PROTECTPOINT 3.1 FILE SYSTEM AGENT WITH VMAX – BACKUP & RECOVERY BEST PRACTICE FOR ORACLE ON ASM

VMAX® Engineering White Paper

ABSTRACT

The Dell EMC® ProtectPoint™ File System Agent integration between VMAX® and Data Domain® allows very significant backup and restore efficiencies for Oracle databases residing on ASM.

October, 2016

Published October 2016

ProtectPoint 3.1 File System Agent with VMAX – Backup and Recovery Best Practice for Oracle on ASM

Part Number H14777.1

TABLE OF CONTENTS

EXECUTIVE SUMMARY

Many applications are required to be fully operational 24x7x365, even as their data continues to grow. At the same time, their RPO and RTO requirements are becoming more stringent. As a result, there is an increasing demand for faster and more efficient data protection.

Traditional back up methods are unable to satisfy this demand due to the long duration and inefficiencies of reading and writing data during full backups. More importantly, during recovery, the recovery process itself (transactions roll forward) cannot start until the initial image of the database is fully restored, which can take a very long time.

This has led many data centers to use storage snapshots for more efficient protection. Storage snapshots provide fast backups and restores. However, unless the snapshot is taken from a remote storage array in sync with the production array, the snapshot data remains within the production storage array; increasing the risk of loss of both snapshots and primary storage if the production system was not available. In addition, it is advantageous to store backups in media that does not consume primary storage, and to benefit from features such as deduplication, compression, and remote replications, which the EMC Data Domain® systems offer.

EMC ProtectPoint™ addresses these gaps by integrating best-in-class Dell EMC products, the VMAX3™ and VMAX® All Flash storage arrays with the Data Domain systems, making the backup and recovery process more automated and efficient.

ProtectPoint allows Oracle Database backup and restore to take place directly between VMAX storage arrays and Data Domain. This capability not only reduces host I/O and CPU overhead, allowing the host to focus on servicing database transactions, but also provides greater efficiency for the backup and recovery process.

Backup efficiencies are introduced by *not requiring any read or write* I/Os of the data files by the host. Instead, VMAX TimeFinder® SnapVX™ creates a snapshot which is a valid backup of the database, and then sends it directly to the Data Domain encapsulated *backup* devices. Although each snapshot provides the current state of the database (that is, a 'full backup'), *only storage changes* from the time of the last backup are sent to Data Domain. It is often referred to as full backup at the cost of incremental data transfers.

Restore efficiencies are introduced in a similar way by *not requiring any read* or *write* I/Os of the data files by the host. Instead, Data Domain places the required backup ID's data on its *restore* devices, which can be made visible to a Mount host for small-scale data retrievals, or mounted to the Production host and cataloged with Oracle Recovery Manager (RMAN), so the RMAN *recover* command can be used to recover the Production database directly.

ProtectPoint 3.1 also offers direct access by hosts to the Data Domain *restore* devices even without a VMAX, a shared pool of the *restore* devices that can service restores from different sets of source *backup* devices, and a rollback feature that automatically restores the backup to Production.

Note: As ProtectPoint supports both VMAX3 and VMAX All Flash systems, the term VMAX is used throughout the paper, referring to both families of VMAX storage array equally.

Note: This white paper addresses the *values and best practices* of ProtectPoint File System Agent v3.1 and VMAX, *where the Oracle Database resides on ASM*. It does not cover the ProtectPoint application agent for Oracle Database residing on file systems.

AUDIENCE

This white paper is intended for database and system administrators, storage administrators, and system architects who are responsible for implementing, managing, and maintaining Oracle Database backup and recovery strategy with VMAX storage arrays. It is assumed that readers have some familiarity with Oracle Database and the VMAX storage, and are interested in achieving higher database availability, performance, and ease of storage management.

PRODUCT OVERVIEW

TERMINOLOGY

The following table explains important terms used in this paper.

Term	Description
Oracle Automatic Storage Management (ASM)	Oracle ASM is a volume manager and a virtual file system for Oracle Database files that supports single-instance Oracle Database and Oracle Real Application Clusters (RAC) configurations. Oracle ASM is Oracle's recommended storage management solution that provides an alternative to conventional volume managers, file systems, and raw devices.
Oracle Real Application Clusters (RAC)	Oracle RAC is a clustered version of Oracle Database based on a comprehensive high-availability stack that can be used as the foundation of a database cloud system as well as a shared infrastructure, ensuring high availability, scalability, and agility for applications.
Restartable vs. Recoverable database	Oracle distinguishes between a *restartable* and *recoverable* state of the database. A *restartable* state requires all log, data, and control files to be consistent (see 'Storage consistent replications' in this table). Oracle can be simply started, performing automatic crash/instance recovery without user intervention. *Recoverable* state on the other hand requires a database media recovery, rolling forward transaction logs to achieve data consistency before the database can be opened.
RTO and RPO	Recovery Time Objective (RTO) refers to the time it takes to recover a database after a failure. Recovery Point Objective (RPO) refers to any amount of data loss after the recovery completes, where RPO=0 means no data loss of committed transactions.
Storage consistent replications	Storage consistent replications refer to storage replications (local or remote) in which the target devices maintain write-order fidelity. That means that for any two dependent I/Os that the application issue, such as log write followed by data update, either both will be included in the replica, or only the first. To Oracle Database, SnapVX consistent snapshot data looks like a host crash, or Oracle 'shutdown abort', a state from which Oracle can simply recover by performing crash/instance recovery when starting. Starting with Oracle Database 11g, Oracle allows database recovery from storage consistent replications without the use of hot-backup mode (details in Oracle support note: 604683.1). The feature has become more integrated with Oracle Database 12c and is called Oracle Storage Snapshot Optimization.
VMAX FAST.X	FAST.X (formerly known as Federated Tiered Storage, or FTS) is a feature of VMAX that allows an external storage system to be connected to the VMAX backend and provides physical capacity that is managed by VMAX software. ProtectPoint uses this feature to create highly available and fast FC-based connectivity between VMAX and Data Domain.
VMAX HYPERMAX OS	HYPERMAX OS is the industry's first open converged storage hypervisor and operating system. It enables VMAX to embed storage infrastructure services like cloud access, data mobility and data protection directly on the array. This delivers new levels of data center efficiency and consolidation by reducing footprint and energy requirements.

VMAX storage group (SG)	A collection of host addressable VMAX devices. An SG can be used to: (a) Present devices to host (LUN masking). (b) Manage grouping of devices for replications such as when using SnapVX and SRDF® (c) Monitor the performance of the devices as a group. SGs can be cascaded, such as the parent SG represents the whole database and used for LUN masking, or remote replications. The child SGs represent database components (for example, data, redo logs, archives logs) and used for backup and recovery with storage snapshots, or for granular performance monitoring.
VMAX TimeFinder SnapVX	TimeFinder SnapVX is the latest generation in TimeFinder local replication software, offering high-scale, in-memory pointer-based, consistent snapshots.

VMAX STORAGE ARRAY

INTRODUCTION TO VMAX STORAGE ARRAY

The VMAX family of storage arrays is built on the strategy of simple, intelligent, modular storage, and incorporates a Dynamic Virtual Matrix interface that connects and shares resources across all VMAX engines, allowing the storage array to seamlessly grow from an entry-level configuration into the world's largest storage array. It provides the highest levels of performance, scale, and availability featuring new hardware and software capabilities.

The VMAX3 family is based on hybrid storage arrays that can contain both flash and spinning drives. VMAX3 family members are VMAX 100K, 200K, and 400K. VMAX3 uses Fully Automated Storage Tiering, or FAST, to place the most active data in flash tier, and the least active data in other storage tiers, based on user-provided service levels.

In 2016, Dell EMC announced new VMAX® All Flash products: VMAX 250F[1], VMAX 450F, and VMAX 850F. The new VMAX architecture is designed to take advantage of the latest, most cost-efficient 3D NAND flash drive technology. It features multi-dimensional scale, large write-cache buffering, back-end write aggregation, high bandwidth, and low latency.

Figure 1: VMAX All Flash storage arrays 450F/850F (left), and 250F (right)

[1] VMAX 450F and 850F were introduced in CY Q1 2016, and VMAX 250F was introduced in CY Q4 2016.

The key VMAX benefits for Oracle Database are:

- Large dynamic random-access memory (DRAM)-based cache enables all writes, including Oracle log writes or batch loads, to complete transactions extremely fast. VMAX cache is a large, mirrored cache which allows for write buffering[2]. As a result, writes to VMAX are faster than other flash arrays that write directly to SSD.

- The *write folding* feature reduces the amount of writes to the SSD drives, which also helps to extend their lives (compared to other flash arrays). Write folding allows buffering of all writes in VMAX cache, often through many repeating database updates (checkpoints). VMAX cache only commits the *latest* changes to the flash media periodically. When VMAX commits data to disk, the *write coalescing* feature allows it to issue larger writes. Thus database I/Os benefit from VMAX cache while the flash media benefits from smaller number of writes.

- The FlashBoost feature helps to improve the response time on random read misses of data not in cache. With FlashBoost, small-block reads (OLTP type workload) bypass the VMAX cache as the data is transferred directly from the back-end (flash) to front-end (host ports). Only then the data is staged in the VMAX cache for future access.

- Dell EMC VMAX TimeFinder SnapVX offers new levels of scale, efficiency, and simplicity. It can use redirect-on-write for added performance, and in-memory pointer-based management for deduplication-like data reduction. In addition, snapshots have names, versions, dates, and automatic expiration. Snapshots are protected, so they can be re-used regardless of changes by the application. They can also cascade any number of times. With SnapVX, Oracle Database replicas can be used to create gold copies for test/dev environments, or for backup and recovery. Snapshots can be restored in seconds, and read/write access to data is always immediate.

- Dell EMC VMAX Symmetrix Remote Data Facility (SRDF®) offers many disaster recovery topologies for Oracle Database, including two, three, and four site replication. SRDF can replicate in synchronous mode, asynchronous mode, cascaded, and zero data loss at any distance with SRDF/STAR. SRDF/Metro creates an active/active high availability solution, using Oracle Extended RAC or virtual machine migration across data centers. SRDF is closely integrated with SnapVX to offer a variety of high availability (HA) and disaster recovery (DR) solutions including remote backup offload and recovery.

- T10 Data Integrity Field (DIF) provides corruption detection and protection for all data, from the time it enters the array until it leaves, including local and remote replications. By using a supported stack, Oracle and Dell EMC integrated the T10-DIF standard for full end-to-end data integrity validation of all database I/O in real time. In addition VMAX offers an optional transparent Data at Rest Encryption (D@RE).

- Dell EMC ProtectPoint is an integration between VMAX and Data Domain that allows Oracle backups and restores to take place directly between the integrated VMAX and Data Domain systems – without any host I/Os to copy data. ProtectPoint reduces both backup time and recovery time, regardless of the size of the database. Only data changes are copied during the backup, or restore.

VMAX FAST.X

FAST.X is a feature of VMAX that allows external storage to be connected to the VMAX backend and provide physical capacity that is managed by VMAX software. Attaching external storage to a VMAX enables the use of physical disk capacity on a storage system that is not a VMAX array, while gaining access to VMAX features, including cache optimizations, local and remote replications, data management, and data migration.

The external storage devices can be *encapsulated* by VMAX, and therefore their data preserved and independent of VMAX-specific structures, or presented as raw disks to VMAX, where HYPERMAX OS will initialize them and create native VMAX device structures.

The encapsulation is implemented entirely within HYPERMAX OS and does not require any additional hardware besides the VMAX and the external storage. Connectivity with the external array is established using fibre channel ports.

Note: While the external storage is presented to and managed by VMAX HYPERMAX OS, and benefits from many of the VMAX features and capabilities, the assumption is that the external storage provides storage protection and therefore VMAX will not add its own RAID to the external storage devices.

[2] This makes VMAX cache considered persistent as in the case of power failure it will vault its content to flash and will restore it when the powers come back.

By using FAST.X, VMAX and Data Domain become an integrated system in which TimeFinder SnapVX local replication technology operates in coordination with Data Domain, using ProtectPoint File System Agent software and providing a powerful Oracle Database backup and recovery solution.

DATA DOMAIN SYSTEM

INTRODUCTION TO DATA DOMAIN

Data Domain deduplication storage systems offer a cost-effective alternative to tape that allows users to enjoy the retention and recovery benefits of inline deduplication, as well as network-efficient replication over the wide area network (WAN) for disaster recovery (DR).

Figure 2: Dell EMC Data Domain deduplication storage system

Data Domain systems reduce the amount of disk storage needed to retain and protect data by 10 to 30 times. Data on disk is available online and onsite for longer retention periods, and restores become fast and reliable. Storing only unique data on disk also means that data can be cost-effectively replicated over existing networks to remote sites for DR. With the industry's fastest deduplication storage controller, Data Domain systems allow more backups to complete faster while putting less pressure on limited backup windows.

All Data Domain systems are built as the data store of last resort, which is enabled by the Dell EMC Data Domain Data Invulnerability Architecture – end-to-end data verification, continuous fault detection and self-healing, and other resiliency features transparent to the application.

UNDERSTANDING DATA DOMAIN DEVICE ENCAPSULATION AND SNAPVX RELATIONSHIP

The Data Domain devices are *encapsulated* within VMAX to preserve their data structures. In that way, the backups in Data Domain system are independent and can be used not only by the original VMAX system, but also by other storage systems if necessary.

The ability to encapsulate Data Domain devices as VMAX devices allows TimeFinder SnapVX to operate on them.

UNDERSTANDING DATA DOMAIN *BACKUP* AND *RESTORE* DEVICES

The VMAX integration with Data Domain uses two sets of encapsulated devices: *backup* devices, and *restore* devices.

- The encapsulated *backup* devices are used as a backup target, and therefore ProtectPoint uses SnapVX to copy the backup data to them. After the incremental copy completes, Data Domain creates a static image from each of the *backup* devices, and together with the appropriate metadata, the static images create a backup-set, benefiting from deduplication, compression, and optional remote replications capabilities. A ProtectPoint catalog of backup sets can be listed, showing for each backup its *backup-id, description, and backup-time.* The backup-time corresponds with the time of the snapshot from which the backup was created.

- The encapsulated *restore* devices are used for database *restore* operations. They can be mounted directly to Production or a Mount host, or their data can be copied with ProtectPoint *rollback* to the VMAX Production devices, overwriting them.

  Note: The restore devices cannot be used for additional SnapVX operations outside of ProtectPoint (for example, using Solutions Enabler to create an additional snapshot from the encapsulated restore devices and then *link –copy* it to another set of target devices). The reason is that ProtectPoint places device locks on them after *restore prepare* that prevent such operations.

For more information on Data Domain refer to: http://www.emc.com/data-protection/data-domain/index.htm.

DATA DOMAIN BLOCK DEVICE SERVICE

Data Domain supports a variety of protocols, including CIFS, NFS, VTL, and now also a block device service that enables it to expose devices as FC targets. The block device service in Data Domain is called vdisk and allows the creation of *backup* and *restore* Data Domain devices that can be encapsulated by VMAX and used by ProtectPoint.

Table 1 lists the basic Data Domain vdisk block device object hierarchy, which is also shown in Figure 3.

Table 1 Data Domain block device hierarchy

Name	Description
Pool	Similar to a 'Department' level. Maximum of 128 pools with DD OS 5.7.1 and above.
Device Group	Similar to the 'Application' level. Maximum of 1024 device groups per pool.
Device	Host device equivalent. Maximum of 2048 block devices for ProtectPoint.

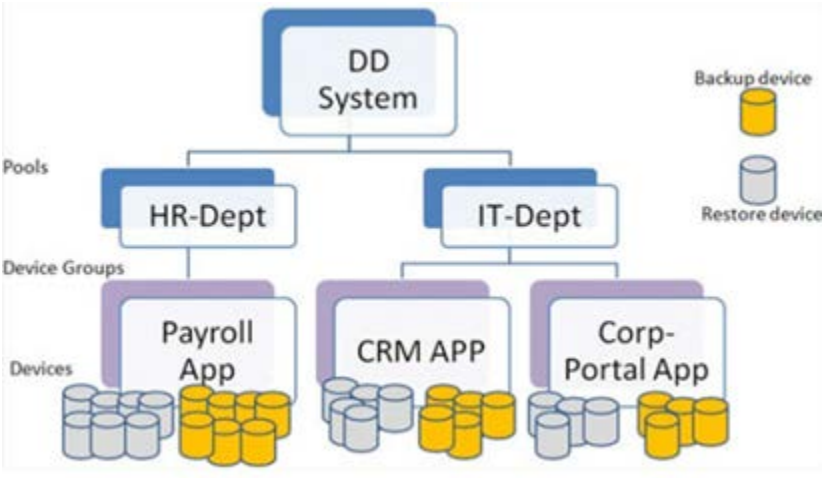

Figure 3: Data Domain block device hierarchy

PROTECTPOINT FILE SYSTEM AGENT

PRODUCT OVERVIEW

Dell EMC ProtectPoint is a software product that takes advantage of the integration between Dell EMC products, VMAX, and Data Domain, to provide a backup offload optimization and automation. The ProtectPoint family consists of the ProtectPoint *File System Agent*, the ProtectPoint *Database Application Agent*, and the ProtectPoint *Microsoft Application Agent*. The following discussion is focused on ProtectPoint File System Agent integration with VMAX for Oracle Database residing on ASM.

ProtectPoint allows Oracle Database backup and restore to take place *entirely* within the integrated systems of VMAX and Data Domain. This capability not only reduces host I/O and CPU overhead, allowing the host to focus on servicing database transactions, but also provides higher efficiency for the backup and recovery process.

Backup efficiencies are introduced by *not requiring any read or write* I/Os of the data files by the host. Instead, TimeFinder SnapVX creates a snapshot which is a valid backup of the database, and then sends it directly to the Data Domain encapsulated *backup* devices. For Oracle Database versions prior to 12c, hot-backup mode is used, though only for the few seconds it takes to create the snapshot – *regardless of the size of the database.* Starting with Oracle Database 12c, hot-backup mode is no longer required when using SnapVX, leveraging a new Oracle feature called Oracle Storage Snapshot Optimization. Although each snapshot provides the current state of the database (that is, a 'Full Backup'), *only storage changes* from the time of the last backup are sent to Data Domain. It is often referred to as full backups at the cost of incremental data transfers. Data Domain then retains the backup data and its metadata in a deduplicated and compressed form, with optional remote replications. The combination of Oracle Database 12c, VMAX, and Data Domain allows the highest backup efficiency.

Restore efficiencies are introduced in a similar way by *not requiring any read* or *write* I/Os of the data files by the host. Instead, Data Domain places the required backup ID's data on its *restore* devices in an internal operation called 'fast-copy'. The *restore* devices are also encapsulated by the VMAX and can be made visible to a Mount host for small-scale data retrievals, or mounted to the Production host and cataloged with RMAN, therefore the RMAN *recover* command can be used to recover the Production database directly.

ProtectPoint 3.1 also offers direct access to the Data Domain *restore* devices even without the use of the encapsulation via VMAX, as well as a shared pool of the *restore* devices that can service restores from different sets of source *backup* devices.

Another benefit of ProtectPoint 3.1 is a rollback feature that automatically restores the backup to Production, overwriting the Production database data files. Rollback is used when Production cannot be recovered and requires complete restore from backup. Also, the rollback feature is efficient, only copying changed data between the Production devices and the encapsulated restore devices with the backup content. As a result, database restore is fast, which means that RTOs are short compared to other solutions.

Note: ProtectPoint rollback requires that the ProtectPoint configuration file specifies a VMAX storage group to identify the encapsulated *restore* devices (instead of Data Domain pool and device group).

HOST AND STORAGE COMPONENTS

When deployed with VMAX and Oracle ASM, ProtectPoint is based on the following components:

- **Production host** (or hosts, in the case of Oracle RAC):
    - Production host connects to native VMAX devices and has no direct visibility to the Data Domain encapsulated backup devices. The Production host may have visibility to the Data Domain encapsulated restore devices if deploying recovery workflow 4b, as described later in Oracle Backup and Recovery Use Cases with ProtectPoint FSA
    - A minimum of three groups of database devices should be defined for maximum flexibility: data files (containing also the control files), redo logs, and FRA (archive logs), each in its own Oracle ASM disk group (for example, +DATA, +REDO, +FRA).
    - If additional ASM disk groups are used containing data files (for example, +UNDO, +DATA2, and so on), they should all be included in operations described in this paper for the +DATA ASM disk group.. Similarly, if more than one redo log ASM disk group is used, then they should all be included in operations described in this paper for the +REDO ASM disk group.

- The separation of data, redo, and archive log files is necessary for backup/recovery solutions based on storage snapshots.
- When Oracle RAC is used it is recommended to use a dedicated ASM disk group for Grid (for example, +GRID) that does not contain any user data. This allows a Mount host or another remote server to already have Grid Infrastructure installed and ASM stack running. During recovery the appropriate ASM disk groups from the backup can simply be mounted by ASM.

- **Management host**
  - The management host is where ProtectPoint software, VMAX Solutions Enabler, and optionally Unisphere for VMAX software are installed. That is the host from which ProtectPoint commands are executed.

  Note: The management host does not need to be the Production host, or the Mount host. It can be any host with access to the VMAX array (via small devices called Gatekeepers). Unlike ProtectPoint *Database Application Agent* which must be installed on the Production host, ProtectPoint *File System Agent* does not communicate directly with the database and can be on any host where Solutions Enabler software is installed, and with connectivity to the VMAX and Data Domain systems.

- **Mount host**
  - **Mount host is optional**. It is used when the DBA prefers to mount the backups not on the Production environment, but rather to a 'sandbox' system where the backup data can be accessed, without interfering with Production operations. In this case the encapsulated *restore* devices are made visible to the Mount host.

  Note: Accessing the Data Domain restore devices directly is only meant for controlled recovery operations, such as limited data extraction, copy, or inspection. For I/O intensive operations, the backup admin should first rollback (copy) the backup to native VMAX devices as Data Domain is not meant to run production-type workloads on its devices.

- **Data Domain system**
  - The Data Domain system uses vdisk service with two identical sets of devices: *backup* devices, and *restore* devices. The *backup* devices are identical to the database production devices in size and quantity.
  - Starting with ProtectPoint 3.1, the *restore* devices can be pooled to allow restore capacity for multiple source systems (sets of *backup* devices). In that case, the pool should have enough available *restore* devices of matching size and capacity as required for the restore operations.
  - The *backup* and *restore* Data Domain devices are created in Data Domain and exposed as VMAX encapsulated devices.
  - Not covered in this paper is a remote Data Domain system. Data Domain Replicator can be used to replicate backups remotely.
- **VMAX storage array**

  VMAX storage array with FAST.X and encapsulated Data Domain *backup* and *restore* device sets.

Note: Refer to the ProtectPoint release notes for details on supported Data Domain systems, host operating systems and more.

PROTECTPOINT FILESYSTEM AGENT CONSIDERATIONS

PROTECTPOINT FILE SYSTEM AGENT AND ORACLE RMAN

HOST-BASED BACKUP AND RECOVERY TIME CHALLENGE

Oracle RMAN is used by many DBAs to perform comprehensive backup, restore, and recovery operations of Oracle Database, as well as to validate the data. Besides performing the backup or recovery operations, RMAN also maintains a catalog with a repository of backups. Natively, RMAN catalogs its own backups. However, RMAN can be pointed to a local file system or ASM disk group containing a database copy and then it will add that copy to its catalog. After it is cataloged, RMAN can use that copy for recovery operations, as will be demonstrated later.

Native RMAN backups run from the database host, where RMAN reads the data from storage, and sends it to the backup target location (over the network, or to disk). While RMAN can use Block Change Tracking (BCT) file to perform incremental backups, and can use multiple links, whenever a full backup or restore is needed, the time it takes has direct dependency on the database size and may create a business challenge. That is where ProtectPoint *Database Application Agent* or *File System Agent* using storage snapshot-based backups can be used.

RMAN PROXY-COPY APIS

RMAN proxy-copy backup is an integration, where RMAN initiates the backup or restore operation, but does not perform it. Instead, it uses the proxy-copy APIs to communicate with the third-party backup manager and the media manager software is responsible for the actual data copy, potentially using storage snapshots such as VMAX SnapVX. An example is ProtectPoint *Database Application Agent*, which is fully integrated with RMAN via the proxy-copy APIs. However, RMAN can use proxy-copy APIs only for databases residing on file systems.

Because RMAN does not support proxy-copy backups with ASM, ProtectPoint *Database Application Agent* cannot be used in this case. Instead, ProtectPoint *File System Agent* (FSA) is the ideal solution. ProtectPoint FSA provides all the advantages of storage snapshots to backup and recovery, such as no dependency on the database size, and very fast backups and restores, and is fully supported with Oracle Database residing on ASM.

PROTECT FILE SYSTEM AGENT AND RMAN

With ProtectPoint FSA the backup and restore operations are initiated by a shell script (and not RMAN). However, as soon as the backup data is restored from Data Domain (possibly in seconds regardless of the database size when mounting the encapsulated *restore* devices to Production), the ASM disk group is renamed, and RMAN can *catalog* it and immediately use it for recovery operations. At that point the DBA can use the breadth of RMAN *recover* command options to perform database recovery procedures directly from RMAN (or SQL if they prefer), such as block corruption recovery, data file recovery, database recovery, and so on.

In summary, ProtectPoint FSA offers a partial integration with RMAN. Both backups and restores are fast, as they use storage snapshots. However, after ProtectPoint restored the data, the DBA can perform database recovery operations using RMAN *recover* command after cataloging the restored data location with RMAN.

PROTECTPOINT FILE SYSTEM AGENT AND RMAN CATALOG

Natively, ProtectPoint uses its own backup catalog, which is stored within Data Domain. However, a few DBAs raised the question whether RMAN catalog can be used as well. Since RMAN does not perform the backup, it is not aware of ProtectPoint backups.

However, it is possible to add the backups to RMAN catalog by adding a few steps to the backup workflow. Specifically, after the backup workflow is executed in full (as described in the next section), it can be quickly mounted to the Mount host (recovery workflow 4a, as shown in the next section) though no actual database restart or recovery is performed. Instead, on the Mount host the ASM +DATA disk group is simply mounted, and then RMAN can catalog its content as a database copy. The RMAN catalog becomes aware of this copy and will list it.

If this step is taken, it is recommended to rename the ASM +DATA disk group to +RESTORED_DATA (or whichever name it will use if performing recovery workflow 4b) before cataloging it, so RMAN will expect to find the data files copy in that disk group name, whether it is on the Mount host or Production. Restore workflow 4b provide a full description of the recovery process and if RMAN has already

cataloged that backup earlier (even if on the Mount host), it does not need to catalog that backup again (even if on Production) – as long as the ASM disk group name matches.

PROTECTPOINT FILE SYSTEM AGENT AND ORACLE RAC

Oracle Real Application Clusters (RAC) offers improved high-availability and load balancing and is very popular, especially on x86_64 and Linux. From storage snapshots perspective, it makes no difference whether the database is clustered or not. The reason is that RAC requires all database files to be visible across all nodes. Therefore, whether the storage snapshots are for a single instance or a clustered database, the replica should always include all data files.

Note that when installing Grid Infrastructure (Oracle 11gR2 and later) the first ASM disk group (for example, +GRID) is created. It is recommended not to place any user data in that disk group. That way, other ASM disk groups with user data can be replicated with storage snapshots, and later can be easily mounted back to the same cluster, or another.

Usually, for VMAX storage, ASM disk groups are created with external redundancy (no ASM mirroring). However, the +GRID ASM disk group is an exception because it is small (since it contains no user data). It can therefore be created with normal redundancy (two mirrors). In this case, ASM automatically creates additional quorum devices, which could be advantageous under high workloads. All other ASM disk groups on VMAX should use external redundancy.

PROTECTPOINT FILE SYSTEM AGENT AND REMOTE REPLICATIONS WITH SRDF

SRDF provides a robust set of remote replication capabilities between VMAX storage arrays, including Synchronous, Asynchronous, three-site (SRDF/STAR), cascaded, Metro (active/active), and more.

ProtectPoint File System Agent and Database Application Agent are both supported with SRDF/S while ProtectPoint File System Agent alone is supported with SRDF/A. Protection of LUNs which is part of the SRDF/Metro relationship is not supported through either agent at this time.

If a ProtectPoint backup is rolled-back (restored back to the original Production devices), the operation is *differential* (only changes between Production devices and the backup are copied). Therefore, SRDF will also only replicate these differential changes to the remote site.

When using SRDF to replicate the Production database remotely, there is no need to replicate the encapsulated *backup* or *restore* devices. The reason is that the encapsulated *backup* devices only contain the latest backup, but not all the prior backups. Those are stored within Data Domain as static-images. The encapsulated *restore* devices only contain a single backup image when it is restored, otherwise they are wiped clean and are empty.

To *execute* the backup operations remotely, consider performing the ProtectPoint backup from an SRDF target. To *replicate* backups taken locally with ProtectPoint to a remote Data Domain system, consider using ProtectPoint replications for Data Domain.

PROTECTPOINT AND SNAPVX

ProtectPoint uses SnapVX technology, but manages it separately from other snapshots that are taken without using ProtectPoint. It uses one snapshot session per backup that it keeps refreshing by creating a new snapshot, then deleting the old one.

The snapshot ProtectPoint uses is first created during ProtectPoint setup steps by using Solutions Enabler. It is not considered a backup and is not required to follow Oracle backup procedures. At that time Solutions Enabler is also used to perform the first *link –copy* that sends the *full copy* of the Production data into the Data Domain system. Although these are considered setup steps, attention should be given so that the initial *link –copy*, being a full copy, does not affect normal database performance. This can be done by performing the setup during low activity window, or by using Solutions Enabler QoS copy settings (for example, `symqos -sg <storage_group_name> set clone pace <0-16>|stop|urgent>`). A pace setting of 4 is recommended for the first full copy from VMAX to Data Domain. To learn more about VMAX QoS see: Dell EMC VMAX3 and VMAX All Flash Quality of Service Controls for Multitenant Environments.

Note: if symqos is used to limit to the copy rate of the initial synchronization between the VMAX devices and Data Domain, *it is important that it is set on the* ***encapsulated backup devices and not on the Production devices***. Setting QoS on the Production devices will slow a Rollback operation (restore) and not the initial synchronization.

Note: with every ProtectPoint *snapshot create* ProtectPoint creates a new snapshot and deletes the previous one. However, during the setup steps, the first snapshot is created *outside* of ProtectPoint. That snapshot remains in the system and consumes both storage capacity and metadata space. After the first ProtectPoint-driven backup it is highly recommended to terminate the original snapshot of the database and FRA that were created during the setup time.

Outside of ProtectPoint, TimeFinder SnapVX can be used to easily and quickly create local database backups, gold copies, test/dev environments, patch tests, reporting instances, and many other use cases. SnapVX snapshots can be taken at any time, and regardless of the size of the database, they are taken within seconds. Restoring snapshots back to their source devices is also very quick. Therefore, as database capacities continue to increase, having a gold copy nearby provides increased availability, protection, and peace of mind. TimeFinder SnapVX replicas can create valid database backup images or database clones. TimeFinder SnapVX is also integrated with SRDF to offload backups to a remote site or restore from a remote site.

For more information about TimeFinder and SRDF best practices for Oracle (without using ProtectPoint) see: Oracle Database Backup and Recovery with VMAX3.

PROTECTPOINT BACKUP ACROSS MULTIPLE VMAX ARRAYS

Most often databases are contained within a single storage array and therefore the backup occurs within a single VMAX array and a single Data Domain system. However, in some cases the database may be spread across multiple VMAX arrays. In that case, ProtectPoint 3.1 supports a topology in which multiple VMAX arrays are connected to the same Data Domain system and the backup operation (SnapVX snapshot) takes place across arrays.

Solutions Enabler and SnapVX technology has supported consistent snapshots across VMAX arrays for many years. ProtectPoint 3.1 can now leverage this capability. It should be noted that ProtectPoint does not require a Solutions Enabler *Composite Group* (CG) to perform the consistent snapshot across arrays. The ProtectPoint configuration file contains a list of Production devices using a combination of VMAX ID and device ID[3]. In the case of multiple arrays this list will contain multiple VMAX IDs and the appropriate VMAX devices. ProtectPoint communicates directly with VMAX Solutions Enabler APIs to perform the snapshot consistently across the arrays.

Another consideration when performing multi-array backup is the ability to perform ProtectPoint rollback. Since rollback requires in the configuration file of the name of a VMAX storage group containing the encapsulated *restore* devices, *in the case of multiple arrays that exact storage group name should exist on all the participating arrays*.

PROTECTPOINT MANAGEMENT AND ORACLE USER PERMISSIONS

Typically, an Oracle operating system (OS) user is used to execute Oracle RMAN or SQL commands, a storage admin OS user is used to perform storage management operations (such as TimeFinder SnapVX, or multipathing commands), and a different OS user may be used to set up and manage the Data Domain system. This type of role and security segregation is common and often helpful in large organizations where each group manages their respective infrastructure with a high level of expertise.

To easily execute the integrated solution described in this white paper, the ability to execute specific commands in Oracle, Solutions Enabler, ProtectPoint, and Data Domain is required. There are two ways to address this:

- Allow the database backup operator controlled access to commands in Solutions Enabler, using VMAX Access Controls (ACLs). Solutions Enabler allows installation for non-root users, and VMAX ACLs allow the storage administrator to permit the backup operator limited control on a set of devices for operations limited to snapshots and monitoring, as required by ProtectPoint.

- Use *sudo*, allowing the backup administrator to execute specific commands for the purpose of their backup (possibly in combination with Access Controls). Alternatively, Solutions Enabler can be installed as a non-root user.

An example for setting up VMAX Access Controls and/or installing Solutions Enabler as a non-root user is provided in Appendix II. In a similar way, Data Domain can create additional user accounts, other than 'sysadmin' that can manage the Data Domain system appropriately. Oracle also allows setting up a backup user and only providing them with a specific set of authorizations appropriate for their task.

[3] The list of device IDs has been replaced in a later ProtectPoint FSA release with a storage group name for simplification.

PROTECTPOINT CONFIGURATION FILE

ProtectPoint relies on a configuration file that contains vital information such as the Data Domain systems information, VMAX Production devices to be backed up, the VMAX storage group name of the encapsulated *restore* devices, and more. Because the list of devices is hard-coded into the configuration file and is critical to the validity of the backup (it must include the correct ASM devices for the different ASM disk groups), it is critical to make sure it is up to date and correct.

Note: This paper was written based on ProtectPoint File System Agent 3.1. It is expected that the next release of ProtectPoint will allow the configuration file to include only the VMAX ID and the storage-group of the devices to back up (instead of a list of the actual devices). This approach simplifies ProtectPoint management and reduces errors. Refer to the latest ProtectPoint documentation set for changes made to the configuration file content and structure.

When changes are made to ASM, such as adding or removing devices to the ASM disk group, the device IDs should also be added or removed from the appropriate Solutions Enabler storage groups, and ProtectPoint configuration file. If devices are added, the ProtectPoint setup steps should be followed for them. It is highly recommended to perform a new backup after making ASM changes. A ProtectPoint restore from an older backup will have the older ASM disk group structure. This is acceptable for logical recovery or when the old backup is cataloged with RMAN. If, however, ProtectPoint *rollback* is used to overwrite Production devices, the ASM changes will have to be re-done.

Since the DBA may use any of the older backups, it is recommended to keep the old ProtectPoint configuration files, renaming them appropriately. Also, enough *restore* devices should be kept to match older backups if changes were made to the ASM disk group device sizes and quantity.

The ProtectPoint 3.1 configuration file contains a number of sections. Refer also to: Example of ProtectPoint **database** configuration file.

- **ProtectPoint Environment**
  - This section contains information, such as the storage array (“VMAX”), and optional fields about the protected application.
- **Primary Data Domain**
  - This section contains connectivity information for the primary Data Domain system.
  - It also contains **critical information about the *restore* devices**. There are two choices in identifying the Data Domain *restore* devices: either by specifying Data Domain *pool* and *device group*, or by specifying a VMAX storage group containing the encapsulated *restore* devices.

Note: ProtectPoint rollback only works when using the VMAX storage group option, containing the encapsulated restore devices. If using multiple VMAX systems the same storage group name should be used on all of them.

```
    # Optional, no default value - The Primary Data Domain
    # pool name containing vdisk devices used for restore
    #
    # By default, restores are performed using FAST.X restore devices which are selected from the VMAX
    # storage group "NsrSnapSG". However, if this and the RESTORE_DEVICE_GROUP fields are specified,
    # then restores are done by selecting restore devices from the specified Data Domain pool and
    # group of restore devices. If either RESTORE_DEVICE_POOL or RESTORE_DEVICE_GROUP are specified,
    # both must be specified and VMAX_FASTX_RESTORE_SG cannot be specified
    # RESTORE_DEVICE_POOL = <Pool name>

    # Optional, no default value - The Primary Data Domain device group used for vdisk restore
    # If either DD_POOL or DD_DEVICE_GROUP are specified, then both must
    # be specified and VMAX_FASTX_RESTORE_SG cannot be specified.
    # RESTORE_DEVICE_GROUP = <Device group name>
...
...
    # Optional, Default = "NsrSnapSG" - the name of the VMAX storage group
    # to use during VMAX restores to select appropriate FAST.X restore devices.
    # If specified, then DD_POOL and DD_DEVICE_GROUP cannot be specified.
    # VMAX_FASTX_RESTORE_SG = <name>
    VMAX_FASTX_RESTORE_SG = rstr_data_sg
```

- In addition, there is a **critical field that should be set if the encapsulated *restore* devices are visible to the ProtectPoint host.**

```
        # Optional, default is false. Indicates whether restore devices to be selected must be visible to the host.
        # SELECT_VISIBLE_RESTORE_DEVICES = TRUE
```

It is critical because if it is set to TRUE, and the restore devices are masked to a different host (or not masked at all) the ProtectPoint restore commands will fail with an error such as:

```
   Error message:
   **** restore prepare failed: No available FAST.X lun was found to perform the
operation…
```

This is especially important if ProtectPoint is run from the Mount host, because at certain times the encapsulated *restore* devices may be masked to that host, and other times they may be masked to the Production host.

- **Secondary Data Domain**

  This section is only relevant if there is a secondary Data Domain system for remote replications.

- **VMAX devices**
  - This is a critical section containing a list of the Production devices that the ProtectPoint configuration file operates on. For example, the **database** configuration file contains a list of both +DATA and +REDO devices, where the **data** or **redo** configuration files only contain their appropriate devices. For example:

```
    # ASM +FRA disk group
    SRC_DEVICE1 = 000196702151:00037
    SRC_DEVICE2 = 000196702151:00038
    SRC_DEVICE3 = 000196702151:00039
    SRC_DEVICE4 = 000196702151:0003A
```

  - This section also contains an optional subset of devices for restore prepare and rollback. For example, in the use cases described in this paper, we want to capture a backup with the **database** devices (both data and redo), and we use all these devices for a *restart* solution. However, if we need a *recovery* solution we often only restore the data devices.

Note: At the time this paper was written and tested with ProtectPoint FSA 3.1, this feature did not work. Instead, we used two different ProtectPoint configuration files: one called **database** which contained both data and redo devices, and another called **data** which only contained the data devices. As described later, we used the appropriate one for the use case.

  - With future releases of ProtectPoint, the configuration file is likely to become simpler and easier to use and maintain. Ensure you read the latest product documentation.

ORACLE BACKUP AND RECOVERY USE CASES WITH PROTECTPOINT FSA

THE BIG PICTURE

This section provides a high level overview of Oracle ASM database backup and recovery use cases with ProtectPoint File System Agent integration, as described in Figure 4. The image includes the workflow steps that are explained with each use case. Following the overview, each use case is described in step-by-step detail.

Note: Remote replications of the primary storage using SRDF, or of the backups using a secondary Data Domain system, are not covered in these use cases, though they can be used as part of the overall solution.

The following use cases are referred to with the workflow numbers shown in Figure 4.

1. Database Backup using ProtectPoint (workflow 1ab and 2).
2. Database RESTART on Mount host (workflow 3, and 4a restart)
3. Database RECOVERY on the Mount host (workflow 3 and 4a recovery)
4. RMAN minor recovery of Production using ProtectPoint backup (workflow 3 and 4b)
5. RMAN recovery of Production after ProtectPoint Rollback (overwriting Production) (workflow 3 and 4c)

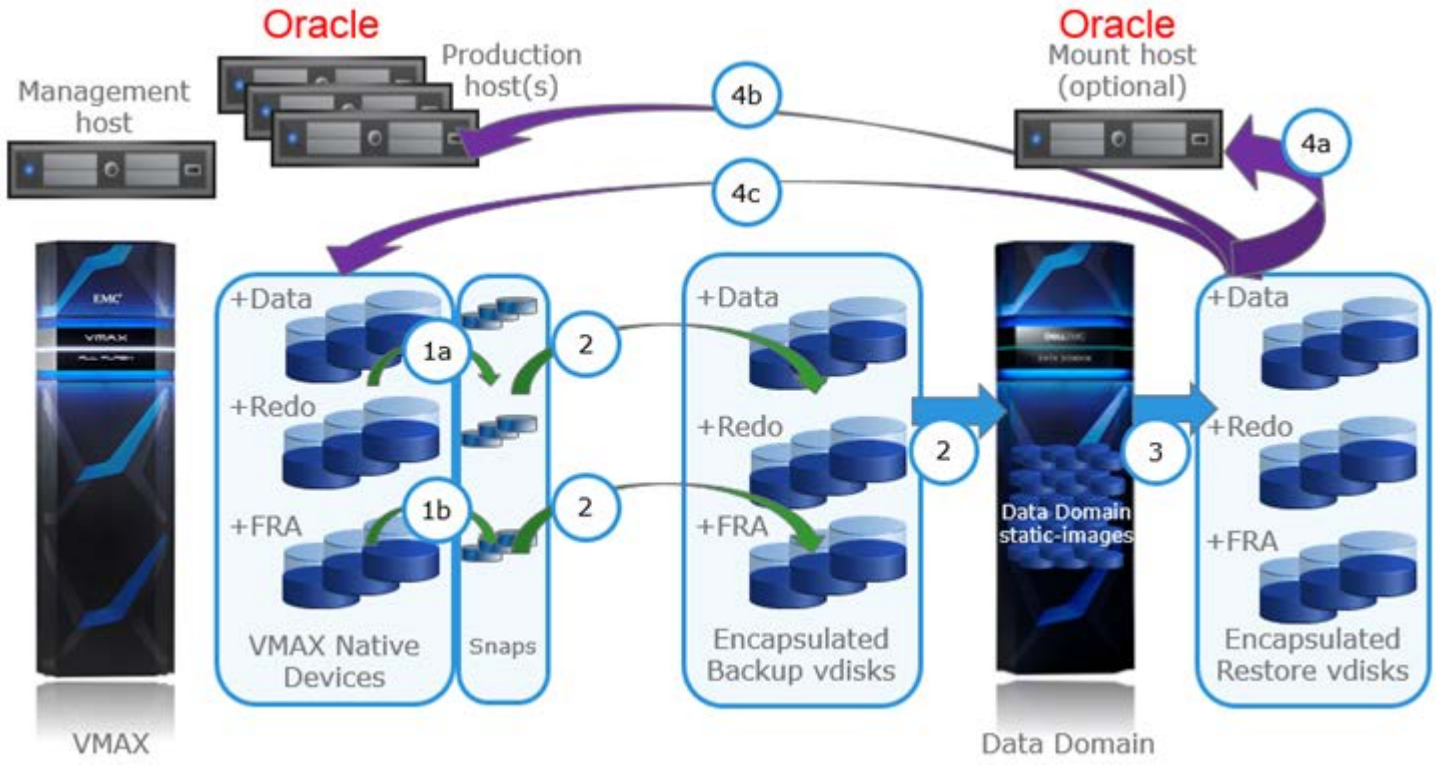

Figure 4 ProtectPoint components and workflow

DATABASE BACKUP USING PROTECTPOINT

1. Begin hot-backup mode (only required for Oracle Database versions prior to 12c)
2. Perform ProtectPoint **snapshot create** using **database** configuration file (workflow 1a).

   ProtectPoint refreshes the storage snapshot that contains both +DATA and +REDO ASM disk groups. This operation takes seconds, regardless of database size.
3. End hot-backup mode (only required for Oracle Database versions prior to 12c)

4. Switch and archive the current logs

   This is a critical step for a database *recovery* solution (though it is not important for a database *restart* solution). The reason is that without this step, during database recovery the data files from the backup will require recovery with transactions that were still in the online redo logs, and those may no longer be available. To make the recovery possible, even when Production logs are unavailable, we need to preserve the minimum required transaction logs. We do this by switching the current logs and archiving them into the FRA, which will be backed-up in the following steps.

5. Perform ProtectPoint **snapshot create** using **fra** configuration file (workflow 1b).

   ProtectPoint refreshes the storage snapshot that contains the +FRA ASM disk group (archive logs).

6. Perform *two* ProtectPoint **backup create**: one using **database** configuration file, the other using **fra** configuration file (workflow 2).

   ProtectPoint executes SnapVX *link -copy* to send incremental changes to the Data Domain encapsulated *backup* devices.

At the end of this process Data Domain adds two backup-IDs: the first with a consistent image of the data, control, and redo log files that can be used for either *restart* or *recovery* solution, and the second with the minimum set of archive logs sufficient to recover the data files (for a minimal *recovery* solution).

DATABASE RESTART ON THE MOUNT HOST

Description: Database *restart* allows the database to perform automatic crash/instance recovery without user intervention and without applying any archive logs. The database opens up to the point-in-time that the snapshot was created, rolling forward any pending committed transaction, and rolling back any uncommitted transactions. To perform a *restart* solution all data, redo, and control files are required to be presented and be consistent (taken with a single snapshot). The restart time depends primarily on the crash-recovery process. Archive logs do not participate in a restart solution.

Purpose: Use this method to access the database's data from the point of time of the backup.

Considerations

- When the encapsulated *restore* devices are visible to the Mount host they can be used directly, saving a huge amount of time in access to the backup, compared to traditional restores or even ProtectPoint rollback.
- An alternative to using the encapsulated *restore* devices is a feature introduced in ProtectPoint 3.1, which allows direct access to Data Domain devices from any host. However, it requires additional zones between the host and Data Domain. Because it is likely that the host is already zoned to the VMAX, it is recommended to use the encapsulated *restore* devices instead of direct access.
- Accessing the Data Domain *restore* devices (either via direct access or via VMAX encapsulation) should be considered carefully, as Data Domain is not meant to run a Production database workload. This method should be used for small-scale database access by the DBA for data inspection, retrieval, or copy.
- Accessing a backup from the Mount host can also be useful prior to a ProtectPoint rollback, if the DBA wants to be sure that they chose the best backup ID prior to a differential copy of the data to Production, which can take a while. Be aware that the *restore* devices are wiped clean after each operation (that is, after the inspection and prior to a rollback) and therefore any database updates performed by the Mount host will not be part of the rollback.

1. Perform *one* ProtectPoint **restore prepare**, using the **database** configuration file and a matching **backup-id** (workflow 3).

   There is no need to restore the FRA backup, because archives are not used in a crash/instance recovery.

   Data Domain uses fast-copy to place the content of the backup-ID (ASM +DATA, +REDO disk groups) on the encapsulated *restore* devices.

2. Add the +DATA and +REDO encapsulated *restore* devices to the Mount host masking view, so they become visible to the host.

   In some environments, the encapsulated *restore* devices may be already masked to the Mount host, so each time after the ProtectPoint *restore prepare* command is executed, the data can be accessed immediately by the Mount host.

3. Mount and open the ASM disk groups on the Mount host (workflow 4a, restart).

4. Open the database by typing: *startup*. Do not perform database media recovery.

5. When finished working with the database, shut it down, dismount the ASM disk groups, and perform ProtectPoint **restore release**.

   ProtectPoint *restore release* is a critical step as ProtectPoint places locks on the encapsulated *restore* devices so no other restore operation can re-use them until the previous one has completed. The *restore release* operation releases these locks. *In addition, it formats the restore devices so prior data is no longer available on them*.

DATABASE RECOVERY ON THE MOUNT HOST

Description: Database *recovery* allows the database to perform media recovery by applying archive logs to the data files. The redo logs from the backup are not used. The database opens up to the point of time that the recovery reached. If only using the data and archives from the backup, this point in time is the same as the backup time. Alternatively, additional archive logs and even the redo logs from Production can be added to the recovery.

Purpose: Use this method for quick access to database from the point of time of the backup, or alternatively roll it farther forward by applying more archive logs from Production or later backups of the FRA.

- **Considerations**: The same considerations as in the Database restart on the Mount host

section.

1. Perform *two* ProtectPoint **restore prepare** operations, one using the **database** configuration file and a matching **backup-ID**, the other using the **fra** configuration file and matching **backup-ID** from the same backup time (workflow 3).

   Use ProtectPoint *backup show list* first to inspect the backups and identify the required backup-IDs.

   During this step Data Domain places the content of the backup-ID (+DATA, +REDO, and +FRA) on the encapsulated *restore* devices.

2. Add the +DATA, +REDO, and +FRA encapsulated *restore* devices to the Mount host masking view, so they become visible to the host.

   In some environments the encapsulated *restore* devices may already be masked to the Mount host, so each time after ProtectPoint restore prepare command is executed, the data can be accessed immediately by the Mount host.

   Note: +REDO ASM disk group is not used during the media recovery. Instead, the redo logs are reset at the end of it. The only reason to present the +REDO disk group is so the database can perform the resetlogs at the end of the recovery and open up. Alternatively, local VMAX devices can be used for that purpose, or if the database is opened read-only the restlogs step is not required.

   ---

3. Mount the three ASM disk groups on the Mount host (workflow 4a, recovery).

4. Copy the backup control file to the control files location using RMAN, as shown in the step-by-step example.

5. Perform minimal database media recovery using the available archive logs in the +FRA.

6. At this point, the database can be opened as read-only, additional archives from Production can be used to roll it even farther back in time, and/or the database can be opened for writes with the *resetlogs* option.

7. When completed, close the database, dismount the ASM disk groups, and perform ProtectPoint **restore release**.

   Refer to notes in the previous section explaining the importance of performing **restore release**.

RMAN MINOR RECOVERY OF PRODUCTION

Description: Only the **+DATA** ASM disk group is used by executing **restore prepare** (and not *rollback*), making the process very fast. Then the encapsulated *restore* devices are made visible to the Production host via device masking. The disk group is renamed (for example, +RESTORED_DATA) to avoid conflict with the original Production +DATA disk group. RMAN catalogs the disk group and can use it to recover the Production database.

Purpose: Use this method to recover the *existing* Production data files. It allows the recovery to start within minutes as the encapsulated *restore* devices are mounted directly to the Production host and not copied first to native VMAX Production devices.

Considerations

This recovery method is best used for small corruptions, such as database block corruptions, a few missing data files, and so on. If the Production host sustained a complete loss of its data files, refer to the next use case describing ProtectPoint *rollback* instead.

1. Perform ProtectPoint **restore prepare** using the **data** configuration file and a **backup-ID**.

   Data Domain places the content of the backup-ID on the encapsulated *restore* devices.

2. Add **only the +DATA** encapsulated *restore* devices to the Production host masking view.

3. Rename the encapsulated ASM disk group to +RESTORED_DATA, mount it to ASM and catalog it with RMAN (workflow 4b).

   Note: If ASMlib is used the Oracle ASM disks will need to be renamed as well.

   After the catalog operation RMAN can use this backup for normal RMAN recovery operations on Production.

4. If RMAN requires missing archive logs, repeat a similar process for older +FRA backups:

   - ProtectPoint **restore prepare** using the **fra** configuration file and a **backup-ID**.
   - Add the +FRA encapsulated *restore* devices to the Production host masking view (only needed first time).
   - Rename the encapsulated +FRA ASM disk group to +RESTORED_FRA, mount it to ASM and use its archive logs.
   - If more than one +FRA backup-ID is required, dismount the +RESTORED_FRA ASM disk group and bring in the next, repeating this step as necessary.

5. When completed, dismount the ASM disk groups that were using the encapsulated *restore* devices, and perform ProtectPoint **restore release**. *Remove the masking views that were used to present them to Production*.

   Refer to the notes in the previous section explaining the importance of performing **restore release**.

RMAN FULL RECOVERY OF PRODUCTION

Description: Protect Point *rollback* is used (and not *restore prepare*). The *rollback* operation places the specified backup ID on the encapsulated *restore* devices, snaps them, and performs a differential *link –copy* of the data back to Production, *overwriting the existing data on Production*. Although it is a differential copy, it still takes some time to complete (based on how far the snapshot data was relative to Production current data). If the DBA is not sure which of the backup IDs to use, a Mount host can be used to inspect the data first as described earlier. At the end of the rollback the Production database is mounted, ready to perform database media recovery using archive and online logs (if available).

Purpose: Use this method if it is clear that the Production database is completely lost. It is better to overwrite its data files with the backup content and roll it forward rather than perform targeted recovery, as described in the previous use case (workflow 4b).

1. Dismount the Production +DATA ASM disk group.

   Note: This is a critical step to avoid any database locks on the Production ASM disk group that may corrupt the data later when the database is trying to mount it and recover. If the data disk group contains the ASM Grid Infrastructure information (as can happen in non-RAC installations), shut down ASM and its services as well (for example, '*crsctl stop has*').

2. Perform ProtectPoint **rollback** using the **data** configuration file and a **backup-ID** (workflow 4c)**.**

Note: Only copy the **data files** back and **not the redo logs**. The reason is that if Production +REDO ASM disk group survived, do not overwrite it with a backup so the current logs can be used in the recovery. When listing the backup IDs, they may have been taken using the **database** ProtectPoint configuration file, but use the **data** configuration file for the rollback.

3. Mount the restored +DATA disk group and perform database media recovery (using RMAN or SQL).

4. If RMAN requires missing archive logs, repeat a similar process from the previous section (workflow 4b) for older +FRA backups.

5. When completed, dismount the ASM disk groups that were using the encapsulated *restore* devices and perform ProtectPoint **restore release**.

While this guide cannot cover all possible backup and recovery scenarios, which vary based on the circumstances and type of failure, it provides an overview and examples of key scenarios that can be used, leveraging ProtectPoint File System Agent.

STEP-BY-STEP WORKFLOW

SETUP

Before starting, prepare a worksheet with all the storage devices ProtectPoint will manage and their usage. If changes are made to the environment over time (such as storage is added to the database) it is important to update the worksheet, and of course ProtectPoint configuration, to support these changes. A sample worksheet is shown in Table 2.

The worksheet starts with the database ASM disk groups, it then shows the VMAX Production device IDs, and the storage groups that contain these devices (though it does not show the parent SG that includes both redo and data SGs). The next section shows the Data Domain backup and restore vdisks and the matching VMAX device IDs and SGs after they are encapsulated.

Table 2 ProtectPoint worksheet with storage information

ASM DG	**Prod**		**DD backup vdisks**				**DD restore vdisks**			
	VMAX Dev	VMAX SG	DDR	WWN (shortened)	VMAX Dev	VMAX SG for backup vdisks	DDR	WWN (shortened)	VMAX Dev	VMAX SG
REDO	033	prod_redo_sg	vdisk-dev0	6002…740001C	041	bkup_redo_sg	vdisk-dev4	6002…7400020	045	rstr_redo_sg
REDO	034	prod_redo_sg	vdisk-dev1	6002…740001D	042	bkup_redo_sg	vdisk-dev5	6002…7400021	046	rstr_redo_sg
REDO	035	prod_redo_sg	vdisk-dev2	6002…740001E	043	bkup_redo_sg	vdisk-dev6	6002…7400022	047	rstr_redo_sg
REDO	036	prod_redo_sg	vdisk-dev3	6002…740001F	044	bkup_redo_sg	vdisk-dev7	6002…7400023	048	rstr_redo_sg
DATA	03B	prod_data_sg	vdisk-dev16	6002…740002C	051	bkup_data_sg	vdisk-dev22	6002…7400032	057	rstr_data_sg
DATA	03C	prod_data_sg	vdisk-dev17	6002…740002D	052	bkup_data_sg	vdisk-dev23	6002…7400033	058	rstr_data_sg
DATA	03D	prod_data_sg	vdisk-dev18	6002…740002E	053	bkup_data_sg	vdisk-dev24	6002…7400034	059	rstr_data_sg
DATA	03E	prod_data_sg	vdisk-dev19	6002…740002F	054	bkup_data_sg	vdisk-dev25	6002…7400035	05A	rstr_data_sg
DATA	03F	prod_data_sg	vdisk-dev20	6002…7400030	055	bkup_data_sg	vdisk-dev26	6002…7400036	05B	rstr_data_sg
DATA	040	prod_data_sg	vdisk-dev21	6002…7400031	056	bkup_data_sg	vdisk-dev27	6002…7400037	05C	rstr_data_sg
FRA	037	prod_fra_sg	vdisk-dev8	6002…7400024	049	bkup_fra_sg	vdisk-dev12	6002…7400028	04D	rstr_fra_dg
FRA	038	prod_fra_sg	vdisk-dev9	6002…7400025	04A	bkup_fra_sg	vdisk-dev13	6002…7400029	04E	rstr_fra_dg
FRA	039	prod_fra_sg	vdisk-dev10	6002…7400026	04B	bkup_fra_sg	vdisk-dev14	6002…740002A	04F	rstr_fra_dg
FRA	03A	prod_fra_sg	vdisk-dev11	6002…7400027	04C	bkup_fra_sg	vdisk-dev15	6002…740002B	050	rstr_fra_dg

1. Perform a system setup as described in Appendix I – ProtectPoint System Setup.

2. At the end of the setup you will have two initial Production snapshots that are linked to the encapsulated *backup* devices. One containing all database data, control, and redo logs. The second containing the archive logs.

3. Three ProtectPoint configuration files are created during setup:

    - One configuration file with devices of +DATA and +REDO ASM disk groups, containing all the database data, control, and redo log files. This configuration file is used for backups that can create a *restartable* recovery solution (workflow 4a, restart). Note that if the data files are spread across multiple ASM disk groups (for example, UNDO, DATA2 or others) *all the devices from these ASM disk groups should be included as well.*

    - One configuration file with devices of just +DATA ASM disk group, containing only data files. This configuration file is used for backups that can create a *recoverable* solution (workflows: 4a recovery, 4b, and 4c). Note that if the data files are spread across multiple ASM disk groups (for example, UNDO, DATA2 or others) *all the devices from these ASM disk groups should be included as well.*

    - One configuration file with all the +FRA ASM disk group devices, containing archive logs.

4. The examples in this white paper use simple Linux shell scripts to simplify execution. Running any of the scripts without parameters will display the required parameters. The content of the scripts is in Appendix III – Scripts. Scripts starting with "se_" use Solutions Enabler commands. Scripts starting with "pp_" use ProtectPoint commands, and scripts starting with "ora_" use Oracle SQL or RMAN commands.
5. When presenting the encapsulated *restore* devices to a host for the first time (Mount, Production, or any other), a rescan of the SCSI bus may be required for the host to recognize them. This can be achieved by a host reboot. However, depending on the operating system, HBA type, or whether ASMlib is used, there are ways to do it online without rebooting. This topic is beyond the scope of this white paper, refer to the HBA and host operating system documentation, or when using ASMlib, use the 'oracleasm scandisks' command. For example, installing sg3_utils Linux kernel package adds: /usr/bin/rescan-scsi-bus.sh command.
6. Unless otherwise specified, all the scripts are executed from the management host (which in this paper is also the Mount host). During backup, even the Oracle scripts are executed from the management host, using Oracle network (Oracle *tnsnames.ora* file) to communicate with the Production host. Make sure that the *tnsnames.ora* file distinguishes clearly between network connections to Production, or a backup mounted on the Mount host.

DATABASE BACKUP USING PROTECTPOINT

1. For Oracle Database **prior to 12c**, place the Production database in hot-backup mode.

```
[root@dsib1136 Scripts]# ./ora_begin_backup.sh
or
SQL> alter database begin backup;
```

2. ProtectPoint **snapshot create** using **database** configuration file (workflow 1a).

   ProtectPoint creates a snapshot of both +DATA and +REDO ASM disk groups.

```
[root@dsib1136 Scripts]# ./pp_snap.sh database "+DATA and +REDO backup"
...
+ protectpoint snapshot create description 'database +DATA and +REDO backup' config-file
/download/ProtectPoint/Configs/PP_database.config
```

3. For Oracle Database prior to 12c, end hot-backup mode.

```
[root@dsib1136 Scripts]# ./ora_end_backup.sh
or
SQL> alter database end backup;
```

4. Switch and archive the Production logs. Also, capture a backup control file.

```
[root@dsib1136 Scripts]# ./ora_switchandarchive.sh
Or
SQL> alter system switch logfile;
SQL> alter system archive log current;
SQL> alter database backup controlfile to '+FRA/CTRL.BCK' reuse;
```

5. ProtectPoint **snapshot create** using **fra** configuration file (workflow 1b).

   ProtectPoint creates a snapshot of the +FRA ASM disk group.

```
[root@dsib1136 Scripts]# ./pp_snap.sh fra "Archive logs backup"
...
+ protectpoint snapshot create description 'fra Archive logs backup' config-file
/download/ProtectPoint/Configs/PP_fra.config
```

6. At the end of this step, we can see the two snapshots using ProtectPoint **backup show list** command. Note the backup ID for each of the snapshots, also the script adds to the description the name of the configuration file that was used.

```
[root@dsib1136 Scripts]# ./pp_backup_show_list.sh database
+ protectpoint backup show list config-file /download/ProtectPoint/Configs/PP_database.config

The catalog query for VMAX backups on the Data Domain is [ALL]
--------------------------------------------------------------------------------
 Backup id   Snapshot time            Duration    Status          Description
--------------------------------------------------------------------------------
 1473637780  Sun Sep 11 19:49:40 2016     N/A      snap-ready      database +DATA and +REDO backup
 1473637850  Sun Sep 11 19:50:50 2016     N/A      snap-ready      fra Archive logs backup
--------------------------------------------------------------------------------
```

7. ProtectPoint **backup create** (workflow 2).

   ProtectPoint pushes the snapshot incremental changes to the Data Domain encapsulated *backup* devices. Perform this operation twice – once with the **database** configuration file and again with the **fra** configuration file.

```
[root@dsib1136 Scripts]# ./pp_backup_create.sh database 1473637780
...
+ protectpoint backup create backup-id 1473637780 config-file
/download/ProtectPoint/Configs/PP_database.config
```

```
[root@dsib1136 Scripts]# ./pp_backup_create.sh fra 1473637850
...
+ protectpoint backup create backup-id 1473637850 config-file
/download/ProtectPoint/Configs/PP_fra.config
```

Note: ProtectPoint **backup create** command only returns the prompt when the incremental copy is done. Until then it displays hashes every few seconds. The scripts *./se_snap_show.sh prod database* and *./se_snap_show.sh prod fra* can be used to monitor the copy progress from another window.

```
[root@dsib1136 Scripts]# ./se_snap_show.sh prod database
...
+ symsnapvx -sg prod_database_sg list -linked -copied -detail -i 30
```

8. To demonstrate the different recovery scenarios a 'test' table was created in the Production database and used to insert records at known times during the backup process. These records will be used during the recovery use cases as a reference to how much recovery was performed.

```
SQL > select to_char(ts, 'YYYY-MM-DD HH24:MM:SS') Time , REC Record from test order by 1;

TIME                RECORD
------------------- --------------------------------
2016-09-11 19:09:00 Before database snapshot
2016-09-11 19:09:05 After database snapshot
2016-09-11 19:09:23 After FRA snapshot
2016-09-11 20:09:12 After PP backup of database
2016-09-11 20:09:59 After PP backup of FRA
```

DATABASE RESTART ON THE MOUNT HOST

1. Perform ProtectPoint **backup list** to choose a backup-id to restore.

```
[root@dsib1136 Scripts]# ./pp_backup_show_list.sh database
+ protectpoint backup show list config-file /download/ProtectPoint/Configs/PP_database.config

The catalog query for VMAX backups on the Data Domain is [ALL]
--------------------------------------------------------------------------------------
 Backup id   Snapshot time             Duration    Status           Description
--------------------------------------------------------------------------------------
 1473637780  Sun Sep 11 19:49:40 2016  000:01:56   complete         database +DATA and +REDO backup
 1473637850  Sun Sep 11 19:50:50 2016  000:00:51   complete         fra Archive logs backup
--------------------------------------------------------------------------------------
```

2. Perform *a single* ProtectPoint **restore prepare** using the **database** configuration file and a **backup-id** (workflow 3).

   Ensure that the ProtectPoint configuration file has the parameter: 'SELECT_VISIBLE_RESTORE_DEVICES = TRUE' set correctly in the Primary Data Domain section (it should be TRUE if the encapsulated *restore* devices are already masked to the same host from which the ProtectPoint commands are executed).

```
[root@dsib1136 Scripts]# ./pp_restore_prepare.sh database 1473637780
+ protectpoint restore prepare backup-id 1473637780 config-file
/download/ProtectPoint/Configs/PP_database.config
...
Updated the catalog record for backup-id [1473637780] from state "complete" to "restore-ready"

[root@dsib1136 Scripts]# ./pp_backup_show_list.sh database
+ protectpoint backup show list config-file /download/ProtectPoint/Configs/PP_database.config
--------------------------------------------------------------------------------------
 Backup id   Snapshot time             Duration    Status           Description
--------------------------------------------------------------------------------------
 1473637780  Sun Sep 11 19:49:40 2016  000:01:56   restore-ready    database +DATA and +REDO backup
 1473637850  Sun Sep 11 19:50:50 2016  000:00:51   complete         fra Archive logs backup
--------------------------------------------------------------------------------------
```

3. It is recommended that a masking view with the VMAX encapsulated *restore* devices is created ahead of time and that the Mount host has scanned for these devices after they were masked (only required the first time). The commands below demonstrate these steps.

```
[root@dsib1136 Scripts]# ./se_mask.sh mount database create
+ symaccess view create -name mount_database_mv -sg rstr_database_sg -ig mount_ig -pg mount_pg

[root@dsib1136 Scripts]# /usr/bin/scsi-rescan (only needed first time devices are presented to host)
```

4. Mount the two ASM disk groups on the Mount host (workflow 4a, restart).

   If RAC is running on the Mount host then it should be already configured and running using a separate ASM disk group (+GRID). However, in the case of a single instance, Oracle High-Availability Services may need to be started first.

```
[root@dsib1136 ~]# su - oracle
[oracle@dsib1136 ~]$ TOGRID
[oracle@dsib1136 ~]$ crsctl start has
CRS-4123: Oracle High Availability Services has been started.
```

   Mount +DATA and +REDO ASM disk groups.

```
[oracle@dsib1136 ~]$ sqlplus "/ as sysasm"
SQL> alter system set asm_diskstring='/dev/mapper/ora*p1';
SQL> alter diskgroup data mount;
SQL> alter diskgroup redo mount;
```

5. Do *not* perform database media recovery. Instead, start the database.

   Note: If the +FRA disk group is not available, consider disabling archive log mode before opening the database, or leaving it in place if archive logs are optional.

```
[oracle@dsib1136 ~]$ TODB
[oracle@dsib1136 ~]$ sqlplus "/ as sysdba"

SQL*Plus: Release 12.1.0.2.0 Production on Mon Sep 12 20:30:11 2016

Copyright (c) 1982, 2014, Oracle.  All rights reserved.

Connected to an idle instance.

SQL> startup
ORACLE instance started.

Total System Global Area 2147483648 bytes
Fixed Size                  2926472 bytes
Variable Size            1744832632 bytes
Database Buffers          385875968 bytes
Redo Buffers               13848576 bytes
Database mounted.
Database opened.
```

6. For reference, review the ‘test’ table. Since the database snapshot included only the first record, and no additional transactions are applied in a restart solution, only that record that was inserted and committed prior to the snapshot and is available in the table.

```
SQL> select to_char(ts, 'YYYY-MM-DD HH24:MM:SS') Time , REC Record from test order by 1;

TIME                RECORD
------------------- --------------------------------
2016-09-11 19:09:00 Before database snapshot

SQL>
```

7. When use of this database backup is no longer needed, close the database and dismount the ASM disk groups. If this is a single instance Oracle and the high availability services were started from the backup diskgroup +DATA, turn it off as well.

```
[oracle@dsib1136 ~]$ TODB
[oracle@dsib1136 ~]$ sqlplus "/ as sysdba"
SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
[oracle@dsib1136 ~]$ TOGRID
[oracle@dsib1136 ~]$ sqlplus "/ as sysasm"
SQL> alter diskgroup data dismount;
```

```
Diskgroup altered.

SQL> alter diskgroup redo dismount;

Diskgroup altered.
SQL> exit
[oracle@dsib1136 ~]$ crsctl stop has       <= if single instance
```

8. After the database is shut down and ASM disk groups are dismounted, the backup can be released. At this point, the device locks on the encapsulated *restore* devices are released, and the devices are formatted by Data Domain from their previous data.

```
[root@dsib1136 Scripts]# ./pp_restore_release.sh database 1473637780
+ protectpoint restore release backup-id 1473637780 config-file
/download/ProtectPoint/Configs/PP_database.config

[root@dsib1136 Scripts]# ./pp_backup_show_list.sh data
+ protectpoint backup show list config-file /download/ProtectPoint/Configs/PP_data.config
-------------------------------------------------------------------------------------
 Backup id   Snapshot time            Duration    Status           Description
-------------------------------------------------------------------------------------
 1473637780  Sun Sep 11 19:49:40 2016  000:01:56   complete         database +DATA and +REDO backup
 1473637850  Sun Sep 11 19:50:50 2016  000:00:51   complete         fra Archive logs backup
-------------------------------------------------------------------------------------
```

DATABASE RECOVERY ON THE MOUNT HOST

1. Perform ProtectPoint **backup list** to choose a backup-id to restore.

```
[root@dsib1136 Scripts]# ./pp_backup_show_list.sh database
+ protectpoint backup show list config-file /download/ProtectPoint/Configs/PP_database.config

The catalog query for VMAX backups on the Data Domain is [ALL]
-------------------------------------------------------------------------------------
 Backup id   Snapshot time            Duration    Status           Description
-------------------------------------------------------------------------------------
 1473637780  Sun Sep 11 19:49:40 2016  000:01:56   complete         database +DATA and +REDO backup
 1473637850  Sun Sep 11 19:50:50 2016  000:00:51   complete         fra Archive logs backup
-------------------------------------------------------------------------------------
```

2. Perform *two* ProtectPoint **restore prepare** operations. One using the **database** configuration file and matching **backup-id**, and the second using the **fra** configuration file and matching backup id (workflow 3).

   Ensure that the ProtectPoint configuration file has the parameter: 'SELECT_VISIBLE_RESTORE_DEVICES = TRUE' set correctly in the Primary Data Domain section (it should be TRUE if the encapsulated *restore* devices are already masked to the same host from which the ProtectPoint commands are executed).

```
[root@dsib1136 Scripts]# ./pp_restore_prepare.sh database 1473637780
+ protectpoint restore prepare backup-id 1473637780 config-file
/download/ProtectPoint/Configs/PP_database.config
...
Updated the catalog record for backup-id [1473637780] from state "complete" to "restore-ready"

[root@dsib1136 Scripts]# ./pp_restore_prepare.sh fra 1473637850
+ protectpoint restore prepare backup-id 1473637850 config-file /download/ProtectPoint/Configs/PP_fra.config

[root@dsib1136 Scripts]# ./pp_backup_show_list.sh database
+ protectpoint backup show list config-file /download/ProtectPoint/Configs/PP_database.config
-------------------------------------------------------------------------------------
 Backup id   Snapshot time            Duration    Status           Description
-------------------------------------------------------------------------------------
 1473637780  Sun Sep 11 19:49:40 2016  000:01:56   restore-ready    database +DATA and +REDO backup
 1473637850  Sun Sep 11 19:50:50 2016  000:00:51   restore-ready    fra Archive logs backup
-------------------------------------------------------------------------------------
```

3. Create two masking views with the VMAX encapsulated *restore* devices. One containing the database devices (data and redo), and the other containing the **fra** devices (archive logs). Rescan the SCSI bus from the Mount host if necessary (only when new devices are presented to the Mount host).

```
[root@dsib1136 Scripts]# ./se_mask.sh mount database create
+ symaccess view create -name mount_database_mv -sg rstr_database_sg -ig mount_ig -pg mount_pg
[root@dsib1136 Scripts]# ./se_mask.sh mount fra create
+ symaccess view create -name mount_fra_mv -sg rstr_fra_sg -ig mount_ig -pg mount_pg

[root@dsib1136 Scripts]# /usr/bin/scsi-rescan (only needed first time devices are presented to host)
```

4. Mount the three ASM disk groups on the Mount host (workflow 4a, recovery).

   If RAC is running on the Mount host then it should be already configured and running using a separate ASM disk group (+GRID). However, in the case of a single instance, Oracle High-Availability Services may need to be started first.

```
[root@dsib1136 ~]# su - oracle
[oracle@dsib1136 ~]$ TOGRID
[oracle@dsib1136 ~]$ crsctl start has
CRS-4123: Oracle High Availability Services has been started.
```

   Mount +DATA and +REDO ASM disk groups.

```
[oracle@dsib1136 ~]$ sqlplus "/ as sysasm"
SQL> alter system set asm_diskstring='/dev/mapper/ora*p1';
SQL> alter diskgroup data mount;
SQL> alter diskgroup redo mount;
SQL> alter diskgroup fra mount;
```

5. Perform minimal database media recovery using the available archive logs in the +FRA, then open the database READ ONLY.

   - In the example, RMAN is used to copy the backup control file to its right place and then SQL is used with automatic media recovery. Alternatively, RMAN can be used for the media recovery (as in use case **4c**).
   - At the end of the recovery, the database is opened as read-only. This is optional, just in case the DBA decides later to apply additional archive logs from Production. Alternatively, the database can be opened as read-write (using the **resetlogs** option).
   - The snapshot time in the recover database command is taken from the time listed for the backup by ProtectPoint.
   - The following example recovers the database using SQL commands, where a later use case (workflow 4c) demonstrates a recovery using RMAN commands.

```
[oracle@dsib1136 ~]$ TODB
[oracle@dsib1136 ~]$ rman
RMAN> connect target /
RMAN> startup nomount;
RMAN> restore controlfile from '+FRA/CTRL.BCK';
RMAN> exit
[oracle@dsib1136 ~]$ sqlplus "/ as sysdba"
SQL> alter database mount;
SQL> recover database until cancel using backup controlfile snapshot time 'SEP-11-2016
19:49:40';
ORA-00279: change 1011536 generated at 09/11/2016 19:01:10 needed for thread 1
ORA-00289: suggestion :
+FRA/SLOB/ARCHIVELOG/2016_09_11/thread_1_seq_5.288.922305143
ORA-00280: change 1011536 for thread 1 is in sequence #5


Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
AUTO
ORA-00279: change 1013117 generated at 09/11/2016 19:52:23 needed for thread 1
ORA-00289: suggestion :
+FRA/SLOB/ARCHIVELOG/2016_09_11/thread_1_seq_6.289.922305143
```

```
ORA-00280: change 1013117 for thread 1 is in sequence #6
ORA-00278: log file
'+FRA/SLOB/ARCHIVELOG/2016_09_11/thread_1_seq_5.288.922305143' no longer needed
for this recovery


ORA-00279: change 1013121 generated at 09/11/2016 19:52:23 needed for thread 1
ORA-00289: suggestion : +FRA
ORA-00280: change 1013121 for thread 1 is in sequence #7
ORA-00278: log file
'+FRA/SLOB/ARCHIVELOG/2016_09_11/thread_1_seq_6.289.922305143' no longer needed
for this recovery


ORA-00308: cannot open archived log '+FRA'
ORA-17503: ksfdopn:2 Failed to open file +FRA
ORA-15045: ASM file name '+FRA' is not in reference form


SQL> alter database open read only;

Database altered.
```

6. For reference, review the 'test' table. Committed transactions have been recovered up to the point of the log switch and FRA backup. This recovery only included the minimum archives required to open the database.

```
[oracle@dsib1136 ~]$ TODB
[oracle@dsib1136 ~]$ sqlplus "/ as sysdba"

SQL> select to_char(ts, 'YYYY-MM-DD HH24:MM:SS') Time , REC Record from test order by 1;

TIME                RECORD
------------------- --------------------------------
2016-09-11 19:09:00 Before database snapshot
2016-09-11 19:09:05 After database snapshot
```

7. When use of this database backup is no longer needed, close the database and dismount the ASM disk groups. If this is a single instance Oracle and the high availability services were started from the backup diskgroup +DATA, turn it off as well.

```
[oracle@dsib1136 ~]$ TODB
[oracle@dsib1136 ~]$ sqlplus "/ as sysdba"
SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
[oracle@dsib1136 ~]$ TOGRID
[oracle@dsib1136 ~]$ sqlplus "/ as sysasm"
SQL> alter diskgroup data dismount;
SQL> alter diskgroup redo dismount;
SQL> alter diskgroup fra dismount;
SQL> exit
[oracle@dsib1136 ~]$ crsctl stop has
```

8. After the database is shut down and ASM disk groups are dismounted, the backup can be released. At this point the device locks on the encapsulated restore devices are released, and the devices are formatted by Data Domain.

```
[root@dsib1136 Scripts]# ./pp_restore_release.sh database 1473637780
+ protectpoint restore release backup-id 1473637780 config-file
/download/ProtectPoint/Configs/PP_database.config

[root@dsib1136 Scripts]# ./pp_restore_release.sh fra 1473637850
+ protectpoint restore release backup-id 1473637850 config-file /download/ProtectPoint/Configs/PP_fra.config

[root@dsib1136 Scripts]# ./pp_backup_show_list.sh data
+ protectpoint backup show list config-file /download/ProtectPoint/Configs/PP_data.config
------------------------------------------------------------------------------------------
 Backup id   Snapshot time             Duration    Status           Description
------------------------------------------------------------------------------------------
 1473637780  Sun Sep 11 19:49:40 2016  000:01:56   complete         database +DATA and +REDO backup
 1473637850  Sun Sep 11 19:50:50 2016  000:00:51   complete         fra Archive logs backup
------------------------------------------------------------------------------------------
```

RMAN MINOR RECOVERY OF PRODUCTION USING PROTECTPOINT BACKUP

1. First, simulate a block corruption to demonstrate this use case[4].
   - Before the corruption, perform a backup (as described in Database backup using ProtectPoint).
   - Introduce a physical block corruption to one of the data files in ASM, and then query the table.

```
SQL> select * from corrupt_test where password='P7777';
select * from corrupt_test where password='P7777'
                                 *
ERROR at line 1:
ORA-01578: ORACLE data block corrupted (file # 5, block # 156)
ORA-01110: data file 5: '+DATA/bad_data_01.dbf'
SQL> quit

[oracle@dsib1141 oracle]$ dbv file='+DATA/bad_data_01.dbf' blocksize=8192

DBVERIFY: Release 12.1.0.2.0 - Production on Fri Sep 16 20:29:43 2016

Copyright (c) 1982, 2014, Oracle and/or its affiliates.  All rights reserved.

DBVERIFY - Verification starting : FILE = +DATA/corrupt_bad_data_01.dbf
...
Total Pages Marked Corrupt   : 1
```

2. There is no ProtectPoint rollback involved in the recovery. Instead, ProtectPoint **restore prepare** alone is used to place the backup ID on the encapsulated **data** restore devices, and those are directly presented to Production and cataloged by RMAN.
3. Perform ProtectPoint **backup list** using either of the configuration files to choose a backup-id to restore.

```
[root@dsib1136 Scripts]# ./pp_backup_show_list.sh data
+ protectpoint backup show list config-file /download/ProtectPoint/Configs/PP_data.config
...
------------------------------------------------------------------------------------------
 Backup id   Snapshot time             Duration    Status           Description
------------------------------------------------------------------------------------------
 1473996921  Thu Sep 15 23:35:21 2016  000:46:07   complete         database +DATA and +REDO before data
corruption
 1473997068  Thu Sep 15 23:37:48 2016  000:00:50   complete         fra Archive logs before data corruption
------------------------------------------------------------------------------------------
```

4. Perform ProtectPoint **restore prepare** using the **data** configuration file (workflow 3).
   - Note that the data devices are a subset of the devices in the database and therefore the database backup can be used to restore just the data devices.

[4] The method to corrupt a database block in ASM is introduced in this blog.

- Ensure that the ProtectPoint configuration file has the parameter: 'SELECT_VISIBLE_RESTORE_DEVICES = TRUE' set correctly in the Primary Data Domain section (it should be TRUE if the encapsulated *restore* devices are already masked to the same host from which the ProtectPoint commands are executed).

```
[root@dsib1136 Scripts]# ./pp_restore_prepare.sh data 1473996921
+ protectpoint restore prepare backup-id 1473996921 config-file
/download/ProtectPoint/Configs/PP_data.config
```

5. Make the **data** encapsulated *restore* devices visible to Production host.

```
[root@dsib1136 Scripts]# ./se_mask.sh prod data create
+ symaccess view create -name prod_rstr_data_mv -sg rstr_data_sg -ig prod_ig -pg prod_pg
...
+ symaccess -sid 000196702151 list view
+ grep 'Symmetrix\|----\|prod_\|mount'
Symmetrix ID          : 000196702151
------------------- ------------------- ------------------- -------------------
mount_gk_mv         mount_ig            mount_pg            mount_gk
prod_db_mv          prod_ig             prod_pg             prod_database_sg
prod_fra_mv         prod_ig             prod_pg             prod_fra_sg
prod_gk_mv          prod_ig             prod_pg             prod_gk
prod_rstr_data_mv   prod_ig             prod_pg             rstr_data_sg
```

6. **Step (4b)**: now that the data encapsulated restore devices are visible to Production host, rename the ASM disk group mounted to Production on the encapsulated restore devices from +DATA to +RESTORED_DATA.

   - Ensure that the **data** encapsulated *restore* devices are visible to the Production host and have Oracle permissions. If necessary, scan the host SCSI bus for new devices. Note that if ASMlib is used the encapsulated *restore* devices will need to be renamed using ASMlib commands (oracleasm renamedisk) prior to the next step of renaming the ASM diskgroup.

   - On Production, rename the encapsulated +DATA ASM disk group to +RESTORED_DATA, and mount it to ASM.

```
[oracle@dsib1141 Scripts]$ cat ora_rename_DATA.txt
/dev/mapper/ora_dd_data1p1 DATA RESTORED_DATA
/dev/mapper/ora_dd_data2p1 DATA RESTORED_DATA
/dev/mapper/ora_dd_data3p1 DATA RESTORED_DATA
/dev/mapper/ora_dd_data4p1 DATA RESTORED_DATA
/dev/mapper/ora_dd_data5p1 DATA RESTORED_DATA
/dev/mapper/ora_dd_data6p1 DATA RESTORED_DATA

[oracle@dsib1141 oracle]$ TOGRID
[oracle@dsib1141 oracle]$ renamedg dgname=DATA newdgname=RESTORED_DATA
config=./ora_rename_DATA.txt asm_diskstring='/dev/mapper/ora_dd*p1'
Parsing parameters..
renamedg operation: dgname=DATA newdgname=RESTORED_DATA config=./ora_rename_DATA.txt
asm_diskstring=/dev/mapper/ora_dd*p1
Executing phase 1
Discovering the group
Checking for heartbeat...
Re-discovering the group
Generating configuration file..
Completed phase 1
Executing phase 2
Completed phase 2

[oracle@dsib1141 oracle]$ sqlplus "/ as sysasm"
SQL> select name, state from v$asm_diskgroup;

NAME                           STATE
------------------------------ -----------
REDO                           MOUNTED
RESTORED_DATA                  DISMOUNTED
FRA                            MOUNTED
DATA                           MOUNTED
SQL> alter diskgroup restored_data mount;
Diskgroup altered.
```

7. Catalog the +RESTORED_DATA ASM disk group on the encapsulated *restore* devices with RMAN. Then use it for RMAN recovery. In this case we recover the block corruption. If additional backups of +DATA are needed, unmount the +RESTORED_DATA disk group and repeat the process with another backup-ID of +DATA.

```
[oracle@dsib1141 Scripts]$ TODB
[oracle@dsib1141 Scripts]$ rman "target=/"
RMAN> connect catalog rco/oracle@catdb       <- Optional, only if RMAN catalog is available.
RMAN> catalog start with '+RESTORED_DATA/SLOB/DATAFILE' noprompt;
...
cataloging files...
cataloging done
```

8. We already know where the corruption is from the 'dbv' command earlier. However, **optionally** check for farther corruptions in the database. This step can take a long time, based on the size of the database.

   First, find where is the corruption by scanning the database and then checking the trace log.

```
RMAN> validate check logical database;
Or
RMAN> validate datafile 5;
```

9. Perform RMAN recovery based on the situation (in this case – physical block corruption of data file 5 block 156, as seen by dbv output above).

```
RMAN> alter tablespace bad_data offline immediate;
RMAN> recover datafile 5 block 156;
...
channel ORA_DISK_1: restoring block(s) from datafile copy
+RESTORED_DATA/SLOB/DATAFILE/bad_data.262.922663881

starting media recovery
media recovery complete, elapsed time: 00:00:02

RMAN> alter tablespace bad_data online;
RMAN> quit;


[oracle@dsib1141 Scripts]$ dbv file='+DATA/bad_data_01.dbf' blocksize=8192

DBVERIFY: Release 12.1.0.2.0 - Production on Sat Sep 17 09:08:51 2016

Copyright (c) 1982, 2014, Oracle and/or its affiliates.  All rights reserved.

DBVERIFY - Verification starting : FILE = +DATA/bad_data_01.dbf


DBVERIFY - Verification complete

Total Pages Examined         : 1280
Total Pages Processed (Data) : 28
Total Pages Failing   (Data) : 0
Total Pages Processed (Index): 0
Total Pages Failing   (Index): 0
Total Pages Processed (Other): 131
Total Pages Processed (Seg)  : 0
Total Pages Failing   (Seg)  : 0
Total Pages Empty            : 1121
Total Pages Marked Corrupt   : 0
Total Pages Influx           : 0
Total Pages Encrypted        : 0
Highest block SCN            : 0 (0.0)
```

10. If RMAN requires missing archive logs during the recovery (if performing a different type of RMAN recovery than block corruption), choose an appropriate +FRA backup-ID from ProtectPoint and mount the +FRA encapsulated *restore* devices to production, renaming the disk group to +RESTORED_FRA. Do not use rollback. Use the archive logs in +RESTORED_FRA as necessary and when completed unmount the disk group and release the restore. These steps are documented in full at the end of the next use case.

RMAN RECOVERY OF PRODUCTION AFTER PROTECTPOINT ROLLBACK, OVERWRITING PRODUCTION DATA DEVICES

As discussed earlier, even if the Mount host was used to inspect different backups directly (before a rollback) using the encapsulated *restore* devices, any changes to the *restore* devices will be lost after the backup is released and they are wiped clean. The rollback operation will refresh the backup content on the *restore* devices prior to copying over to Production.

As explained earlier, to prevent wiping the online redo logs on Production, *only the ProtectPoint* **data** *configuration file will be used, and not the* **database** *configuration file* (which contains both data and redo devices). If the redo logs on Production are no longer available, it is still recommended to recreate the ASM disk group on the native Production VMAX devices and let the 'resetlogs' step recreate them.

1. Perform ProtectPoint **backup list** to choose a backup-id to restore.

```
[root@dsib1136 Scripts]# ./pp_backup_show_list.sh data
+ protectpoint backup show list config-file /download/ProtectPoint/Configs/PP_data.config

The catalog query for VMAX backups on the Data Domain is [ALL]
-------------------------------------------------------------------------------
 Backup id   Snapshot time             Duration    Status           Description
-------------------------------------------------------------------------------
 1473637780  Sun Sep 11 19:49:40 2016  000:01:56   complete         database +DATA and +REDO backup
 1473637850  Sun Sep 11 19:50:50 2016  000:00:51   complete         fra Archive logs backup
-------------------------------------------------------------------------------
```

2. Shut down Production database and dismount the ASM disk group that will be restored on Production host prior to the rollback.

   It is a very important step to ensure no locks by the database or ASM remain on the previously mounted +DATA ASM disk group.

On Production

```
[oracle@dsib1141 ~]$ TODB
[oracle@dsib1141 ~]$ sqlplus "/ as sysdba"
SQL> shutdown immediate;
SQL> exit
[oracle@dsib1141 ~]$ TOGRID
[oracle@dsib1141 ~]$ sqlplus "/ as sysasm"
SQL> alter diskgroup data dismount;
SQL> exit

Or (if single instance and Grid is running from DATA):

[oracle@dsib1141 ~]$ TOGRID
[oracle@dsib1141 ~]$ crsctl stop has
```

1. Perform ProtectPoint **restore rollback** using the **data** configuration file and matching **backup-id** (workflow 3).

   Ensure that the ProtectPoint configuration file has the parameter: 'SELECT_VISIBLE_RESTORE_DEVICES = TRUE' set correctly in the Primary Data Domain section (it should be TRUE if the encapsulated *restore* devices are already masked to the same host from which the ProtectPoint commands are executed).

```
[root@dsib1136 Scripts]# ./pp_restore_rollback.sh data 1473637780
...
+ protectpoint rollback backup-id 1473637780 config-file /download/ProtectPoint/Configs/PP_data.config
*** Using VMAX config file "/download/ProtectPoint/Configs/PP_data.config" ***
Rollback will overwrite all the contents of the following source devices:
000196702151:0003B
000196702151:0003C
000196702151:0003D
000196702151:0003E
000196702151:0003F
000196702151:00040
You must unmount the file systems containing the above source devices before proceeding with the rollback.
Are you sure you would like to continue (Default is 'no') [yes/no]?
yes
Performing rollback of backup "1473637780"
This command may take a long time to complete...
Updated the catalog record for backup-id [1473637780] from state "complete" to "rollback"
```

2. Monitoring the progress of the rollback can be done from another terminal window. It may take a few minutes for the snapshot session to start, as ProtectPoint first needs to copy the backup to the restore devices.

```
[root@dsib1136 Scripts]# ./se_snap_show.sh rstr data
+ symsnapvx -sg rstr_data_sg list -linked -copied -detail -i 30
```

3. *Only if the rollback process was terminated prematurely* use the rollback **reset** option to bring the backup status back to 'complete'. In addition, ProtectPoint will leave lock #9 on the production and restore **data** devices that should be cleared manually. Finally, the SnapVX session can be unlinked and terminated.

```
[root@dsib1136 Scripts]# protectpoint rollback backup-id 1473637780 config-file
/download/ProtectPoint/Configs/PP_data.config reset

[root@dsib1136 Scripts]# symsg show rstr_data_sg
[root@dsib1136 Scripts]# symsg show prod_data_sg
[root@dsib1136 Scripts]# symdev release -lock 9 -devs 57:5c
[root@dsib1136 Scripts]# symdev release -lock 9 -devs 3b:40
[root@dsib1136 Scripts]# ./se_snap_unlink.sh rstr data PROTECTPOINT
[root@dsib1136 Scripts]# ./se_snap_terminate.sh rstr data PROTECTPOINT
```

4. Mount the ASM disk groups on Production host (workflow 4c).

   If RAC is running on the Production host then it should be configured already and running using a separate ASM disk group (+GRID). In that case, just mount the +DATA ASM disk group that was restored. However, in the case of a single instance, Oracle High-Availability Services may need to be started first.

```
[oracle@dsib1141 ~]$ TOGRID
[oracle@dsib1141 ~]$ crsctl start has
CRS-4123: Oracle High Availability Services has been started.

Or:

[oracle@dsib1136 ~]$ sqlplus "/ as sysasm"
SQL> alter diskgroup data mount;

Then:

SQL> select name, state from v$asm_diskgroup;

NAME                           STATE
------------------------------ -----------
REDO                           MOUNTED
FRA                            MOUNTED
DATA                           MOUNTED
```

5. Perform database media recovery using the available archive logs in Production, bringing any missing archive logs from backup.

   - In the example, RMAN is used to copy the backup control file to its right place and to perform the recovery. Alternatively, SQL can be used for the media recovery (as in use case **4a, recovery**).

   - **Note:** The snapshot time in the recover database command is taken from the time listed for the backup by ProtectPoint.

```
[oracle@dsib1141 ~]$ TODB
[oracle@dsib1141 ~]$ rman

Recovery Manager: Release 12.1.0.2.0 - Production on Mon Mar 30 09:42:29 2015

Copyright (c) 1982, 2014, Oracle and/or its affiliates.  All rights reserved.

RMAN> connect target /

connected to target database (not started)
RMAN> startup nomount;
RMAN> restore controlfile from '+FRA/CTRL.BCK';
RMAN> alter database mount;
RMAN> recover database until time "TO_DATE('09/11/16 19:50:00','mm/dd/yy hh24:mi:ss')"
snapshot time "TO_DATE('09/11/16 19:49:40','mm/dd/yy hh24:mi:ss')";
```

```
Starting recover at 14-SEP-16
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=1250 device type=DISK

starting media recovery

archived log for thread 1 with sequence 5 is already on disk as file
+REDO/SLOB/ONLINELOG/group_1.256.922031897
archived log for thread 1 with sequence 6 is already on disk as file
+REDO/SLOB/ONLINELOG/group_2.257.922031905
archived log for thread 1 with sequence 7 is already on disk as file
+REDO/SLOB/ONLINELOG/group_3.258.922031913
archived log file name=+REDO/SLOB/ONLINELOG/group_1.256.922031897 thread=1 sequence=5
archived log file name=+REDO/SLOB/ONLINELOG/group_2.257.922031905 thread=1 sequence=6
archived log file name=+REDO/SLOB/ONLINELOG/group_3.258.922031913 thread=1 sequence=7
media recovery complete, elapsed time: 00:00:07
Finished recover at 14-SEP-16
RMAN> alter database open resetlogs;

Statement processed
```

Note: There is no need to use ProtectPoint restore release after a rollback. The backup is already in 'complete' state.

```
[root@dsib1136 Scripts]# ./pp_backup_show_list.sh data
+ protectpoint backup show list config-file /download/ProtectPoint/Configs/PP_data.config

The catalog query for VMAX backups on the Data Domain is [ALL]
--------------------------------------------------------------------------------
 Backup id   Snapshot time             Duration    Status           Description
--------------------------------------------------------------------------------
 1473637780  Sun Sep 11 19:49:40 2016  000:01:56   complete         database +DATA and +REDO backup
 1473637850  Sun Sep 11 19:50:50 2016  000:00:51   complete         fra Archive logs backup
--------------------------------------------------------------------------------
```

Optional: If RMAN requires missing archive logs during the recovery, choose an appropriate +FRA backup from Data Domain and mount the +FRA encapsulated *restore* devices to the Production host, renaming the disk group to +RESTORED_FRA. Use the archives, and if more are necessary, unmount this disk group and repeat the process.

Note: Do not use ProtectPoint rollback to not over-write the existing archive logs. Instead, the encapsulated *restore* devices are masked to the Production, renamed, and RMAN will catalog this disk group.

1. ProtectPoint **backup restore prepare** using the **fra** configuration file and a **backup-id**.

```
[root@dsib1136 Scripts]# ./pp_backup_show_list.sh data
+ protectpoint backup show list config-file /download/ProtectPoint/Configs/PP_data.config

--------------------------------------------------------------------------------
 Backup id   Snapshot time             Duration    Status           Description
--------------------------------------------------------------------------------
 1473637780  Sun Sep 11 19:49:40 2016  000:01:56   complete         database +DATA and +REDO backup
 1473637850  Sun Sep 11 19:50:50 2016  000:00:51   complete         fra Archive logs backup
--------------------------------------------------------------------------------

[root@dsib1136 Scripts]# ./pp_restore_prepare.sh fra 1473637850
+ protectpoint restore prepare backup-id 1473637850 config-file /download/ProtectPoint/Configs/PP_fra.config
```

2. Add the +FRA encapsulated *restore* devices to the Production host masking view (remove them first from the Mount host masking view if they were associated with its masking view)

```
[root@dsib1136 Scripts]# ./se_mask.sh mount fra list

+ symaccess -sid 000196702151 list view
+ grep 'Symmetrix\|----\|prod_\|mount'
Symmetrix ID          : 000196702151
------------------ ------------------ ------------------ ------------------
mount_database_mv   mount_ig           mount_pg           rstr_database_sg
mount_fra_mv        mount_ig           mount_pg           rstr_fra_sg
mount_gk_mv         mount_ig           mount_pg           mount_gk
prod_db_mv          prod_ig            prod_pg            prod_database_sg
prod_fra_mv         prod_ig            prod_pg            prod_fra_sg
prod_gk_mv          prod_ig            prod_pg            prod_gk

[root@dsib1136 Scripts]# ./se_mask.sh mount fra delete
[root@dsib1136 Scripts]# ./se_mask.sh prod fra create
+ symaccess view create -name prod_rstr_fra_mv -sg rstr_fra_sg -ig prod_ig -pg prod_pg
+ symaccess -sid 000196702151 list view
+ grep 'Symmetrix\|----\|prod_\|mount'
Symmetrix ID          : 000196702151
------------------ ------------------ ------------------ ------------------
mount_database_mv   mount_ig           mount_pg           rstr_database_sg
mount_gk_mv         mount_ig           mount_pg           mount_gk
prod_db_mv          prod_ig            prod_pg            prod_database_sg
prod_fra_mv         prod_ig            prod_pg            prod_fra_sg
prod_gk_mv          prod_ig            prod_pg            prod_gk
prod_rstr_fra_mv    prod_ig            prod_pg            rstr_fra_sg
```

3. On Production, rescan the SCSI bus and, if using DM-multipath, restart the service.

```
[root@dsib1141 scripts]# ./os_rescan.sh
[root@dsib1141 scripts]# service multipathd restart

[root@dsib1141 scripts]# multipath -ll | grep dd_fra
ora_dd_fra4 (360000970000196702151533030303530) dm-16 EMC     ,SYMMETRIX
ora_dd_fra3 (360000970000196702151533030303446) dm-12 EMC     ,SYMMETRIX
ora_dd_fra2 (360000970000196702151533030303445) dm-21 EMC     ,SYMMETRIX
ora_dd_fra1 (360000970000196702151533030303444) dm-13 EMC     ,SYMMETRIX
```

4. Rename the encapsulated +FRA ASM disk group to +RESTORED_FRA, mount it to ASM, and use its archive logs.

```
[oracle@dsib1141 oracle]$ cat ora_rename_FRA.txt
/dev/mapper/ora_dd_fra1p1 FRA RESTORED_FRA
/dev/mapper/ora_dd_fra2p1 FRA RESTORED_FRA
/dev/mapper/ora_dd_fra3p1 FRA RESTORED_FRA
/dev/mapper/ora_dd_fra4p1 FRA RESTORED_FRA

[oracle@dsib1141 oracle]$ TOGRID
[oracle@dsib1141 oracle]$ renamedg dgname=FRA newdgname=RESTORED_FRA config=./ora_rename_FRA.txt
verbose=yes asm_diskstring='/dev/mapper/ora_dd*p1'
...
renamedg operation: dgname=FRA newdgname=RESTORED_FRA config=./ora_rename_FRA.txt verbose=yes
asm_diskstring=/dev/mapper/ora_dd*p1
Executing phase 1
...
Completed phase 1
Executing phase 2
...
Completed phase 2
[oracle@dsib1141 oracle]$ sqlplus "/ as sysasm"
SQL> select name, state from v$asm_diskgroup;

NAME                           STATE
------------------------------ -----------
REDO                           MOUNTED
FRA                            MOUNTED
DATA                           MOUNTED
RESTORED_FRA                   DISMOUNTED
```

```
SQL> alter diskgroup RESTORED_FRA mount;

Diskgroup altered.

[oracle@dsib1141 Scripts]$ TODB
[oracle@dsib1141 Scripts]$ rman
RMAN> connect target /
RMAN> catalog start with '+RESTORED_FRA/SLOB/ARCHIVELOG' noprompt;
```

5. When done, dismount the +RESTORE_FRA ASM disk group and release the ProtectPoint restored FRA backup ID. Repeat the process as necessary with other backups of the FRA disk group. When no more archive logs from backups are required for Production remove the masking view of the restored devices from Production.

```
From Production as Grid user:
SQL> alter diskgroup RESTORED_FRA dimount;

From management host:
[root@dsib1136 Scripts]# ./pp_restore_release.sh fra 1473637850
[root@dsib1136 Scripts]# ./se_mask.sh prod fra delete
+ symaccess view delete -name prod_rstr_fra_mv
```

CONCLUSION

ProtectPoint File System Agent (FSA) offers a solution to the growing challenge of maintaining backup and recovery SLAs, even in the face of growing database capacities and increased workload. With ProtectPoint FSA the backup time is no longer dependent on the size of the database.

Furthermore, with ProtectPoint FSA the management of the backup and recovery solution can be performed by a DBA or backup administrator, including execution of the required ProtectPoint, Solutions Enabler, and database commands. This can be done by using VMAX Access Controls to limit the scope of Solutions Enabler to manage snapshot and monitoring of the appropriate set of storage devices, and in parallel setting up Solutions Enabler for non-root user.

As demonstrated in this paper, while high-level of backup efficiency and performance is achieved when using ProtectPoint FSA, RMAN can still be used to catalog the backups, or during recovery.

APPENDIXES

APPENDIX I – PROTECTPOINT SYSTEM SETUP

SETUP STEPS OVERVIEW

To prepare the system for ProtectPoint, complete the following steps:

1. Set up physical connectivity
2. Set up Management host
3. Set up Production host
4. Set up Mount host (optional)
5. Set up Data Domain system
6. Set up encapsulated vdisks
7. Set up initial SnapVX sessions
8. Set up ProtectPoint software

As described earlier, it is critical to set up a table from the beginning describing the ProtectPoint environment, and keep updating it with any changes. A sample is presented in Table 3 below, containing the configuration used in this paper. However, it should be noted that the table does not contain the parent SG information (for example, prod_database_sg contained both prod_redo_sg, and prod_data_sg).

Table 3 Devices and SG configuration

ASM DG	Prod		DD backup vdisks				DD restore vdisks			
	Dev	SG	Dev	WWN (shortened)	SG	DDR	Dev	WWN (shortened)	SG	DDR
REDO	033	prod_redo_sg	041	6002…740001C	bkup_redo_sg	vdisk-dev0	045	6002…7400020	rstr_redo_sg	vdisk-dev4
REDO	034	prod_redo_sg	042	6002…740001D	bkup_redo_sg	vdisk-dev1	046	6002…7400021	rstr_redo_sg	vdisk-dev5
REDO	035	prod_redo_sg	043	6002…740001E	bkup_redo_sg	vdisk-dev2	047	6002…7400022	rstr_redo_sg	vdisk-dev6
REDO	036	prod_redo_sg	044	6002…740001F	bkup_redo_sg	vdisk-dev3	048	6002…7400023	rstr_redo_sg	vdisk-dev7
DATA	03B	prod_data_sg	051	6002…740002C	bkup_data_sg	vdisk-dev16	057	6002…7400032	rstr_data_sg	vdisk-dev22
DATA	03C	prod_data_sg	052	6002…740002D	bkup_data_sg	vdisk-dev17	058	6002…7400033	rstr_data_sg	vdisk-dev23
DATA	03D	prod_data_sg	053	6002…740002E	bkup_data_sg	vdisk-dev18	059	6002…7400034	rstr_data_sg	vdisk-dev24
DATA	03E	prod_data_sg	054	6002…740002F	bkup_data_sg	vdisk-dev19	05A	6002…7400035	rstr_data_sg	vdisk-dev25
DATA	03F	prod_data_sg	055	6002…7400030	bkup_data_sg	vdisk-dev20	05B	6002…7400036	rstr_data_sg	vdisk-dev26
DATA	040	prod_data_sg	056	6002…7400031	bkup_data_sg	vdisk-dev21	05C	6002…7400037	rstr_data_sg	vdisk-dev27
FRA	037	prod_fra_sg	049	6002…7400024	bkup_fra_sg	vdisk-dev8	04D	6002…7400028	rstr_fra_dg	vdisk-dev12
FRA	038	prod_fra_sg	04A	6002…7400025	bkup_fra_sg	vdisk-dev9	04E	6002…7400029	rstr_fra_dg	vdisk-dev13
FRA	039	prod_fra_sg	04B	6002…7400026	bkup_fra_sg	vdisk-dev10	04F	6002…740002A	rstr_fra_dg	vdisk-dev14
FRA	03A	prod_fra_sg	04C	6002…7400027	bkup_fra_sg	vdisk-dev11	050	6002…740002B	rstr_fra_dg	vdisk-dev15

SET UP PHYSICAL CONNECTIVITY

The assumption is that the physical system connectivity was done as part of system installation by Dell EMC personnel. Make sure that:

- SAN connectivity exists between switch(s) and:
    - Data Domain
    - VMAX
    - Management host (could be any host with ProtectPoint FSA and Solutions Enabler installed)
    - Production host
    - Mount host (optional)
- SAN zones are created for:
    - Data Domain FC ports with VMAX FAST.X DX ports
    - Management host and VMAX front-end ports
    - Production host and VMAX front-end ports
    - Mount host (optional) and VMAX front-end ports

- Follow the ProtectPoint File System Agent Installation and Administration Guide for discussion about connectivity between VMAX and Data Domain. Figure 5 shows an example from the guide.

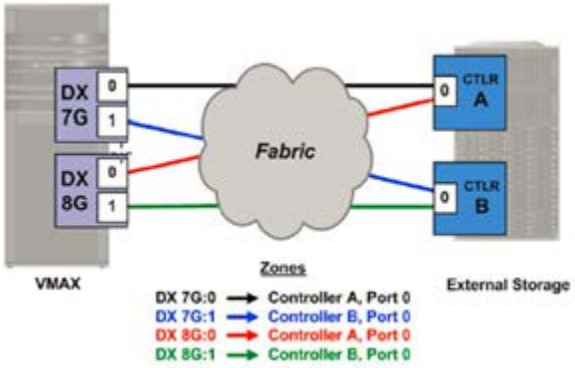

Figure 5 VMAX and Data Domain SAN connectivity

SET UP MANAGEMENT HOST SOFTWARE AND MASKING VIEWS

The Management host is where ProtectPoint commands are executed and monitored. This host is likely to be a separate host from the Production host(s), but it could be the same as the Mount host (such as in the case described in this paper).

In this paper, the hosts used for demonstration are: 'dsib1141' used as the Production host, and 'dsib1136' used as the Management and Mount host.

Perform the following operations to set up the Management host (or Mount host, if they are the same):

1. Install Solutions Enabler (SE) CLI software.
2. If VMAX Access Controls (ACL) are to be used to limit the set of devices and operations that the Management host can perform, ensure that the Dell EMC personnel also initialize the ACL database in the VMAX so the first ACL management host and users can be added.
3. Post SE installation:
    - Update the path to include SE binaries (for example, **`export PATH=$PATH:/usr/symcli/bin`**).
    - If a single VMAX is managed, add its ID to the environment variable so it will not be needed during SE CLI execution. (for example, **`export SYMCLI_SID=000196702151`**)
4. Create a masking view for the management host with just gatekeepers.
    - Alternatively, if the Mount host is the same as the Management host, refer to the Mount host setup section.
    - The gatekeepers (GKs) are in their own SG and masking view as the Management host only requires GKs to communicate with the VMAX storage. Even when the Management and Mount host are the same, it is not recommended to mix GKs devices with SGs used for database devices.

```
#!/bin/bash
# To find HBA port WWNs run the following command:
# cat /sys/class/fc_host/host?/port_name
set -x
export SYMCLI_SID=000196702151
symaccess -type storage   -name mgmt_gk create devs 13:17 # gatekeeprs for management host
symaccess -type initiator -name mgmt_ig create
symaccess -type initiator -name mgmt_ig add -wwn <hba1_port1_wwn>
symaccess -type initiator -name mgmt_ig add -wwn <hba2_port1_wwn>
symaccess -type port      -name mgmt_pg create -dirport 1D:8,2D:8
symaccess create view     -name mgmt_mv –sg mgmt_gk –ig mgmt_ig –pg mgmt_pg
```

5. Install Unisphere for VMAX (optional). Unisphere was not used in the tests and operations described in this paper.

6. Refresh the SE database: **`symcfg discover`**. Then list the available storage devices: **`symdev list -gb`**.

7. If there are available gatekeepers they can be used. Otherwise, create additional small communication devices. (for example, **`symconfigure -sid 151 -cmd "create gatekeeper count=8 ;" commit)`**

8. Update DNS or /etc/hosts with references to Production host, Mount host, and Data Domain. For example, if using /etc/hosts:

```
[root@dsib1141 scripts]# cat /etc/hosts
10.108.245.141  dsib1141.lss.emc.com    dsib1141        Prod
10.108.245.136  dsib1136.lss.emc.com    dsib1136        Mount/Management
10.108.244.18   dsib0018.lss.emc.com    dsib0018        DDS
```

SET UP PRODUCTION HOST

Production database devices may already exist. If not, they can be created using Solutions Enabler (SE) CLI or Unisphere. Note that although the devices are for the Production database, the Solutions Enabler commands are executed from the Management host.

Note: If Grid Infrastructure is used (Oracle RAC), these devices do not need to be backed up as they do not contain any user data.

The following example creates via SE:

- 4 x 10 GB devices for +REDO ASM disk group
- 6 x 200 GB devices for +DATA ASM disk group
- 4 x 100 GB for +FRA ASM disk group.

```
symconfigure -sid 151 -cmd "create dev count=4,size=10  GB, emulation=FBA, config=tdev ;" commit
symconfigure -sid 151 -cmd "create dev count=4,size=100 GB, emulation=FBA, config=tdev ;" commit
symconfigure -sid 151 -cmd "create dev count=6,size=200 GB, emulation=FBA, config=tdev ;" commit

Or
symdev create –tdev –cap 10 –captype gb –N 4
symdev create –tdev –cap 100 –captype gb –N 4
symdev create –tdev –cap 200 –captype gb –N 6
```

Create **a masking view** for the Production host.

Note: Each ASM disk group type (+DATA, +REDO, and +FRA) gets its own storage-group (SG).

This example also creates a cascaded SG that includes *ALL the database devices* for: data, control, and log files (but not archive logs/FRA). This cascaded SG (called *prod_database_sg*) is used to create a *storage consistent* replica of the database for restart solutions when a Mount host is used.

Note: For RAC deployments, use a cascaded *Initiator Group*, such as each host with all its initiators are contained in separate IGs, and a parent IG that contains all of them and is used for masking. This way, nodes can easily be added or removed from the configuration and immediately have their access to the database added or removed.

This example shows masking views for the Production host (the scripts use the storage group names as shown in these examples).

```
#!/bin/bash
# To find HBA port WWNs run the following command:
# cat /sys/class/fc_host/host?/port_name
set -x
export SYMCLI_SID=000196702151

symaccess -type port -name prod_pg create -dirport 1D:8,2D:8,3D:8,4D:8                <- Port group

symaccess -type initiator -name prod_ig create                                       <- Initiator group
symaccess -type initiator -name prod_ig add -wwn 21000024ff3de26e
symaccess -type initiator -name prod_ig add -wwn 21000024ff3de26f
symaccess -type initiator -name prod_ig add -wwn 21000024ff3de19c
symaccess -type initiator -name prod_ig add -wwn 21000024ff3de19d

symaccess -type storage -name prod_redo_sg create 33:36                              <- REDO SG
symaccess -type storage -name prod_data_sg create 3B:40                              <- DATA SG
symaccess -type storage -name prod_fra_sg  create 37:3A                              <- FRA SG

symaccess -type storage -name prod_database_sg create sg prod_redo_sg,prod_data_sg   <- Parent SG

symaccess create view -name prod_db_mv –sg prod_database_sg –ig prod_ig –pg prod_pg          <- DB masking view
symaccess create view -name prod_fra_mv –sg prod_fra_sg –ig prod_ig –pg prod_pg      <- FRA masking view
```

If using PowerPath there is no concept of “nice-names”. If using native multipathing (device mapper) and utilizing the ‘nice-names’ feature, it is recommended to include not only the Production devices in /etc/multipath.conf, but also entries for the +DATA and +FRA encapsulated *restore* devices, in case they are mounted to Production for recovery. In the following example, they are called “ora_dd_dataNN”, and “ora_dd_fraNN”:

```
...
multipaths {
       multipath {
               wwid                    360000970000196702151533030303333
               alias                   ora_redo1
       }
       multipath {
               wwid                    360000970000196702151533030303335
               alias                   ora_redo2
       }
       multipath {
               wwid                    360000970000196702151533030303334
               alias                   ora_redo3
       }
       multipath {
               wwid                    360000970000196702151533030303336
               alias                   ora_redo4
       }
       multipath {
               wwid                    360000970000196702151533030303342
               alias                   ora_data1
       }
...
       multipath {
               wwid                    360000970000196702151533030303537
               alias                   ora_dd_data1
       }
       multipath {
               wwid                    360000970000196702151533030303538
               alias                   ora_dd_data2
       }
...
```

SET UP MOUNT HOST (OPTIONAL)

The Mount host is optional and can be used for logical recoveries or to browse through Data Domain backups (using the encapsulated *restore* devices) before performing a rollback.

Create a masking view for the Mount host, with the following guidelines:

- If the Mount host is also the Management host, add a masking view with gatekeepers separately from other masking views that contain the encapsulated *restore* devices.
- The views with the appropriate encapsulated *restore* devices are created *later* as part of the recovery scenarios, using the script: './se_mask.sh'. However, the appropriate port groups (PG), initiator groups (IG), and storage groups (SG) should be prepared ahead of time, as shown here.
- If the Mount host is also the Management host, the GK's view should be created at this time.

```
export SYMCLI_SID= 000196702151
symaccess -type initiator -name mount_ig create                          <- Mount host IG
symaccess -type initiator -name mount_ig add -wwn 21000024ff3de192
symaccess -type initiator -name mount_ig add -wwn 21000024ff3de193
symaccess -type initiator -name mount_ig add -wwn 21000024ff3de19a
symaccess -type initiator -name mount_ig add -wwn 21000024ff3de19b

symaccess -type port -name mount_pg create -dirport 1D:8,4D:8          <- Mount host PG

symaccess -type storage -name rstr_redo_sg create 45:48                      <- REDO SG
symaccess -type storage -name rstr_data_sg create 57:5C                      <- DATA SG
symaccess -type storage -name rstr_fra_sg  create 4D:50                      <- FRA SG

symaccess -type storage -name rstr_database_sg create sg rstr_redo_sg,rstr_data_sg    <- Parent SG

# if Mount host is also Management host:
symaccess -type storage -name mount_gk create 13:17
symaccess view create -name mount_gk_mv -sg mount_gk -ig mount_ig -pg mount_pg
```

- If using dm-multipath ensure you add entries in /etc/multipath.conf for the +DATA, +REDO, and +FRA encapsulated *restore* devices. In the following example, they are called "ora_dd_dataNN", "ora_dd_redoNN" and "ora_fraNN":

```
...
       multipath {
               wwid                    360000970000196702151533030303537
               alias                   ora_dd_data1
       }
       multipath {
               wwid                    360000970000196702151533030303435
               alias                   ora_dd_redo1
       }
       multipath {
               wwid                    360000970000196702151533030303444
               alias                   ora_dd_fra1
       }

...
```

- If RAC is used on the Mount host, it is recommended to configure Grid Infrastructure (+GRID ASM disk group) in advance with the cluster configuration and quorum devices. This way, when a backup is ready to be mounted to the Mount host the encapsulated *restore* devices will be masked to the host (made visible), and ASM can simply mount (open) the ASM disk groups.

- If RAC is not used on the Mount host, then ASM will not be able to start until it has access to the initial ASM disk group (which will not be available until a backup-set is mounted to the Mount host). To prepare, perform the following steps:
    - Install Grid and Oracle Database *binaries only* with the same version as Production (do not create disk groups or database).
    - Extract the ASM init.ora file from *Production*, and then copy it to the *Mount host*.

```
[oracle@dsib1141 dbs]$ TOGRID
[oracle@dsib1141 dbs]$ sqlplus "/ as sysasm"
SQL> create pfile='/tmp/initASM.ora' from spfile;
[oracle@dsib1141 dbs]$ scp /tmp/initASM.ora dsib1136:/download/ProtectPoint/Scripts/initASM.ora
```

    - Later on, when the ASM disk group devices become visible to the Mount host, run the following commands (mounting the appropriate ASM disk groups as necessary to the recovery scenario).

```
[oracle@dsib1136 ~]$ srvctl add asm -p /download/ProtectPoint/Scripts/initASM.ora
[oracle@dsib1136 ~]$ srvctl start asm
[oracle@dsib1136 ~]$ srvctl status asm
```

- When preparing for the encapsulated *restore* devices on the mount host:
    - It is highly recommended that the +DATA, +REDO, and +FRA encapsulated *restore* devices are presented to the Mount host during setup, and the Mount host is rebooted once, so the devices can be registered with the host. In that way the host will not need to be rebooted again later or the SCSI bus scanned to discover them.
    - Match Mount host ASM disk string and device names with Production:
        - If Production uses ASMlib, then during the recovery use cases on the Mount host, it can simply rescan for the new storage devices, find its own labels, and mount them. No further work is required.
        - If Production uses Dell EMC PowerPath (without ASMlib), then during recovery use cases on the Mount host, ASM will find its own labels. No further work is required.
    - If dm-multipath is used, the file /etc/multipath.conf should contain similar aliases to the +DATA, +REDO, and +FRA devices from Production, however, *using WWNs of the matching encapsulated restore devices*. This step can only take place later, after the vdisks have been encapsulated.

SET UP DATA DOMAIN SYSTEM

Licenses and SSH

1. License the Data Domain system for vdisk service, remote replications, and so on.

```
[root@dsib1136]# ssh sysadmin@DDS license show
[root@dsib1136]# ssh sysadmin@DDS license add <license-key>
```

2. Set secure SSH between management host and Data Domain system. Note that "DDS" is an entry in /etc/hosts with the IP address of the Data Domain system.

   Note: *Only follow this step if Data Domain CLI will be scripted from Management host*. That way, Data Domain will not ask for password with each set of commands. There is no need to follow this step when ProtectPoint CLIs are used exclusively to communicate with the Data Domain system.

```
[root@dsib1136]# ssh-keygen -t rsa
[root@dsib1136]# ssh sysadmin@DDS adminaccess add ssh-keys < ~/.ssh/id_rsa.pub
```

Set up vdisk service and devices

1. Enable FC service (this may already be enabled).

```
sysadmin@dsib0018# scsitarget enable
sysadmin@dsib0018# scsitarget status
```

2. Enable vdisk service.

```
sysadmin@dsib0018# vdisk enable
sysadmin@dsib0018# vdisk status
```

3. Create a vdisk Pool (for example, ERP).

```
sysadmin@dsib0018# vdisk pool create ERP user sysadmin
sysadmin@dsib0018# vdisk pool show list
```

4. Create vdisk device group (for example, OLTP).

```
sysadmin@dsib0018# vdisk device-group create OLTP pool ERP
sysadmin@dsib0018# vdisk device-group show list
```

5. Create two identical groups of vdisk devices; one for *backup* and one for *restore* devices matching in capacity to the Production host +REDO, +DATA, and +FRA devices.

```
sysadmin@dsib0018# vdisk device create pool ERP device-group OLTP count 8 capacity 10 GiB
sysadmin@dsib0018# vdisk device create pool ERP device-group OLTP count 8 capacity 100 GiB
sysadmin@dsib0018# vdisk device create pool ERP device-group OLTP count 12 capacity 200 GiB
sysadmin@dsib0018# vdisk device show list pool ERP
Device        Device-group   Pool   Capacity                                                WWNN
                                        (MiB)
-----------   ------------   ----   --------   ---------------------------------------------

vdisk-dev0    OLTP           ERP       10241   60:02:18:80:00:08:a0:24:19:05:74:cc:27:40:00:1c
vdisk-dev1    OLTP           ERP       10241   60:02:18:80:00:08:a0:24:19:05:74:cc:27:40:00:1d
vdisk-dev2    OLTP           ERP       10241   60:02:18:80:00:08:a0:24:19:05:74:cc:27:40:00:1e
vdisk-dev3    OLTP           ERP       10241   60:02:18:80:00:08:a0:24:19:05:74:cc:27:40:00:1f
...
```

Note: If the VMAX native devices were created using capacity notation of “MB” or “GB” you can also create the vdisks with “MiB” and “GiB” matching capacities. Otherwise, inspect the geometry of the VMAX native device using a symdev show command, and use a matching heads/cylinders/sectors geometry when creating the vdisk devices.

For example, using <heads> <cylinders> and <sectors> instead of MiB or Gib:

```
# On Management host:
[root@dsib1141 ~]# symdev show 013
...
        Geometry             : Native
            {
            Sectors/Track        :        256      => Equivalent to vdisk “Sectors per track”
            Tracks/Cylinder      :         15      => Equivalent to vdisk “Heads”
            Cylinders            :       5462      => Equivalent to vdisk “Cylinders”
            512-byte Blocks      :   20974080
            MegaBytes            :      10241
            KiloBytes            :   10487040
            }

# On Data Domain:
sysadmin@dsib0018# vdisk device create count <count> heads <head-count> cylinders <cylinder-
count> sectors-per-track <sector-count> pool <pool-name> device-group <device-group-name>
```

Create Access Group for Data Domain device masking

Create a Data Domain Access Group and add the vdisks to it.

> **Note**: Data Domain uses a similar concept to VMAX device masking with an access group containing the initiators (VMAX back-end port in this case, which will only be visible *after* the zoning was done correctly), and the vdisks devices to be encapsulated.

```
sysadmin@dsib0018# scsitarget group create ERP_AG service vdisk          <- Create Access Group
sysadmin@dsib0018# scsitarget initiator show list                        <- List VMAX DX ports
Initiator     System Address            Group   Service
-----------   -----------------------   -----   -------
initiator-1   50:00:09:73:50:21:9c:85   ERP_AG     VDisk
initiator-2   50:00:09:73:50:21:9c:c5   ERP_AG     VDisk
initiator-3   50:00:09:73:50:21:9c:89   ERP_AG     VDisk
initiator-4   50:00:09:73:50:21:9c:c9   ERP_AG     VDisk
-----------   -----------------------   -----   -------
sysadmin@dsib0018# scsitarget group add ERP_AG initiator initiator-*     <- Add DX ports to Access Group
sysadmin@dsib0018# scsitarget endpoint show list                         <- List DD FC ports
Endpoint        System Address   Transport      Enabled   Status
-------------   --------------   ------------   -------   ------
endpoint-fc-0   0a               FibreChannel   Yes       Online       <- Check their status
endpoint-fc-1   0b               FibreChannel   Yes       Online
endpoint-fc-2   1a               FibreChannel   Yes       Online
endpoint-fc-3   1b               FibreChannel   Yes       Online
-------------   --------------   ------------   -------   ------
sysadmin@dsib0018# vdisk device show list pool ERP                       <- List vdisk devices
sysadmin@dsib0018# scsitarget group add ERP_AG device vdisk-dev*         <- Add vdisks to Access Group
sysadmin@dsib0018# scsitarget group show detailed ERP_AG                 <- List Access Group details
```

Set up encapsulated vdisk

After the vdisks are created and added to a Data Domain Access Group they will become visible to the VMAX and therefore can be encapsulated.

In the following example, a script on the Management host is used to capture the WWN of the Data Domain vdisks and encapsulate them:

1. ssh to the Data Domain system and capture the list of vdisks in pool ERP with their WWNs. Save it in file: ‘vdisk_wwn.txt’.

2. Remove everything except the WWN of each vdisk. Remove the colon from the WWNs. Save the output in file: ‘vdisk_wwn_only.txt’.

3. Create a command line for symconfigure that creates encapsulated external disks. Save it in file: ‘CMD.txt’.

4. Execute the command file from a symconfigure CLI command.

```
#!/bin/bash
set -x

# Get vdisk WWN's from DDS
############################

# DDS output looks like this:
# Device        Device-group   Pool   Capacity                                                WWNN
#                                        (MiB)
# -----------   ------------   ----   --------   ---------------------------------------------

  vdisk-dev0    OLTP           ERP       10241   60:02:18:80:00:08:a0:24:19:05:74:cc:27:40:00:1c
  vdisk-dev1    OLTP           ERP       10241   60:02:18:80:00:08:a0:24:19:05:74:cc:27:40:00:1d
  vdisk-dev2    OLTP           ERP       10241   60:02:18:80:00:08:a0:24:19:05:74:cc:27:40:00:1e
# ...
ssh sysadmin@DDS "vdisk device show list pool ERP" | grep vdisk > ./vdisk_wwn.txt
```

```
# Leave only WWNs and remove the colon
#####################################
rm -f ./vdisk_wwn_only.txt
while read line; do
  stringarray=($line)
  echo ${stringarray[4]} | sed 's/[\:_-]//g' >> ./vdisk_wwn_only.txt
done < ./vdisk_wwn.txt

# Create a symconfigure command file
####################################
rm -f ./CMD.txt
while read line; do
  CMD="add external_disk wwn=$line, encapsulate_data=yes;"
  echo $CMD >> ./CMD.txt
done < ./vdisk_wwn_only.txt

# Execute the command file
##########################
symconfigure -sid 2151 -nop -v -file ./CMD.txt commit
```

To list encapsulated devices, use these commands:

```
[root@dsib1141 scripts]# symdev list -encapsulated –gb
[root@dsib1141 scripts]# symdev list -encapsulated -wwn_encapsulated
```

Now that the vdisks are encapsulated, build the storage groups for them using these commands:

```
symaccess -type storage -name rstr_redo_sg create devs 45:48
symaccess -type storage -name rstr_data_sg create devs 57:5C
symaccess -type storage -name rstr_fra_sg create devs 4D:50
symaccess -type storage -name rstr_database_sg create sg rstr_data_sg,rstr_redo_sg
```

Set up initial SnapVX sessions

To create the initial snapshot, use the SnapVX **establish** command. This example uses two storage groups (SGs) for the SnapVX session between the native VMAX devices and the Data Domain encapsulated *backup* devices: prod_database_sg (which includes all data, log, and control files), and prod_fra_sg (which includes the archive logs). Use the following scripts to create the initial snapshots.

***Script note**: The 'se_snap_create.sh' first parameter specifies on which devices to operate: Production (prod) or Restore (rstr). The second parameter indicates whether to snap data (just +DATA), database (+DATA and +REDO), or fra (+FRA, archive logs).*

```
[root@dsib1136 ~]# ./se_snap_create.sh prod database
[root@dsib1136 ~]# ./se_snap_create.sh prod fra
```

Use the SnapVX *link –copy* command to copy the snapshot data to Data Domain encapsulated *backup* devices. This example uses the SGs: bkup_database_sg, and bkup_fra_sg as the target SGs for the copy. Use the following scripts to perform it.

***Script note**: The 'se_snap_link.sh' first parameter specifies on which devices to operate: Production (prod) or Restore (rstr). The second parameter indicates whether to link-copy data (just +DATA), database (+DATA and +REDO), or fra (archive logs).*

***Script note**: The 'se_snap_verify.sh' first parameter specifies on which devices to operate: Production (prod) or Restore (rstr). The second parameter indicates whether to link-copy data (just +DATA), database (+DATA and +REDO), or fra (archive logs). The third uses '0' to indicate this is the initial link (that is, the initial snapshot created by the admin user) or '1' for monitoring later snapshots created by ProtectPoint Commands (ProtectPoint renames the snapshot from its original name to: "NSM_SNAPVX").*

Note: The first *link -copy* is a full copy. Subsequent links will only send changed data to the encapsulated *backup* devices.

```
[root@dsib1136 ~]# ./se_snap_link.sh prod database
[root@dsib1136 ~]# ./se_snap_link.sh prod fra
[root@dsib1136 ~]# ./se_snap_verify.sh prod database 0    <- monitor the copy progress
[root@dsib1136 ~]# ./se_snap_verify.sh prod fra 0         <- monitor the copy progress
```

When monitoring *link -copy* to Data Domain wait until the copy changed to 'D' (destaged) state, which means all write pending tracks from VMAX cache were sent to Data Domain back-end devices.

```
[root@dsib1136 scripts]# symsnapvx -sg prod_fra_sg -snapshot_name prod_fra list -linked -detail
-i 15
Storage Group (SG) Name   : prod_fra_sg
SG's Symmetrix ID         : 000196702151    (Microcode Version: 5977)

-------------------------------------------------------------------------------------------
Sym                                        Link  Flgs                          Remaining  Done
Dev   Snapshot Name                   Gen  Dev   FCMD Snapshot Timestamp       (Tracks)   (%)
----- ------------------------------- ---- ----- ---- ----------------------- ---------- ----
00037 prod_fra                           0 0003B .D.X Sun Mar 22 21:13:04 2015          0  100
00038 prod_fra                           0 0003C .D.X Sun Mar 22 21:13:04 2015          0  100
00039 prod_fra                           0 0003D .D.X Sun Mar 22 21:13:04 2015          0  100
0003A prod_fra                           0 0003E .D.X Sun Mar 22 21:13:04 2015          0  100
                                                                                  ----------
                                                                                           0


Flgs:

  (F)ailed   : F = Force Failed, X = Failed, . = No Failure
  (C)opy     : I = CopyInProg, C = Copied, D = Copied/Destaged, . = NoCopy Link
  (M)odified : X = Modified Target Data, . = Not Modified
  (D)efined  : X = All Tracks Defined, . = Define in progress
```

Note: When an initial *link -copy* is executed while IOs are running it is important to monitor the WP. It was discovered that when the WP count gets to 60 percent the Host IOPS drops off considerably. This is because at 60 percent WP the copy becomes "Sync copy". So the VMAX will wait for a response from the DD that the track is written to disk before accepting the next IO. This is just for the initial full copy. The incremental copies are not affected in this way.

Set up ProtectPoint File System Agent software

1. Copy ProtectPoint File System Agent software to the Management host where Solutions Enabler is installed, then untar the kit and install the rpm.

```
[root@dsib1136 Software]# tar xvfz ppagents31_linux_x86_64.tar.gz
[root@dsib1136 Software]# rpm –ivh emcppfsagent-3.1.0-0.x86_64.rpm
```

2. Update the user PATH to include the location of the software. For example: /opt/emc/ppfsagent/bin/protectpoint.
3. Update /etc/hosts to include IPv6 and localhost.

```
[root@dsib1136 config]# cat /etc/hosts
127.0.0.1   localhost
::1       localhost       loopback
```

4. Update or create the ProtectPoint configuration files. Each backup session (database, or fra in this case) will have its own ProtectPoint configuration file with its unique devices. While working with the configuration file is cumbersome, when it is set, it can be reused with every backup without a change.

   Note: Remember to update the configuration file if the Oracle database or FRA devices change.

   Note: To simplify updating the configuration file(s) always refer to the configuration in Table 2 (and keep it up-to-date).

5. During the ProtectPoint installation, the first ProtectPoint configuration is created. Find it in the installation base directory under ./config/, for example, using the default installation directory: /opt/emc/ppfsagent/config/. Copy the configuration file to a new location and update it.

   Example of ProtectPoint **database** configuration file:

```
[root@dsib1136 ~]# cat /download/ProtectPoint/Configs/PP_database.config
########################################################################
# ProtectPoint 3.1 example input file for vmax configuration
# Indentation used for readability
#
# THIS IS JUST A SAMPLE AND MUST BE MODIFIED TO SUIT YOUR ENVIRONMENT
#
########################################################################
# Mandatory - ProtectPoint environment
[GENERAL]
    # Mandatory, no default value - ProtectPoint Array Type, VMAX=Symmetrix 3
    # Do NOT change this value
    PP_ARRAY_TYPE=VMAX

    # Application Name, optional, no default value
    # APP_NAME = <Application name>
    APP_NAME = Slob

    # Application version, optional, no default value
    # APP_VERSION = <Application version>
    APP_VERSION = 2.3

    # Application information/description, optional, no default value
    # APP_INFO = <Application information>

    # Optional, default value=<install dir>/lockbox - The directory of the ProtectPoint RSA lock box
    # LOCKBOX_DIR = <Directory path>

    # Optional, default value=<install dir>/logs -  the directory of the ProtectPoint log files
    # LOG_DIR = <Directory path>

    # Optional, default value=2
    # 2: error+warning, 3: error + warning + info, 4: error + warning + info + debug
    # DEBUG_LEVEL = 2

    # Optional, default value=4MB -  The log file size
    # LOGFILE_SIZE = <Log file size in MB>

    # Optional, default=16 - Number of ProtectPoint log files to retain
    # LOGFILE_COUNT = <Number of log files>

###################### Primary Data Domain ##################################
# Mandatory - The Primary Data Domain system where VMAX-managed devices will be backed up to
[PRIMARY_SYSTEM]
    # Mandatory, no default value - The Primary Data Domain vdisk and Boost hostname or IP
    # If DD_BOOST_FC is set to true, DD_SYSTEM is the name of the Data Domain Fiber Channel server
    # DD_SYSTEM = <Hostname/IP>
    DD_SYSTEM = dsib0018

    # Optional, indicates that ProtectPoint should use Fibre-Channel communications to the
    # Data Domain server instead of standard network communications.
    # DD_BOOST_FC = TRUE

    # Optional, no default value - The Primary Data Domain Boost port number
    # DD_PORT = <Port number>

    # Mandatory, no default value - The Primary Data Domain vdisk username
    # Note: Only one of DDVDISK_USER and DDBOOST_USER is required. If only
    # one is supplied, its value will be used for both.
    # DDVDISK_USER = <Username>
    DDVDISK_USER = sysadmin

    # Optional, no default value - The Primary Data Domain
    # pool name containing vdisk devices used for restore
    #
    # By default, restores are performed using FAST.X restore devices which are selected from the VMAX
```

```
    # storage group "NsrSnapSG". However, if this and the RESTORE_DEVICE_GROUP fields are specified,
    # then restores are done by selecting restore devices from the specified Data Domain pool and
    # group of restore devices. If either RESTORE_DEVICE_POOL or RESTORE_DEVICE_GROUP are specified,
    # both must be specified and VMAX_FASTX_RESTORE_SG cannot be specified
    # RESTORE_DEVICE_POOL = <Pool name>

    # Optional, no default value - The Primary Data Domain device group used for vdisk restore
    # If either DD_POOL or DD_DEVICE_GROUP are specified, then both must
    # be specified and VMAX_FASTX_RESTORE_SG cannot be specified.
    # RESTORE_DEVICE_GROUP = <Device group name>

    # Mandatory, no default value - The Primary Data Domain DDBOOST user name
    # Note: Only one of DDVDISK_USER and DDBOOST_USER is required. If only
    # one is supplied, its value will be used for both.
    # DDBOOST_USER = <Username>
    DDBOOST_USER = sysadmin

    # Mandatory, no default value - The name of the storage unit or a top-level directory within
    # the Primary Data Domain where the ProtectPoint catalog is maintained
    # DD_PATH = <ddboost path>
    DD_PATH = Slob

    # Optional, Default = "NsrSnapSG" - the name of the VMax storage group
    # to use during VMax restores to select appropriate FAST.X restore devices.
    # If specified, then DD_POOL and DD_DEVICE_GROUP cannot be specified.
    # VMAX_FASTX_RESTORE_SG = <name>
    VMAX_FASTX_RESTORE_SG = rstr_database_sg

    # Optional, default is false. Indicates whether restore devices to be selected must be visible to the
host.
    SELECT_VISIBLE_RESTORE_DEVICES = TRUE

################## Secondary Data Domain ##################################
# Optional - The Secondary Data Domain system where a user-replicated backup will be recovered from
# [SECONDARY_SYSTEM]
    # Mandatory, no default value - The Secondary Data Domain Boost hostname or IP
    # If DD_BOOST_FC is set to true, DD_SYSTEM is the name of the Data Domain Fiber Channel server
    # DD_SYSTEM = <Hostname/IP>

    # Optional, indicates that ProtectPoint should use Fibre-Channel communications to the
    # Data Domain server instead of standard network communications.
    # DD_BOOST_FC = TRUE

    # Optional, no default value - The Secondary Data Domain Boost port number
    # DD_PORT = <Port number>

    # Mandatory, no default value - The Secondary Data Domain vdisk username
    # Note: Only one of DDVDISK_USER and DDBOOST_USER is required. If only
    # one is supplied, its value will be used for both.
    # DDVDISK_USER = <Username>

    # Optional, no default value - The Secondary Data Domain
    # pool name containing vdisk devices used for restore
    #
    # By default, restores are performed using FAST.X restore devices which are selected from the VMax
    # storage group "NsrSnapSG". However, if this and the RESTORE_DEVICE_GROUP fields are specified,
    # then restores are done by selecting restore devices from the specified Data Domain pool and
    # group of restore devices. If either RESTORE_DEVICE_POOL or RESTORE_DEVICE_GROUP are specified,
    # both must be specified and VMAX_FASTX_RESTORE_SG cannot be specified
    # RESTORE_DEVICE_POOL = <Pool name>

    # Optional, no default value - The Secondary Data Domain device group used for vdisk restore
    # If either DD_POOL or DD_DEVICE_GROUP are specified, then both must
    # be specified and VMAX_FASTX_RESTORE_SG cannot be specified.
    # RESTORE_DEVICE_GROUP = <Device group name>

    # Mandatory, no default value - The Secondary Data Domain DDBOOST user name
    # Note: Only one of DDVDISK_USER and DDBOOST_USER is required. If only
    # one is supplied, its value will be used for both.
    # DDBOOST_USER = <Username>

    # Mandatory, no default value - The name of the storage unit or a top-level directory within
    # the Secondary Data Domain where the ProtectPoint catalog is maintained
```

```
    # DD_PATH = <ddboost path>

    # Optional, Default = "NsrSnapSG" - the name of the VMax storage group
    # to use during VMax restores to select appropriate FAST.X restore devices.
    # If specified, then DD_POOL and DD_DEVICE_GROUP cannot be specified.
    # VMAX_FASTX_RESTORE_SG = <name>

    # Optional, default is false. Indicates whether restore devices to be selected must be visible to the
host.
    # SELECT_VISIBLE_RESTORE_DEVICES = TRUE

#################### VMax Devices ########################
# Mandatory - The VMax device information section
[BACKUP_SOURCE_DEVICES]
    # Mandatory, no default value - The VMAX symid+sym_deviceid (format "VMax ID:VMax Device ID")
    # SRC_DEVICE1 = <SYMID:DEVID>
    # ASM +DATA disk group
    SRC_DEVICE1 = 000196702151:0003B
    SRC_DEVICE2 = 000196702151:0003C
    SRC_DEVICE3 = 000196702151:0003D
    SRC_DEVICE4 = 000196702151:0003E
    SRC_DEVICE5 = 000196702151:0003F
    SRC_DEVICE6 = 000196702151:00040
    # ASM +REDO disk group
    SRC_DEVICE7 = 000196702151:00033
    SRC_DEVICE8 = 000196702151:00034
    SRC_DEVICE9 = 000196702151:00035
    SRC_DEVICE10 = 000196702151:00036

    # Optional - Additional VMax device device information
    # SRC_DEVICEn = <SYMID:DEVID>

# Optional - A subset of the source devices that will be restored during the "restore prepare" and
"rollback" operations.
# [RESTORE_SOURCE_DEVICES]
[RESTORE_SOURCE_DEVICES]
    # Optional, no default value - The VMAX symid+sym_deviceid (format "VMax ID:VMax Device ID")
    # SRC_DEVICE1 = <SYMID:DEVID>
    # SRC_DEVICEn = <SYMID:DEVID>
```

Note: ProtectPoint does not compare the content of the SnapVX sessions (or storage groups) with the devices listed in the configuration file. ProtectPoint will only operate on the devices that appear in the configuration file.

6. Validate the configuration file using the ProtectPoint command: **config validate**.

Note: Validate the configuration file only after SSH credentials are established, the configuration file is updated with the devices information, and the initial SnapVX sessions are created and linked.

```
 [root@dsib1136 Scripts]# protectpoint config validate config-file
/download/ProtectPoint/Configs/PP_database.config
```

APPENDIX II – PROVIDING SOLUTIONS ENABLER ACCESS TO NON-ROOT USERS

The following appendix describes how to provide database administrators controlled access to Solutions Enabler so they can perform their TimeFinder replication, backup and recovery procedures without needing root access. The feature is called Solutions Enabler array-based access control and can be configured from Unisphere or Solutions Enabler, as shown below.

The components of array-based access control are:

- **Access Groups**—These groups contain unique *Host ID* and descriptive *Host Name* of the non-root users. The host ID is provided by running 'symacl –unique' command on the appropriate host.
- **Access Pools**—Specify the set of devices for operations.
- **Access Control Entry (ACE)**—Entries in the Access Control Database specifying the permissions level for the Access Control Groups and on which pools they can operate.

Array-based access control commands are executed using *symacl –sid –file <filename> preview | prepare | commit*. Where *preview* verifies the syntax, *prepare* runs preview and checks if the execution is possible, and *commit* performs the prepare operations and executes the command.

The storage administrator PIN can be set in an environment variable, SYMCLI_ACCESS_PIN, or entered manually.

Install Solutions Enabler for non-root user

1. On the *Application Management host,* install Solutions Enabler for the Oracle user. The installation has to be performed as root user, though the option for allowing a non-root user is part of the installation.

```
[root@dsib1136 SE]# ./se830409_install.sh -install
...
Install root directory of previous Installation : /home/oracle/SE
Working root directory [/usr/emc] : /home/oracle/SE
...
Do you want to run these daemons as a non-root user? [N]:Y
 Please enter the user name  : oracle
...
```

2. Update the SE daemon_users file:

```
[root@dsib1136 ~]# cd /var/symapi/config/
[root@dsib1136 config]# vi daemon_users
# Add entry to allow user access to base daemon
oracle  storapid
oracle  storgnsd
```

3. Add the SE binaries directory to the user path (for example, /home/oracle/SE/bin), then test the user access:

```
[root@dsib1136 config]# su – oracle
[oracle@dsib1136 ~]$ symcfg disc
[oracle@dsib1136 ~]$ sympd list -gb
```

Set Management Host in Access Controls Database

Dell EMC support personnel will run a Wizard on SymmWin. First, they enter the storage administrator management host unique ID, then the Admin user and PIN (password). The storage administrator should provide them the PIN and unique ID. The unique ID is provided by running: 'symacl –unique' on the Storage Management host. After that they can create access controls from the Storage Management host, as shown below.

Create array-based access control for ProtectPoint Management host

1. Create the Application Access Control Group:

```
[root@dsib1136 ~]# echo "create accgroup protectpoint;" > ./acl_pp_create_accgrp.cmd
[root@dsib1136 ~]# symacl commit -file ./acl_pp_create_accgrp.cmd
```

2. On the *Application Management* host (where the DBA executes backup and recovery operations), get the unique host ID:

```
[root@dsib1136 ~]# symacl -sid 151 -unique
The unique id for this host is: 2F5A05AC-50498CC9-9C38777E
```

3. Add the Application Management host to the Application Access Control group:

```
[root@dsib1136 ~]# echo "add host accid 2F5A05AC-50498CC9-9C38777E name protectpoint_mgmt to
accgroup protectpoint;" > acl_pp_add_host.cmd
[root@dsib1136 ~]# symacl commit -file ./acl_pp_add_host.cmd
```

4. Create Application Access Control pool:

```
[root@dsib1136 ~]# echo "create accpool protectpoint;" > acl_pp_create_pool.cmd
[root@dsib1136 ~]# symacl commit -file ./acl_pp_create_pool.cmd
```

5. Add the Application storage devices to the pool (including target devices):

```
[root@dsib1136 ~]# echo "add dev 13:2A to accpool protectpoint;" > acl_pp_add_devs.cmd
[root@dsib1136 ~]# echo "add dev 3B:42 to accpool protectpoint;" >> acl_pp_add_devs.cmd

[root@dsib1136 ~]# symacl commit -file ./acl_pp_add_devs.cmd
```

6. The Oracle user needs to run the command from the Application Management host before granting access.

```
[oracle@dsib1136 ~]$ sympd list
Symmetrix ID: 000196702151
Symmetrix access control denied the request
```

7. Grant permissions to the Application Access Group (choose appropriately based on documentation).

```
[root@dsib1136 ~]# echo "grant access=BASE,SNAP,BASECTRL to accgroup protectpoint for accpool
protectpoint;" > acl_pp_grant_access.cmd
[root@dsib1136 ~]# symacl commit -file ./acl_pp_grant_access.cmd
```

8. The Oracle user needs to run the command from the ProtectPoint Management host prior to granting access.

```
[oracle@dsib1136 ~]$ sympd list
Symmetrix ID: 000196702151

        Device Name            Dir                  Device
---------------------------- ------- ------------------------------------
                                                                    Cap
Physical               Sym   SA :P   Config        Attribute    Sts   (MB)
---------------------------- ------- ------------------------------------
...
```

9. Review the created access controls.

```
[root@dsib1136 ~]# symacl list -accgroup
[root@dsib1136 ~]# symacl list –accpool
[root@dsib1136 ~]# symacl list –acl
[root@dsib1136 ~]# symacl show accpool protectpoint
[root@dsib1136 ~]# symacl show accgroup protectpoint
```

APPENDIX III – SCRIPTS USED IN THE USE CASES

ORACLE SCRIPTS

ora_begin_backup.sh—The script uses Oracle network to log into Production database and begin hot-backup mode (for Oracle Database prior to 12c).

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/ora_begin_backup.sh
#!/bin/bash
set -x
su - oracle << !
set -x
echo Update the following parameters to a secure backup user that can only perform these operations
export ORACLE_HOME=/u01/app/oracle/12.1/db
export ORACLE_SID=slob
export ORACLE_NET=slob_prod
export ORACLE_USER=system
export ORACLE_PASSWORD=manager

sqlplus \${ORACLE_USER}/\${ORACLE_PASSWORD}@\${ORACLE_NET} << EOF
set echo on feedback on termout on
alter database begin backup;
quit
EOF
!
```

ora_end_backup.sh—The script uses Oracle network to log into Production database and end hot-backup mode (for Oracle Database prior to 12c).

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/ora_end_backup.sh
#!/bin/bash
set -x
su - oracle << !
set -x
echo Update the following parameters to a secure backup user that can only perform these operations
export ORACLE_HOME=/u01/app/oracle/12.1/db
export ORACLE_SID=slob
export ORACLE_NET=slob_prod
export ORACLE_USER=system
export ORACLE_PASSWORD=manager

sqlplus \${ORACLE_USER}/\${ORACLE_PASSWORD}@\${ORACLE_NET} << EOF
set echo on feedback on termout on
alter database end backup;
quit
EOF
!
```

ora_switchandarchive.sh—The script uses Oracle network to log into Production database, perform switch log files, archive the current log, and save a backup of the control file to the +FRA disk group (as it will be backed up shortly after and thus the backup control file will be part of the backup).

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/ora_switchandarchive.sh
#!/bin/bash
set -x
su - oracle << !
set -x
echo Update the following parameters to a secure backup user that can only perform these operations
export ORACLE_HOME=/u01/app/oracle/12.1/db
export ORACLE_SID=slob
export ORACLE_NET=slob_prod
export ORACLE_USER=system
export ORACLE_PASSWORD=manager

#sqlplus "/ as sysdba" << EOF
sqlplus \${ORACLE_USER}/\${ORACLE_PASSWORD}@\${ORACLE_NET} << EOF
alter system switch logfile;
alter system archive log current;
ALTER DATABASE BACKUP CONTROLFILE TO '+FRA/CTRL.BCK' REUSE;
quit
EOF
!
```

PROTECTPOINT SCRIPTS

pp_backup_show_list.sh—The script requires one parameter, which specifies the configuration file to use. Regardless of the choice, it will list all the backups in Data Domain (though the command still requires a configuration file parameter to identify the DD system).

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/pp_backup_show_list.sh
#!/bin/bash
set -x
if [ "$#" -ne 1 ]; then
  echo "options: database|data|fra"
  exit
fi
OPT=$1
case $OPT in
  database|data|fra)
    CONF=$PP_CONF_LOC/PP_${OPT}.config
    protectpoint backup show list config-file $CONF
    ;;

  *)
    echo "options: database|data|fra"
    exit
    ;;

esac
```

pp_snap.sh—ProtectPoint initiates a new backup by refreshing a snapshot. The script requires two parameters: the first determines which configuration file to use. The second parameter provides a description for the backup that is saved in ProtectPoint catalog.

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/pp_snap.sh
#!/bin/bash
set -x
if [ "$#" -ne 2 ]; then
  echo "1) options: database|data|fra 2) backup-description"
  exit
fi
OPT=$1
DESC=$2

case $OPT in
  database|data|fra)
    DESC="$OPT $DESC"
    CONF=$PP_CONF_LOC/PP_${OPT}.config
    protectpoint snapshot create description "$DESC" config-file $CONF
    echo "Backup time: "; date
    ;;

  *)
    echo "options: database|data|fra"
    exit
    ;;

esac
```

pp_backup_create.sh—ProtectPoint sends the backup incrementally to Data Domain system, using SnapVX link –copy. The script requires two parameters: the first determines which configuration file to use. The second parameter is the backup ID that was created during **pp_snap.sh**. Use **pp_backup_show_list.sh** to list the backups first.

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/pp_backup_create.sh
#!/bin/bash
set -x
if [ "$#" -ne 2 ]; then
  echo "1) options: database|data|fra 2) backup-id"
  exit
fi
OPT=$1
BACK_ID=$2
case $OPT in
  database|data|fra)
    CONF=$PP_CONF_LOC/PP_${OPT}.config
    protectpoint backup create backup-id $BACK_ID config-file $CONF
```

```
    ;;

  *)
    echo "options: 1) database|data|fra 2) backup-id"
    exit
    ;;

esac
```

pp_backup_show_detailed.sh—Detailed information about a specific backup id. The script requires two parameters: the first determines which configuration file to use. The second parameter is the backup ID.

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/pp_backup_show_detailed.sh
#!/bin/bash
set -x
if [ "$#" -ne 2 ]; then
  echo "1) options: database|data|fra 2) backup-id number"
  exit
fi
OPT=$1
BACK_ID=$2
case $OPT in
  database|data|fra)
    CONF=$PP_CONF_LOC/PP_${OPT}.config
    protectpoint backup show detailed backup-id $BACK_ID config-file $CONF
    ;;

  *)
    echo "options: database|data|fra"
    exit
    ;;

esac
```

pp_backup_delete.sh—Deletes a ProtectPoint backup. The script requires two parameters: the first determines which configuration file to use. The second parameter is the backup ID.

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/pp_backup_delete.sh
#!/bin/bash
set -x
if [ "$#" -ne 2 ]; then
  echo "1) options: database|data|fra 2) backup-id number"
  exit
fi
OPT=$1
BACK_ID=$2
case $OPT in
  database|data|fra)
    CONF=$PP_CONF_LOC/PP_${OPT}.config
    protectpoint backup delete backup-id $BACK_ID config-file $CONF
    ;;

  *)
    echo "options: database|data|fra"
    exit
    ;;

esac
```

pp_restore_prepare.sh—ProtectPoint places the backup-ID content onto the restore devices and places a lock on them. The script requires two parameters: the first determines which configuration file to use. The second parameter is the backup ID.

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/pp_restore_prepare.sh
#!/bin/bash
set -x
if [ "$#" -ne 2 ]; then
  echo "options: 1) database|data|fra 2) backup-id"
  exit
fi
OPT=$1
case $OPT in
```

```
  database|data|fra)
    CONF=$PP_CONF_LOC/PP_${OPT}.config
    protectpoint restore prepare backup-id $2 config-file $CONF
    ;;

  *)
    echo "options: 1) database|data|fra 2) backup-id"
    exit
    ;;

esac
```

pp_restore_release.sh—ProtectPoint removes the backup-ID content from the restore devices (wipe them clean) and release the locks it placed on them. The script requires two parameters: the first determines which configuration file to use. The second parameter is the backup ID.

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/pp_restore_release.sh
#!/bin/bash
set -x
if [ "$#" -ne 2 ]; then
  echo "options: 1) database|data|fra 2) backup-id"
  exit
fi
OPT=$1
case $OPT in
  database|data|fra)
    CONF=$PP_CONF_LOC/PP_${OPT}.config
    protectpoint restore release backup-id $2 config-file $CONF
    ;;

  *)
    echo "options: 1) database|data|fra 2) backup-id"
    exit
    ;;

esac
```

pp_restore_rollback.sh—ProtectPoint performs a rollback; it places the content of the backup-ID on the restore devices, snap them, and link their data back to the Production devices (overwriting them). The script requires two parameters: the first determines which configuration file to use. The second parameter is the backup id. Remember that most often even if the backup used the ‘database’ configuration file, a rollback will use just the ‘data’ configuration file.

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/pp_restore_rollback.sh
#!/bin/bash
set -x
if [ "$#" -ne 2 ]; then
  echo "options: 1) database|data|fra 2) backup-id"
  exit
fi
OPT=$1
case $OPT in
  database|data|fra)
    CONF=$PP_CONF_LOC/PP_${OPT}.config
    protectpoint rollback backup-id $2 config-file $CONF
    ;;

  *)
    echo "options: 1) database|data|fra 2) backup-id"
    exit
    ;;

esac
```

SOLUTIONS ENABLER SCRIPTS:

se_mask.sh—The script assists with creating, deleting, or showing masking views for the recovery scenarios in the paper. The scripts requires two parameters: the first determines which configuration file to use. The second parameter is the operation (create, delete, or list).

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/se_mask.sh
#!/bin/bash
set -x
if [ "$#" -ne 3 ]; then
  echo "options: prod|mount database|data|fra create|delete|list"
  exit
fi

MASK_TO_HOST=$1
MASK_DEVICES=$2
MASK_OP=$3

if [ $MASK_TO_HOST != "prod" ] && [ $MASK_TO_HOST != "mount" ]; then
  echo "specify if creating should be done to production or mount host"
  echo "options: prod|mount database|data|fra create|delete|list"
fi

if [ $MASK_DEVICES != "database" ] && [ $MASK_DEVICES != "data" ] && [ $MASK_DEVICES != "fra" ]; then
  echo "specify which devices to create: database (data+redo), data, or fra."
  echo "options: prod|mount database|data|fra create|delete|list"
fi

if [ $MASK_OP != "create" ] && [ $MASK_OP != "delete" ] && [ $MASK_OP != "list" ]; then
  echo "specify whether to create, delete, or list createing views"
  echo "options: prod|mount database|data|fra create|delete|list"
fi

# To no overwrite Production original mv's add '_rstr_'
if [ $MASK_TO_HOST == "prod" ]; then
  VIEW_NAME="prod_rstr_${MASK_DEVICES}_mv"
else
  VIEW_NAME="mount_${MASK_DEVICES}_mv"
fi

if [ $MASK_OP == "create" ]; then
  symaccess view create -name $VIEW_NAME -sg rstr_${MASK_DEVICES}_sg \
                        -ig ${MASK_TO_HOST}_ig -pg ${MASK_TO_HOST}_pg
elif [ $MASK_OP == "delete" ]; then
  symaccess view delete -name $VIEW_NAME
fi

symaccess -sid $SYMCLI_SID list view | grep "Symmetrix\|----\|prod_\|mount"
```

se_snap_create.sh—The script assists in creating a snapshot outside of ProtectPoint. The scripts requires two parameters: the first determines which SG to use for the snapshot; the one from Production, or the one from the restore devices. The second parameter is the type of snapshot (database, data, or fra).

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/se_snap_create.sh
#!/bin/bash
set -x
if [ "$#" -ne 2 ]; then
  echo "options: prod|rstr database|data|fra"
  exit
fi
DEV_ORIGIN=$1
FILE_TYPE=$2
if [ $DEV_ORIGIN != "prod" ] && [ $DEV_ORIGIN != "rstr" ]; then
  echo "specify production primary devices, or encapsulated restore devices"
  echo "options: prod|rstr database|data|fra"
  exit
fi
if [ $FILE_TYPE != "database" ] && [ $FILE_TYPE != "data" ] && [ $FILE_TYPE != "fra" ]; then
  echo "specify database or fra devices"
  echo "options: prod|rstr database|data|fra"
  exit
fi
```

```
symsnapvx -sg ${DEV_ORIGIN}_${FILE_TYPE}_sg -name ${DEV_ORIGIN}_${FILE_TYPE} establish -v

symsnapvx list
```

se_snap_link.sh—The script assists in linking a snapshot outside of ProtectPoint. The scripts requires two parameters: the first determines which SG to use for the snapshot link; the one from Production, or the one from the restore devices. The second parameter is the type of snapshot (database, data, or fra).

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/se_snap_link.sh
#!/bin/bash
set -x
if [ "$#" -ne 2 ]; then
  echo "options: prod|rstr database|data|fra"
  exit
fi
DEV_ORIGIN=$1
FILE_TYPE=$2
if [ $DEV_ORIGIN != "prod" ] && [ $DEV_ORIGIN != "rstr" ] ||
   [ $FILE_TYPE != "database" ] && [ $FILE_TYPE != "data" ] && [ $FILE_TYPE != "fra" ]
then
  echo "options: prod|rstr database|data|fra"
  exit
fi

if [ $DEV_ORIGIN == "prod" ]; then
  SRS_SG=prod_${FILE_TYPE}_sg
  TGT_SG=bkup_${FILE_TYPE}_sg
fi
if [ $DEV_ORIGIN == "rstr" ]; then
  SRS_SG=rstr_${FILE_TYPE}_sg
  TGT_SG=prod_${FILE_TYPE}_sg
fi
if [ $DEV_ORIGIN == "rstr" ] && [ $FILE_TYPE == "database" ]; then
  echo "This is a block to prevent overwriting the production redo logs unintentionally."
  exit
fi
SNAP_NAME=${DEV_ORIGIN}_${FILE_TYPE}
symsnapvx -sg ${SRS_SG} -lnsg ${TGT_SG} -snapshot_name ${SNAP_NAME} link -copy -exact

symsnapvx list
```

se_snap_show.sh—The script assists monitoring ProtectPoint snapshot copy progress from Production to Data Domain (protectpoint backup) or from Data Domain to Production (protectpoint rollback). The scripts requires two parameters: the first determines which SG to monitor; the one from Production, or the one from the restore devices. The second parameter is the type of snapshot (database, data, or fra).

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/se_snap_show.sh
#!/bin/bash
set -x
if [ "$#" -ne 2 ]; then
  echo "options: prod|rstr database|data|fra"
  exit
fi
DEV_ORIGIN=$1
FILE_TYPE=$2
if [ $DEV_ORIGIN != "prod" ] && [ $DEV_ORIGIN != "rstr" ]; then
  echo "specify link source: production devices, or encapsulated restore devices"
  echo "options: prod|rstr database|data|fra"
  exit
fi
if [ $FILE_TYPE != "database" ] && [ $FILE_TYPE != "data" ] && [ $FILE_TYPE != "fra" ]; then
  echo "specify which devices to link: database, data or fra devices"
  echo "options: prod|rstr database|data|fra"
  exit
fi

if [ $DEV_ORIGIN == "prod" ]; then
  SG=prod_${FILE_TYPE}_sg
fi
if [ $DEV_ORIGIN == "rstr" ]; then
  SG=rstr_${FILE_TYPE}_sg
```

```
fi

symsnapvx -sg $SG list -linked -copied -detail -i 30
```

se_snap_terminate.sh—The script assists in terminating a snapshot. Normally it is not used, unless ProtectPoint rollback was stopped prematurely and a cleanup is required. The scripts requires three parameters: the first determines which SG to terminate: the one from Production, or the one from the restore devices. The second parameter is the type of snapshot (database, data, or fra). The third parameter determines if this snapshot is managed by ProtectPoint (since it changes the original snapshot name).

```
[root@dsib1136 Scripts]# cat /download/ProtectPoint/Scripts/se_snap_terminate.sh
#!/bin/bash
set -x
if [ "$#" -ne 3 ]; then
  echo "options: 1) prod|rstr 2) database|data|fra 3) PROTECTPOINT|SG"
  exit
fi
DEV_ORIGIN=$1
FILE_TYPE=$2
SNAP_NAME=$3

if [ $DEV_ORIGIN != "prod" ] && [ $DEV_ORIGIN != "rstr" ] ||
   [ $FILE_TYPE != "database" ] && [ $FILE_TYPE != "data" ] && [ $FILE_TYPE != "fra" ]
then
  echo "options: 1) prod|rstr 2) database|data|fra 3) PROTECTPOINT|SG"

  exit
fi

if [ $SNAP_NAME == "SG" ]; then
  SNAPSHOT_NAME=${DEV_ORIGIN}_${FILE_TYPE}
elif [ $SNAP_NAME == "PROTECTPOINT" ]; then
  if [ $DEV_ORIGIN == "prod" ]; then
    SNAPSHOT_NAME=PROTECTPOINT_SNAP
  else
    SNAPSHOT_NAME=PROTECTPOINT_RESTORE
  fi
fi

#if [ $DEV_ORIGIN == "prod" ]; then
#  exit # Blocked to prevent terminating Prod snap. remove if truely necessary.
#fi

symsnapvx -sg ${DEV_ORIGIN}_${FILE_TYPE}_sg -snapshot_name $SNAPSHOT_NAME terminate -v

symsnapvx list
```

se_encapsulate.sh—The script assists in manual encapsulation of Data Domain vdisks by capturing their WWNs, removing the colons (':'), and creating a symconfigure command to perform the encapsulation in a single operation.

```
[root@dsib1136 Scripts]# cat ./se_encapsulate.sh
#!/bin/bash
set -x
# Get vdisk WWN's from DDS
############################
# DDS output looks like this:
# Device        Device-group   Pool   Capacity                                              WWNN
#                                     (MiB)
# -----------   ------------   ----   --------   ----------------------------------------------
# vdisk-dev0    OLTP           ERP       10241   60:02:18:80:00:08:a0:24:19:05:48:90:7a:d0:00:00
# vdisk-dev1    OLTP           ERP       10241   60:02:18:80:00:08:a0:24:19:05:48:90:7a:d0:00:01
# vdisk-dev2    OLTP           ERP       10241   60:02:18:80:00:08:a0:24:19:05:48:90:7a:d0:00:02
# ...

ssh sysadmin@DDS "vdisk device show list pool ERP" | grep vdisk > ./vdisk_wwn.txt

# Remove irrelevant lines and remove colon from WWNs
####################################################
rm -f ./vdisk_wwn_only.txt
while read line; do
  stringarray=($line)
  echo ${stringarray[4]} | sed 's/[\:_-]//g' >> ./vdisk_wwn_only.txt
```

```
done < ./vdisk_wwn.txt

# Create and execute a symconfigure command file
################################################
rm -f ./CMD.txt
while read line; do
  CMD="add external_disk wwn=$line, encapsulate_data=yes;"
  echo $CMD >> ./CMD.txt
done < ./vdisk_wwn_only.txt
symconfigure -sid 2151 -nop -v -file ./CMD.txt commit
```

se_aclx.sh—The script assists in creating the masking views from the management host, which in this paper is the same as the Mount host. Note that many of the other scripts in this paper depend on the storage group names as they are specified in this masking view script.

```
[root@dsib1136 ~]# cat /download/ProtectPoint/Scripts/se_aclx.sh
#!/bin/bash
# To find HBA port WWNs run the following command:
# cat /sys/class/fc_host/host?/port_name

set -x
export SYMCLI_SID=000196702151

symaccess -type storage -name prod_redo_sg create devs 33:36
symaccess -type storage -name prod_fra_sg create devs 37:3A
symaccess -type storage -name prod_data_sg create devs 3B:40
symaccess -type storage -name prod_database_sg create sg prod_redo_sg,prod_data_sg

symaccess -type storage -name rstr_redo_sg create devs 45:48
symaccess -type storage -name rstr_data_sg create devs 57:5C
symaccess -type storage -name rstr_fra_sg create devs 4D:50
symaccess -type storage -name rstr_database_sg create sg rstr_data_sg,rstr_redo_sg

symaccess -type initiator -name prod_ig create
symaccess -type initiator -name prod_ig add -wwn 21000024ff3de26e
symaccess -type initiator -name prod_ig add -wwn 21000024ff3de26f
symaccess -type initiator -name prod_ig add -wwn 21000024ff3de19c
symaccess -type initiator -name prod_ig add -wwn 21000024ff3de19d

symaccess -type initiator -name mount_ig create
symaccess -type initiator -name mount_ig add -wwn 21000024ff3de192
symaccess -type initiator -name mount_ig add -wwn 21000024ff3de193
symaccess -type initiator -name mount_ig add -wwn 21000024ff3de19a
symaccess -type initiator -name mount_ig add -wwn 21000024ff3de19b

symaccess -type port -name prod_pg create -dirport 1D:8,2D:8,3D:8,4D:8
symaccess -type port -name mount_pg create -dirport 1D:8,4D:8

symaccess view create -name prod_db_mv -sg prod_database_sg -ig prod_ig -pg prod_pg
symaccess view create -name prod_fra_mv -sg prod_fra_sg -ig prod_ig -pg prod_pg
symaccess view create -name mount_gk_mv -sg mount_gk -ig mount_ig -pg mount_pg
```

REFERENCES

- Dell EMC VMAX All Flash storage for mission-critical Oracle databases
- Oracle Database Backup and Recovery with VMAX3
- HYPERMAX OS Local Replication TimeFinder SnapVX and TimeFinder Emulation – Technical Note
- Dell EMC VMAX3 and VMAX All Flash Quality of Service Controls for Multitenant Environments
- ProtectPoint Data Sheet
- http://www.emc.com/data-protection/data-domain/index.htm