default afterward.

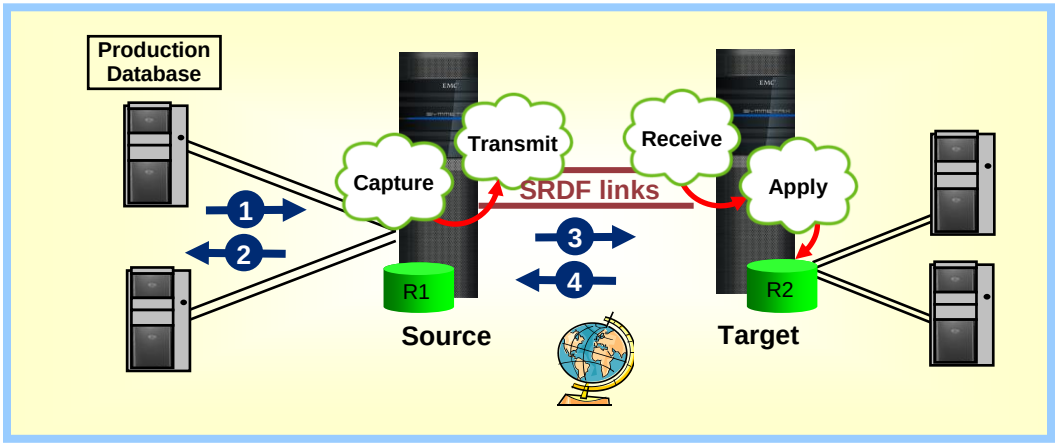


Figure 5. SRDF/Asynchronous replication

SRDF/A Consistency Exempt

New to Enginuity 5874 is the ability to add or remove devices from an SRDF/A session without breaking the session consistency to perform that operation. When dynamic SRDF devices are added the *consistency exempt* flag is set, allowing them to synchronize without interrupting the consistency attributes of the other devices in the SRDF/A session. After they are in sync for two cycles the flag will be automatically removed, allowing them to join the session consistency attributes. When devices are suspended the consistency exempt flag will be automatically set, thus allowing them to be removed without interrupting the SRDF session consistency. These new and flexible abilities enhance database protection and availability.

SRDF/A Multi-Session Consistency

Like SRDF/S, SRDF/A can replicate from multiple source arrays to multiple target arrays while maintaining write-order consistency between cycles. When dependent write consistently across multiple Symmetrix arrays is required, the SRDF/A Multi-Session Consistency (MSC) option is used and the coordination of cycle switching across the arrays is performed with the assistance of SRDF redundant host daemons. The daemons merely wait for ready conditions on all the arrays and then send the switch cycle command, keeping communication light and efficient. Similar to TimeFinder consistent split, also when SRDF/A MSC is used there is a brief hold of write I/O on all the arrays simultaneously during cycle switch to preserve write-order consistency.

SRDF Adaptive Copy replication mode

SRDF Adaptive Copy replication facilitates long-distance data sharing and migration (see Figure 6). SRDF Adaptive Copy replication allows the primary and secondary volumes to be more than one I/O out of synchronization. The maximum number of I/Os that can be out of synchronization is known as the maximum skew value, and can be set using SRDF monitoring and control software. There is no attempt to preserve the ordering of write I/Os when using SRDF Adaptive Copy replication.

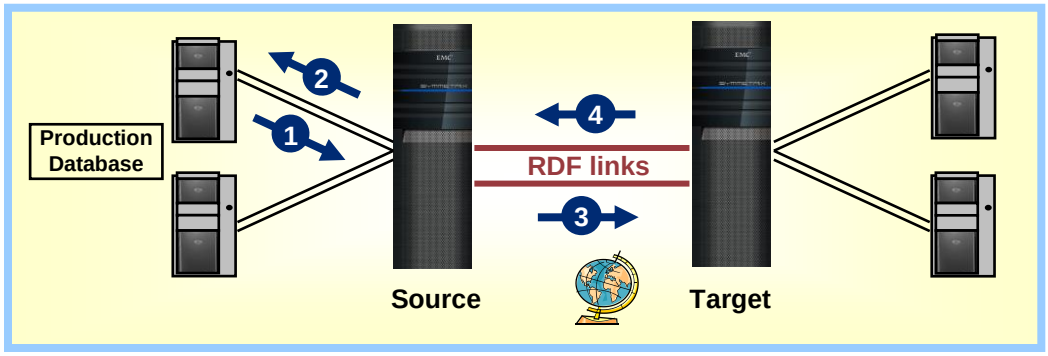


Figure 6. SRDF Adaptive Copy mode

SRDF Adaptive Copy replication is useful as an interim step before changing to an Oracle-supported SRDF/S or SRDF/A replication. It is also used for point-in-time long-distance bulk transfer of data. For example, if the connection between the two sides is lost for a long period of time allowing the buildup of a large number of changes to accumulate, resumption of the links can cause a heavy surge in link traffic (created by the backlog of changes added to those generated by normal production traffic). By using SRDF Adaptive Copy replication, the backlog of invalid tracks is synchronized using the SRDF low priority queue, while new writes are buffered in cache and sent across using the high priority SRDF queue without impacting the host application. Once the backlog of changes has been transferred, or the total amount of changed tracks has reached a specified number, the mode can be changed to SRDF/S or SRDF/A replication to achieve database protection.

SRDF Adaptive Copy replication is *not* supported for database restart or database recovery solutions with Oracle databases. Using SRDF Adaptive Copy replication by itself for disaster protection of Oracle databases will lead to a corrupt and unusable remote database.

SRDF topologies

SRDF can be set in many topologies other than the single SRDF source and target. Thus SRDF satisfies different needs for high availability and disaster restart. It can use a single target or two concurrent targets; it can provide a combination of synchronous and asynchronous replications; it can provide a three-site solution that allows no data loss over very long distances and more. Some of the basic topologies that can be used with SRDF are shown in the following section[1].

Concurrent SRDF

SRDF allows simultaneous replication of single R1 source devices to up to two target devices using multiple SRDF links. All SRDF links can operate in either Synchronous or Asynchronous mode or one or more links can utilize Adaptive Copy mode for efficient utilization of available bandwidth on that link. This topology allows simultaneous data protection over short and long distances.

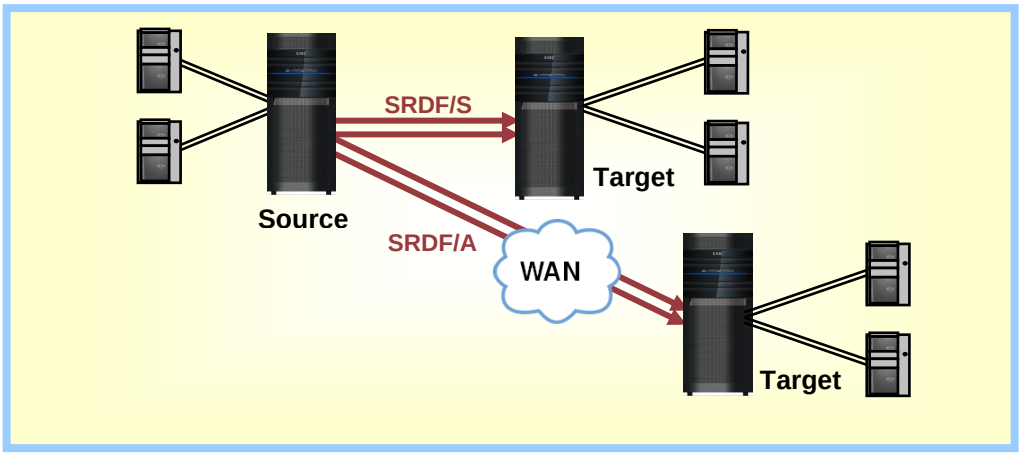


Figure 7. Concurrent SRDF

Cascaded SRDF

SRDF allows cascaded configurations in which data is propagated from one Symmetrix to the next. This configuration requires Synchronous mode for the first SRDF leg and Asynchronous or Adaptive Copy modes for the next. This topology provides remote replications over greater distances with varying degree of bandwidth utilization and none to limited data loss (depends on the choice of SRDF modes and disaster type).

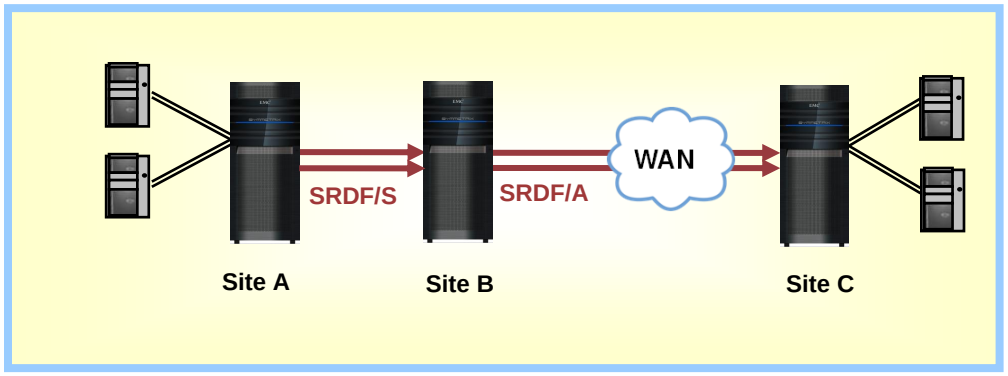


Figure 8. Cascaded SRDF

[1] For full coverage of SRDF topologies, please refer to SRDF product guide.

SRDF/Extended Distance Protection

SRDF currently supports multi-site replications in cascaded SRDF configuration. This feature is enhanced to support a more efficient two-site DR solution over extended distances with zero or near zero data loss. In this configuration the storage cache alone is used on the intermediate site for a temporary pass-through data store of the modified tracks before copying them over to the tertiary site. SRDF/S and Adaptive Copy are allowed between primary and secondary sites. SRDF/A and Adaptive Copy are available between secondary and tertiary sites.

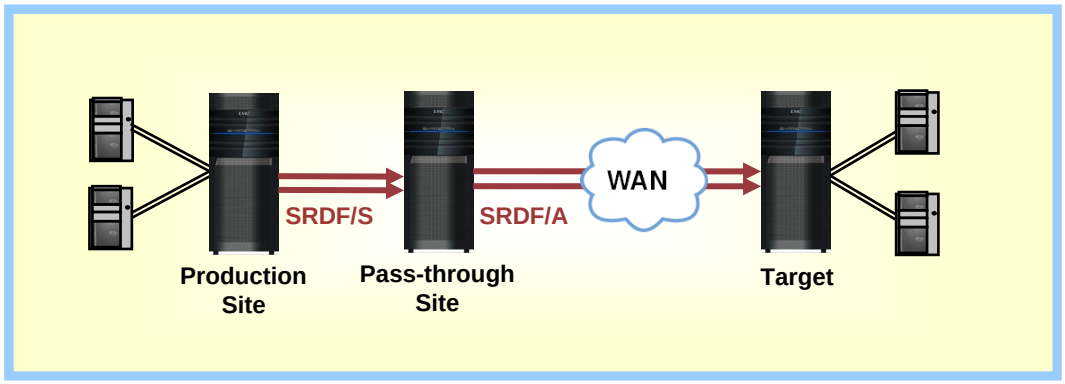


Figure 9. SRDF/Extended Distance Protection

The major benefits of this configuration are:

- New long-distance replication solution with the ability to achieve zero RPO at the target site
- A lower-cost alternative in which to achieve no data loss for target site disaster restart

SRDF/Star

SRDF/Star is a two- or three-site protection topology where data is replicated from source Site A to two other Symmetrix systems simultaneously (Site B and Site C). The data remains protected even in case one target site (B or C) goes down. If site A (the primary site) goes down, the customer can choose where to come up (site B or C) based on SRDF/Star information. If the storage data in the other surviving site is more current then changes will be incrementally sent to the surviving site that will come up. For protection and compliance, remote replications can start immediately to the new DR site. For example, if database operations resume in Site C, data will be sent first from Site B to create a no data loss solution, and then Site B will become the new DR target. SRDF/Star has a lot of flexibility and can change modes and topology to achieve best protection with each disaster scenario. For full description of the product refer to the SRDF product guide.

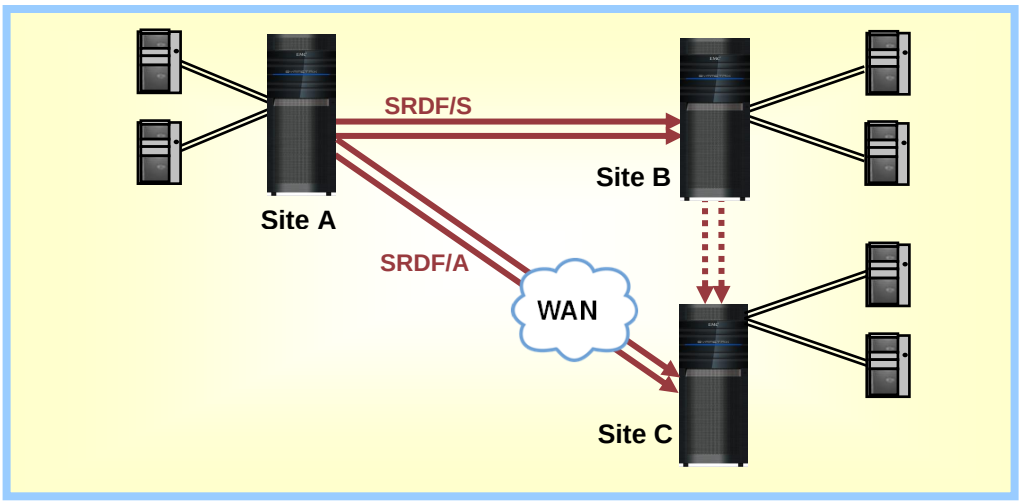


Figure 10. SRDF/Star

Leveraging TimeFinder and SRDF for data consistency

EMC TimeFinder and SRDF solutions with Enginuity Consistency Assist (ECA consistent split) allow creation of dependent write-order consistent storage-based replicas. The replicas are created by temporarily holding write I/Os to all source devices included in the replica. Since all writes are held, no dependent writes can be issued (as they depend on a previous completion of the held I/O). For example Oracle will not write to data files (checkpoint) until the redo writes for these data changes were fully recorded in the log files.

SRDF/S and SRDF/A modes ensure the dependent write-order consistency of the replication by synchronizing each and every dependent I/O (SRDF/S mode) or by synchronizing across cycles of transferred data (SRDF/A mode). In an actual disaster that leads to the loss of source location, database restart operations can be completed at the remote location without the delays associated with finding and applying recovery across applications in the correct sequence or to a coordinated time before the failure.

In addition to disaster restart benefits, SRDF significantly enhances disaster recovery operations by using fast and reliable replication technology to offload the Oracle backup operations to a remote site and later return the restored data to the local site as shown in the use cases section.

ASM rebalancing and consistency technology

ASM provides a seamless and nonintrusive mechanism to expand and shrink the diskgroup storage. When disk storage is added or removed, ASM will perform a redistribution (rebalancing) of the striped data[2]. This entire rebalance operation is done while the database is online, thus providing higher availability to the database. The main objective of the rebalance operation is to always provide an even distribution of file extents, workload, and data protection across all disks in the diskgroup.

With Symmetrix arrays as the storage, it is considered a best practice to use ASM external redundancy for data protection. The Symmetrix RAID protection will be utilized to provide RAID 1, RAID 5, or RAID 6 internal disk protection.

The split operation of storage-based replicas is sensitive to the rebalancing process, which may cause ASM diskgroup inconsistencies if the diskgroup device members are split at slightly different times. These inconsistencies are a result of possible ASM metadata changes occurring while a split operation is in process. Upon startup if ASM detects an inconsistency, metadata logs will be used to perform ASM instance recovery. In addition Oracle provides tools and procedural steps to avoid inconsistencies when

[2] A disk failure will also trigger a rebalance; however, this is specific to ASM failure groups.

splitting storage-based replicas; however, these procedures can be simplified and streamlined with the use of EMC consistency technology.

Since EMC consistent split technology suspends database I/O to preserve write ordering, it also has the side effect of preventing any ASM metadata changes during the split. Performing a consistent split will prevent ASM metadata inconsistencies during the replication process, eliminating the otherwise extra steps or possible unusable replica if ASM rebalance was active while performing a non-consistent split.

Leveraging TimeFinder and SRDF for business continuity solutions

Database storage layout and best practices

ASM and Solutions Enabler device planning

Table 1 shows the RAC database and Symmetrix device layout that was used in the use cases. All the devices (LUNs) were 50 GB in size and the database actual size was about 400 GB.

Table 1. ASM diskgroups, and Symmetrix device and composite groups

ASM diskgroups	Database devices	Recovery Device Groups (DG)	Restart Device Groups (DG)	SRDF Consistency Group (CG)
+DATA	18 LUNs x 50 GB	DATA_DG	DB_DG	ALL_CG
+REDO	4 LUNs x 50 GB	REDO_DG		
+FRA	3 LUNs x 50 GB	FRA_DG		

The database primary devices (also TimeFinder and SRDF source devices) were using Symmetrix RAID 1 protection. TimeFinder/Clone targets were using RAID 5 protection to improve storage utilization. SRDF target devices also used RAID 1 to match the same protection level as the primary database devices.

ASM general best practices[3]

- ASM was using external redundancy (no software mirroring) in accordance with EMC's recommendation of leveraging the Symmetrix array RAID protection instead.
- ASM was set with three diskgroups: +REDO (redo logs), +DATA (data, control, temp files), and +FRA (archives, flashback logs). Typically EMC recommends separating logs from data for performance monitoring and backup offload reasons. When SRDF is used, temp files can go to their own "+TEMP" diskgroup if replication bandwidth is limited as temp is not required for database restart or recovery. In these use cases, however, SRDF FC bandwidth was not an issue and temp files were included in the +DATA diskgroup. Finally, +FRA can typically use a lower-cost storage tier like SATA drives and therefore require their own diskgroup.

TimeFinder best practices

- Multiple Symmetrix device groups were used for TimeFinder/Clone (or snap) operations, allowing finer granularity of operations. For recovery solutions, data files (together with control files), log files, and archive logs each had their own DG, allowing the replica of each to take place at slightly different times as shown in the recovery use cases. For example, if a valid datafile's backup replica should be restored to production, and the production logs are intact, by separating the datafiles and logs to their own DG and ASM diskgroups, such a restore won't compromise the logs and full database recovery would be possible. For a restart solution, a single DG was used that includes all data (control) and log files, allowing them to be split consistently creating a restartable and consistent replica.
- Note that TimeFinder operations can span Symmetrix arrays. When that is the case instead of a device group (DG) a composite group (CG) should be used, following the exact same best practices as shown for the DG in this paper.
- It is recommended to issue TimeFinder and SRDF commands from a management (or the target) host and not the database production host. The reason is that in rare cases when consistent split is used, under heavy write activity Symmetrix management commands may be queued behind database writes, interfering with completing the replication and the replica will be deemed invalid.
- It is recommended to use Symmetrix Generic Name Services (GNS) and allow them to be replicated to the SRDF targets. GNS manages all the DG and CG definitions in the array and can replicate them to the SRDF target so the management host issuing TimeFinder and SRDF commands will be able to operate on the same CG and DG as the source (without having to re-create them).

[3] These ASM best practices can easily be applied to other volume managers, filesystems, or raw devices.

- For the sake of simplicity the use cases assume that GNS is used and replicated remotely. When remote TimeFinder or SRDF operations are used, they are issued on the target host. It is also possible to issue remote TimeFinder and SRDF commands from the local management host using the *–rdf* flag; however it requires the SRDF links to be functional.
- Note that remote TimeFinder replica creation from an SRDF/A target should always use the *–consistent* flag to coordinate SRDF/A cycle switching with the TimeFinder operation and simply put, guarantee that the replica is consistent.

SRDF best practices

- SRDF, whether synchronous or asynchronous, should always use a composite group (CG) with consistency enabled (also called a consistency group). While enabling consistency is a requirement for SRDF/A, it is a common misconception that SRDF/S being a synchronous replication doesn't benefit from it. However SRDF/S with consistency enabled will guarantee that if even a single source device can't replicate to its target, all the SRDF devices in that session will stop replicating, preserving the target consistent image.
- For SRDF replications a single CG was used that included all the database devices (data, control and log files). As shown in Table 1 it also included the FRA devices. SRDF on its own is a restart solution and since database crash recovery never uses archive logs there is no need to include FRA in the SRDF replications. However there are two reasons why they could be included. The first is if Flashback database functionality is required for the target. Replicating the FRA (and the flashback logs) in the same consistency group with the rest of the database allows its usage on the target of flashback functionality. The second reason is that to allow offload of backup images remotely, the archive logs are required (as shown in Use Case 6).
- It is always recommended to have a clone copy available at the SRDF target as a gold copy protection from *rolling disasters*. Rolling disasters is a term used when a first interruption to normal replication activities is followed by a secondary database failure on the source, leaving the database without an immediately available valid replica. For example, if SRDF replication was interrupted for any reason for a while (planned or unplanned) and changes were accumulated on the source, once the synchronization resumes and until the target is synchronized (SRDF/S) or consistent (SRDF/A), the target is not a valid database image. For that reason it is best practice before such resynchronization to take a TimeFinder gold copy replica at the target site, which will preserve the last valid image of the database, as a protection from rolling disasters.
- While the source database was clustered, since Oracle RAC is based on a shared storage architecture, by virtue of replicating all the database components (data, log, and control files) the target database has the option of being started in cluster, or non-clustered mode. Regardless of the choice, it is not recommended to replicate the cluster layer (voting disks or cluster configuration devices) since these contain local hosts and subnets information. It is best practice that if a cluster layer is required at the target hosts, it should be configured ahead of time, based on target hostnames and subnets, and therefore be ready to bring up the database whenever the time comes.

Use Case 1: Offloading database backups from production

This use case illustrates how to offload database backups from production to a local TimeFinder/Clone, and then using Oracle RMAN to perform further backup.

While the Oracle database is in hot backup mode on the production host, a TimeFinder/Clone activate is performed to create a recoverable replica of the database. This is a valid backup image that can be used to perform quick recovery of the Oracle database. The image can also be mounted to another host for RMAN backups.

High-level steps

1. Place the database in hot backup mode.
2. Activate the DATA_DG clone (with *–consistent* since ASM is used).
3. End hot backup mode.

4. Archive the current log.
5. Copy two backup control files to the FRA ASM diskgroup.
6. Activate the ARCHIVE_DG clone (with *–consistent* since ASM is used).
7. Optionally mount the clone devices on a backup host and perform RMAN backup.

Device groups used

DATA_DG and ARCH_DG

Detailed steps

On the production host

1. Place the production database in hot backup mode.

```
# export ORACLE_SID=RACDB1
# sqlplus "/ as sysdba"
SQL> alter database begin backup;
```

2. Activate the TimeFinder/Clone DATA_DG replica. The clone replica includes data and control files. Use *–consistent* with ASM or filesystems.

```
# symclone -dg DATA_DG -tgt -consistent activate
```

3. End hot backup mode.

```
SQL> alter database end backup;
```

4. Switch logs and archive the current log file.

```
SQL> alter system archive log current;
```

5. Create two backup control files and place them in the FRA diskgroup for convenience (RMAN syntax is shown, although SQL can be used as well). One will be used to mount the database for RMAN backup; the other will be saved with the backup set.

```
RMAN>run {
      allocate channel ctl_file type disk;
copy current controlfile to '+FRA/control_file/control_start';
copy current controlfile to '+FRA/control_file/control_bakup';
release channel ctl_file;
}
```

6. Activate the TimeFinder/Clone ARCHIVE_DG replica. The clone replica includes the archive logs and backup control files. Use *–consistent* with ASM or filesystems. If RMAN Catalog is used synchronize it first to register the most recent archive logs.

```
RMAN>resync catalog;
```

```
# symclone -g ARCH_DG -tgt -consistent activate
```

On the backup host

The database replica can be used as a valid disk backup or as a source for backup to a tertiary media such as tape or a disk library. In this example RMAN will be used to perform the backup.

Target/Backup host prerequisites:

- The ASM devices (or partitions) on clone volumes have correct Oracle permissions.
- The ASM_DISKSTRING parameter in the init.ora file for the ASM instance includes the path to clone volumes.
- The ASM_DISKGROUPS parameter in the init.ora file for the ASM instance contains the names of the production database diskgroups.
- It is not necessary to have the database mounted as RAC. Prior to mounting the database comment out, update ASM and database instance init.ora parameters as necessary. Specifically change CLUSTER_DATABASE to false if clustered mode is not needed. If the database is to be started in clustered mode then the cluster layer (and software) should already be installed and configured on the target host (not replicated with TimeFinder or SRDF)

7. (Continuing from step 6 on the previous page) Start the ASM instance. If other volume managers or filesystems are used their appropriate import and mount commands will be used instead. Make sure all the diskgroups were mounted correctly by ASM.

```
# export ORACLE_SID=+ASM
# sqlplus "/ as sysdba"
SQL> startup
```

8. Mount the database instance. A database backup that was taken with hot backup mode is valid for recovery only as long as it has not been opened read-writeable (with the *resetlogs* option). For that reason, it should be only mounted, which is the minimum prerequisite for RMAN backup. It can also be opened in read-only mode after enough archive logs are applied to resolve any data files' fuzziness. Before starting the database in mount mode, change the CONTROL_FILES in the init.ora file to point to the backup control file.

```
control_files = +FRA/control_file/control_start
```

```
# export ORACLE_SID=CLONE_DB
# sqlplus "/ as sysdba"
SQL> startup mount
```

9. Back up the database with RMAN from the backup host. The control file copy that was not used to mount the instance (*control_bak*) should be part of the backup set. The *control_start* file should not be backed up because the SCN will be updated when the database is mounted for backup.

```
RMAN>run {allocate channel t1 type disk;
      backup format 'ctl%d%s%p%t'
controlfilecopy '+FRA/control_file/control_bak';
backup full format 'db%d%s%p%t' database;
backup format 'al%d%s%p%t' archivelog all;
release channel t1;
}
```

Note: The format specifier %d is for date, %t for 4-byte timestamp, %s for backup set number, and %p for the backup piece number.

Use Case 2: Parallel database recovery

This use case illustrates how to perform parallel database recovery by restoring a local TimeFinder backup replica and applying logs to it, even while TimeFinder restore continues in the background.

The clone copy created in Use Case 1 can be used to perform database recovery of the production database. Database recovery operations can start as soon as TimeFinder/Clone restore operation has started, providing a much faster RTO compared to common solutions that require an initial restore of the backup image from the tertiary media destination, and only once it was fully restored, database recovery operations can start. Recovery can be performed using the archived logs available on the production host or restored from the TimeFinder/Clone image. Like in this example, if recovery takes place on production, and archive logs including even online redo logs are available, a full media recovery (no data loss) can be achieved. If the production logs (or not all archive logs) are available, database incomplete media recovery can still be performed.

High-level steps

1. Shut down production database and ASM instances.
2. Restore only the DATA_DG clone (split afterwards).
3. Start ASM.
4. Mount the database.
5. Perform database recovery and open the database.

Device group used

DATA_DG

Detailed steps

On the production host

1. Shut down any production database and ASM instances (if still running).

```
# export ORACLE_SID=RACDB1
# sqlplus "/ as sysdba"
SQL> shutdown abort

# export ORACLE_SID=+ASM1
# sqlplus "/ as sysdba"
SQL> shutdown abort
```

2. Restore the TimeFinder/Clone replica. Note the *–force* is required if the source device is also part of an active SRDF session with remote R2 devices. In this case it is assumed that production archive and redo logs are available, therefore, just the DATA_DG (with data and control files) is restored.
   As soon as the restore starts it is possible to continue with the next step. However make sure to split the clone replica at a later time when the background restore completed. Note that TimeFinder restore protects the replica from changes to the source devices.

```
# symclone -dg DATA_DG -tgt restore [-force]
# symclone -dg DATA_DG -tgt split
```

3. Start the ASM instance (follow the same activities as in Use Case 1, step 7).
4. Mount the database (follow the same activities as in Use Case 1, step 8).
5. Recover and open the production database. Use *resetlogs* if incomplete recovery was performed.

```
# export ORACLE_SID=RACDB1
# sqlplus "/ as sysdba"
SQL> startup mount
```

```
SQL> recover automatic database using backup controlfile until
cancel;
SQL> alter database open;
```

Use Case 3: Local restartable replicas of production

This use case illustrates how to create local restartable clones (or snaps) of production for database repurposing, such as creating test, development, and reporting copies.

While the Oracle database is running transactions on the production host, *without* the use of hot backup mode activate a consistent TimeFinder/Clone session to create a restartable replica of the database. The replica can be mounted to another host for purposes such as test, dev, reporting, and so on. Mounting multiple replicas of the same database on the same host is possible; however that topic is beyond the scope of this paper.

High-level steps

1. Activate the DB_DG clone (with –*consistent* to create restartable replica).
2. Start the ASM instance.
3. Start the database instance.
4. Optionally, refresh the clone replica from production at a later time.
5. Optionally, if CRS is used, register the database and its services with CRS for automation and ease of management (using the *srvctl* command syntax).

Device group used

DB_DG

Detailed steps

On the target host

1. Activate the TimeFinder/Clone DB_DG replica. The clone replica includes all data, control, and log files. Use –*consistent* to make sure the replica maintains dependent write consistency and therefore a valid restartable replica from which Oracle can simply perform crash recovery.

```
# symclone –dg DB_DG –tgt –consistent activate
```

Note: Follow the same target host prerequisites as in Use Case 1 prior to step 7.

2. Start the ASM instance (or perform import/mount if other volume managers or filesystems are used). Make sure all the diskgroups were mounted correctly by ASM.

```
# export ORACLE_SID=+ASM
# sqlplus "/ as sysdba"
SQL> startup
```

3. Simply start the database instance. No recovery or archive logs are needed.

```
# export ORACLE_SID=CLONE_DB
# sqlplus "/ as sysdba"
SQL> startup
```

At this point the clone database is opened and available for user connections.

4. Optionally, it is easy and fast to refresh the TimeFinder replica from production as TimeFinder/Clone operations are incremental as long as the clone session is not terminated. Once the clone session is reactivated, the target devices are available immediately for use, even if background copy is still taking place.

4.1. Shut down the clone database instance since it needs to be refreshed

```
SQL> shutdown abort
```

4.2. Re-create and activate the TimeFinder/Clone replica from production. This will initiate the background copy operation.

```
# symclone -dg DB_DG -tgt recreate
# symclone -dg DB_DG -tgt activate -consistent
```

4.3. Start the clone ASM and database instances by following steps 2 and 3 again.

5. Optionally, if CRS is used, register the database and services (if ASM is used, then register ASM as well).

```
srvctl add asm -n $NODE1 -i $ASM_INST1 -o $ORACLE_HOME
srvctl add asm -n $NODE2 -i $ASM_INST2 -o $ORACLE_HOME

srvctl add database -d $DB_NAME -o $ORACLE_HOME
srvctl add instance -d $DB_NAME -i $DB_INST1 -n $NODE1
srvctl add instance -d $DB_NAME -i $DB_INST2 -n $NODE2

# REVIEW CONFIGURATION
srvctl config database -d $DB_NAME -a

# ADD SERVICES
# srvctl add service -d $DB_NAME -s $DB_SERVICE -r $DB_INST1
# srvctl add service -d $DB_NAME -s $DB_SERVICE -r $DB_INST2
```

Use Case 4: Remote mirroring for disaster protection (synchronous and asynchronous)

This use case illustrates how to create remote mirrors of a production database for disaster protection using SRDF/S or SRDF/A.

High-level steps

1. Perform initial synchronization of SRDF in Adaptive Copy mode.
2. Once the SRDF target is close enough to the source, change the replication mode to SRDF/S or SRDF/A.
3. Enable SRDF consistency.

Device group used

ALL_CG

Detailed steps

1. Perform initial synchronization of SRDF in Adaptive Copy mode. Repeat this step or use the skew parameter until the SRDF target is close enough to the source.

```
# symrdf -cg ALL_CG set mode acp_wp skew <number>]
# symrdf -cg ALL_CG establish
```

2. Once the SRDF target is close enough to the source change the replication mode to SRDF/S or SRDF/A.

2.1. For SRDF/S, set protection mode to sync:

```
# symrdf -cg ALL_CG set mode sync
```

2.2. For SRDF/A, set protection mode to async:

```
# symrdf -cg ALL_CG set mode async
```

3. Establish SRDF replication if the copy is not already active and enable consistency.

```
# symrdf -cg ALL_CG enable
# symrdf -cg ALL_CG establish [-full]
# symrdf -cg ALL_CG verify -synchronized -i 60
```

Use Case 5: Remote restartable database replicas for repurposing

This use case illustrates how to create remote restartable clones (or snaps[4]) of production for database repurposing without interrupting SRDF protection

Once synchronized, an SRDF/S or SRDF/A session can be split at any time to create the dependent write consistent remote replica based on the R2 target devices. At that time SRDF will keep track of any changes on both source and target devices and only these changes will be copied over the next time SRDF is synchronized (refresh the target devices) or restored (refresh the source devices).

However it is a better practice to keep SRDF synchronized to maintain remote replication and protection, and instead activate a remote TimeFinder replica such as clone or snap (currently supported with SRDF/S only), and alternatively additional snapshots can be taken from the remote clone. These replicas of the database are dependent write consistent and can be used for activities such as test, development, reporting, data processing, publishing, and more. It also can serve as gold copy protection from rolling disasters as explained earlier in the SRDF best practices section.

High-level steps

1. Activate the *remote* DB_DG clone (use –*consistent* to create restartable replica).
2. Start the *remote* ASM instance.
3. Start the *remote* database instance.
4. Optionally, refresh the *remote* clone replica from production (SRDF targets) at a later time.

Device group used

DB_DG

Detailed steps

On the target host

1. Activate the TimeFinder/Clone DB_DG *remote* replica. The clone replica includes all data, control, and log files. Use –*consistent* to make sure the replica maintains dependent write consistency and therefore a valid restartable replica from which Oracle can simply perform crash recovery.

```
# symclone -dg DB_DG -tgt -consistent activate
```

Note: Follow the same target host prerequisites as in Use Case 1 prior to step 7.

2. Start the ASM instance. Follow the same activities as in Use Case 3 step 2.
3. Start the database instance. Follow the same activities as in Use Case 3 step 3.

[4] With Enginuity 5874, TimeFinder/Snap is not supported off a synchronized SRDF/A target. In general it is preferred to use a clone of the R2 instead, or create multi-writeable snapshots from that clone.

At this point the clone database is opened and available for user connections.

4. Optionally, to refresh the database clone follow the same activities as in Use Case 3 step 4.

Use Case 6: Remote database valid backup replicas

This use case illustrates how to create remote database clones that are a valid Oracle backup image and can be used for database recovery.

By creating TimeFinder remote replicas that are valid for database recovery, backup to tertiary media can be performed at the remote site. Also, the TimeFinder replica itself is a valid backup to disk that can be used to recover production if necessary.

Note for SRDF/A: The SRDF *checkpoint* command will return control to the user only after the source device content reached the SRDF target devices (SRDF will simply wait two delta sets). This is useful for example when production is placed in hot backup mode before the remote clone is taken.

High-level steps

1. Place the database in hot backup mode.
2. *If using SRDF/A*, perform SRDF *checkpoint* (no action required for SRDF/S).
3. Activate a *remote* DATA_DG clone (with –*consistent* if SRDF/A and/or ASM are used).
4. End hot backup mode.
5. Archive the current log.
6. Copy two backup control files to the FRA ASM diskgroup.
7. *If using SRDF/A* then perform SRDF *checkpoint* (no action required for SRDF/S).
8. Activate the *remote* ARCHIVE_DG clone (with –*consistent* if SRDF/A and/or ASM is used).
9. Optionally mount the *remote* clone devices on the backup host and perform RMAN backup.

Device groups used

DATA_DG and ARCH_DG for TimeFinder operations, ALL_CG for SRDF operations

Detailed steps

On the production host

1. Place production in hot backup mode. Follow the same activities as in Use Case 1 step 1.
2. If SRDF/A is used then an SRDF *checkpoint* command will make sure the SRDF target has the datafiles in backup mode as well.

```
# symrdf –cg ALL_CG checkpoint
```

3. Activate the *remote* DATA_DG clone. Use –*consistent* if SRDF/A is used and/or ASM. Follow the same activities as in Use Case 1 step 2.
4. End hot backup mode. Follow the same activities as in Use Case 1 step 3.
5. Switch logs and archive the current log file. Follow the same activities as in Use Case 1 step 4.
6. Create two backup control files and place in the FRA diskgroup for convenience. Follow the same activities as in Use Case 1 step 5.
7. If SRDF/A is used then an SRDF *checkpoint* command will make sure the SRDF target has the FRA diskgroup (with the last archives and backup control files) at the target.

```
# symrdf –cg ALL_CG checkpoint
```

8. Activate the *remote* TimeFinder/Clone ARCHIVE_DG replica. Follow the same activities as in Use Case 1 step 6.

9. Optionally mount the *remote* clone devices on the backup host and perform RMAN backup. Follow the same activities as in the "On the backup host" section in Use Case 1.

Use Case 7: Parallel database recovery from remote backup replicas

This use case illustrates how to perform parallel production database recovery by restoring a remote TimeFinder/Clone backup image simultaneously with SRDF restore, and then applying Oracle logs to the production database in parallel. This is similar to Use Case 2, only the recovery is from a remote replica.

High-level steps

1. Shut down production database and ASM instances.
2. Restore the *remote* DATA_DG clone (split afterwards). Restore SRDF in parallel.
3. Start ASM.
4. Mount the database.
5. Perform database recovery (possibly while the TimeFinder and SRDF restore are still taking place) and open the database.

Device groups used

DATA_DG; ALL_CG for SRDF operations

Detailed steps

On the production host

1. Shut down any production database and ASM instances (if still running). Follow the same activities as in Use Case 2 step 1.
2. If SRDF is still replicating from the source to target then stop the replication.

```
symrdf -cg ALL_CG split
```

3. Start the restore of the *remote* TimeFinder/Clone replica to the SRDF target devices. You do not need to wait for it to finish before moving to the next step.

```
symclone -dg DATA_DG -tgt restore [-force]
```

4. Start the SRDF restore in parallel.

```
# symrdf -cg ALL_CG restore
```

   In some cases, the distance may be long, bandwidth may be limited, and many changes have to be restored. In these cases it might make more sense to change SRDF mode to Adaptive Copy first until the differences are small before placing it again in SRDF/S or SRDF/A mode.

5. Start ASM on the production host. Follow the same activities as in Use Case 1 step 7.
6. Mount the database. Follow the same activities as in Use Case 1 step 8.
7. Recover and open the production database. Follow the same activities as in Use Case 2 step 5.

Use Case 8: Fast database recovery from a restartable replicas

This use case illustrates fast database recovery by using the most recent consistent (restartable) replica and applying logs to it.

Oracle supports various database recovery scenarios based on dependent write consistent storage replicas created using SRDF and/or TimeFinder. Oracle support is covered in metalink note ID 604683.1**.** The purpose of this use case is not to replace backup strategy such as nightly backups based on hot backup mode. Instead, it offers a complementary use case such as when RTO requirements are very strict. It could be a compelling solution to run the database in archivelog mode, and perform periodic snapshots without

placing the database in hot backup mode. If recovery is required, the last snapshot is restored and in parallel the limited transactions since that snapshot was taken are restored, creating a fast database recovery solution.

Consider this scenario. The database is in archive log mode and periodic TimeFinder consistent clones or snaps are created that include only the data. At some point a database recovery is required based on the last replica (clone in this example).

High-level steps

1. Shut down production database and ASM instances.
2. Restore the most recent DATA_DG clone (split afterwards).
3. Start ASM.
4. Mount the database.
5. Only in the case of incomplete recovery is it necessary to perform a scan to update data file headers.
6. Perform database full or incomplete recovery (possibly while the TimeFinder background restore is still taking place).

Device group used

DATA_DG

Detailed steps

1. Shut down any production database and ASM instances (if still running). Follow the same activities as mentioned in Use Case 2 step 1.
2. Restore the most recent DATA_DG TimeFinder replica. Follow the same activities as mentioned in Use Case 2 step 2.
3. Start the ASM instance (follow the same activities as in Use Case 1 step 7).
4. Mount the database (follow the same activities as in Use case 1 step 8).
5. If incomplete recovery is required, perform `dbms_backup_restore.scandatafile()` as shown in the example below (data files can be scanned in parallel to reduce the overall time). Full recovery (when redo logs are available) does not require the scan operation.
6. Perform database recovery based on one of the following options.

Full (complete) database recovery

When all online redo logs and archive logs are available it is possible to perform a full media recovery of the Oracle database to achieve a no data loss of committed transactions.

```
SQL> recover automatic database;
SQL> alter database open;
```

Note: It might be necessary to point the location of the online redo logs or archive logs if the recovery process didn't locate them automatically (common in RAC implementations with multiple online or archive logs locations). The goal is to apply any necessary archive logs as well as the online logs fully.

Point-in-time database recovery

When a full media recovery is not desirable, or when some archives or online logs are missing, an incomplete recovery can be performed. When performing incomplete recovery enough logs need to be applied to pass the maximum point of data file fuzziness so they are all consistent. After passing that point additional archive can potentially be applied. The following is a sample script (based on the Oracle metalink note mentioned previously) that can help identify the minimum SCN required to open the database, and update the data file headers accordingly.

Scan datafile script:

```
spool scandatafile.out
set serveroutput on
declare
  scn number(12) := 0;
  scnmax number(12) := 0;
begin
  for f in (select * from v$datafile) loop
    scn := dbms_backup_restore.scandatafile(f.file#);
    dbms_output.put_line('File ' || f.file# ||' absolute fuzzy
scn = ' || scn);
    if scn > scnmax then scnmax := scn; end if;
  end loop;

  dbms_output.put_line('Minimum PITR SCN = ' || scnmax);
end;
```

Sample output generated by the scan data script:

```
SQL> @./scandata.sql
File 1 absolute fuzzy scn = 27088040
File 2 absolute fuzzy scn = 27144475
File 3 absolute fuzzy scn = 27164171
…
File 22 absolute fuzzy scn = 0
Minimum PITR SCN = 27164171
```

Perform incomplete database recovery (sample commands):

```
SQL> alter database recover database until change 27164171;
SQL> alter database open resetlogs;
```

Conclusion

Symmetrix VMAX is a new offering in the Symmetrix family with enhanced scalability, performance, availability, and security features. It facilitates deployment of Oracle databases and applications to be deployed rapidly and with ease.

With the introduction of enterprise Flash drives, and together with Fibre Channel and SATA drives, Symmetrix provides a consolidation platform covering performance, capacity, and cost requirements of small and large databases. The correct use of storage tiers together with the ability to move data seamlessly between tiers allow customers to place their most active data on the fastest tiers, and their less active data on high-density, low-cost media like SATA drives. Features such as Auto-provisioning allow ease of storage provisioning to Oracle databases, clusters, and physical or virtual server farms.

TimeFinder and SRDF technologies simplify high availability and disaster protection of Oracle databases and applications, and provide the required level of scalability from the smallest to the largest databases. SRDF and TimeFinder are easy to deploy and very well integrated with Oracle products like Automatic Storage Management (ASM), RMAN, Grid Control, and more. The ability to offload backups from production, rapidly restore backup images, or create restartable database clones enhances the Oracle user experience and data availability.

Oracle and EMC have been investing in an engineering partnership to innovate and integrate both technologies since 1995. The integrated solutions increase database availability, enhance disaster recovery strategy, reduce backup impact on production, minimize cost, and improve storage utilization across a single database instance or RAC environments.

Appendix: Test storage and database configuration

This appendix contains a description of the storage and database configurations used in the test use cases.

General test environment

It is assumed that:

- Oracle is installed on the target host with similar options to production and configured for ASM use (CSS, or Cluster Synchronization Service, is active).
- Copies of the production init.ora files for the ASM instance and the database instance were copied to the target host and modified if required to fit the target host environment.
- The appropriate Clone, R2, or Remote Clone (whichever is appropriate for the test) is accessible by the target host.

The SRDF and TimeFinder tests were performed while an OLTP workload was running, simulating a high number of concurrent Oracle users.

Although, TimeFinder and SRDF commands can be issued from any host connected to the Symmetrix, in the following test cases, unless specified otherwise, they were issued from the production host. The term "Production host" is used to specify the primary host where the source devices are used, and "Target host" is used to specify the host where the Clones, R2, or Remote clone devices are used.

Test setup

Figure 11 depicts the test setup containing Oracle RAC on the production site and associated TimeFinder/Clone and SRDF devices for local and remote replication.

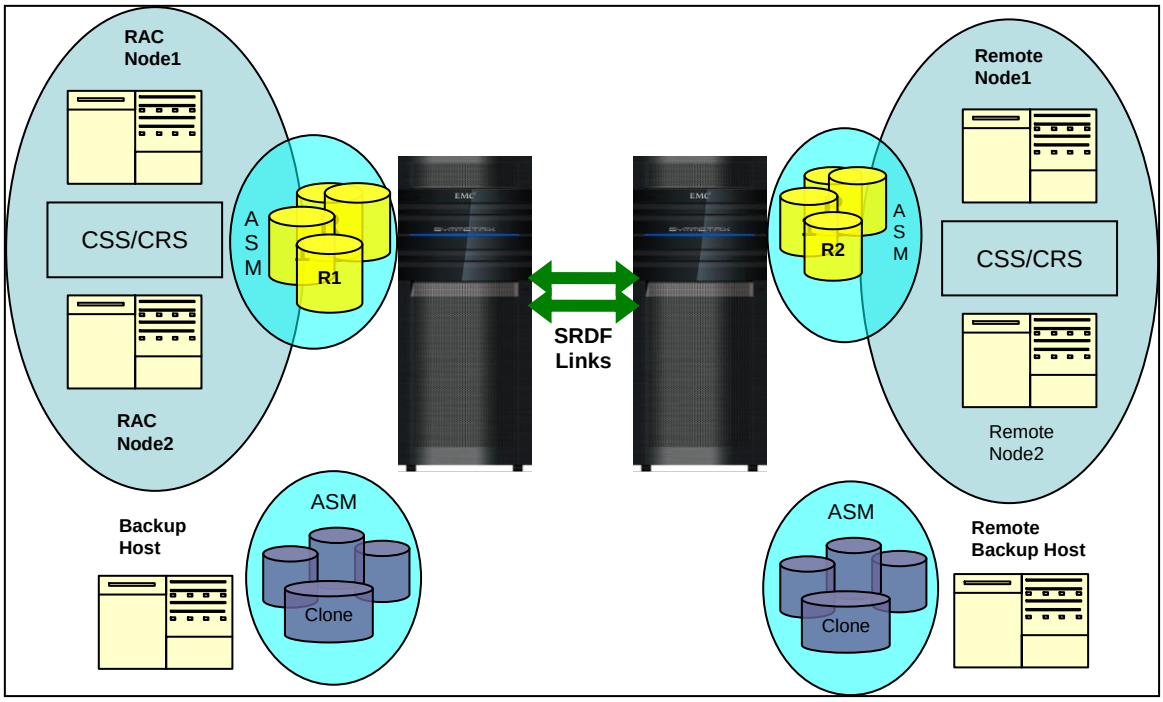


Figure 11. Test configuration

Storage and device specific configuration:

- All RAC nodes share the same set of devices and have proper ownerships.
- PowerPath is used to support multipathing and load balancing.
- PowerPath device names are uniform across all RAC nodes.
- Symmetrix device groups are created for shared storage for RAC.
- ASM diskgroups were configured on Symmetrix devices.
- Appropriate local and remote replication relationships were created using SYMCLI commands for TimeFinder/Clone and SRDF.

Table 2. Test hardware

	Model	OS	Oracle version
Local "Production" Host: RAC Node 1	Dell	Red Hat Enterprise Linux 5.0	11g release 1 (11.1.0.6.0)
Local "Production" Host: RAC Node 2	Dell	Red Hat Enterprise Linux 5.0	11g release 1 (11.1.0.6.0)
Remote "Target" Host	Dell	Red Hat Enterprise Linux 5.0	11g release 1 (11.1.0.6.0)

	Type	Enginuity version
Symmetrix	VMAX	5874
Symmetrix	VMAX	5874