Technical Report

Oracle Best Practices on NetApp SolidFire

Bobby Oommen, NetApp
January 2019 | TR-4606

Abstract

NetApp® SolidFire® storage offers a compelling advantage for a wide range of Oracle Database application use cases. This paper introduces SolidFire storage to database administrators and provides important system design paradigms to consider when using SolidFire storage for Oracle databases. From these design points, the reader can learn about application profiles that are ideally suited to SolidFire storage and how to identify those types of applications. By following the guidelines in this document, the reader can also learn how to effectively design, implement, and run Oracle databases on SolidFire.

■ NetApp®

TABLE OF CONTENTS

1 Introduction

NetApp SolidFire storage systems were born out of some of the largest cloud infrastructures in the world. They are designed to serve next-generation data center needs such as scaling with multitenancy, set-and-forget management, and guaranteed performance. SolidFire architecture provides you with greater predictability for your shared storage infrastructure. With SolidFire storage, you can optimize solid-state drive (SSD) capacity to create high utilization and volume performance. The innovative architecture of SolidFire features can be leveraged across several different Oracle Database use cases to provide real benefits to administrators.

This guide discusses basic steps and best practices to set up an Oracle Database instance on SolidFire storage volumes. NetApp SolidFire storage systems have the following capabilities that support next-generation data center needs:

- Thin provisioning
- Compression and deduplication
- Quality of service (QoS)

These features reduce the amount of storage space that is required without affecting performance. You can use these features with various database use cases.

1.1 Thin Provisioning

SolidFire uses 4K granular thin provisioning that does not require any reserve space, increasing effective capacity by consuming less space. This feature increases efficiency and reduces overhead by using the smallest allocation possible and by maintaining alignment with the native 4KB allocation format that modern operating systems use.

Because SolidFire volumes do not use any reserve space, you can deploy a volume capacity for the estimated maximum size of the database. You can therefore purchase enough physical hardware to support the actual space that is consumed by the database. As database space consumption approaches the physical limits of the installed cluster, you can dynamically add nodes to the cluster to increase its physical capacity. This process is completely transparent to the database and does not require downtime or reconfiguration of the operating system or the database.

Furthermore, SolidFire redundant data protection automatically redistributes existing data over the added nodes to provide optimal load balancing of both existing and new data. With this deployment paradigm, you can configure logical storage capacity once for the lifetime of the supported databases rather than using incremental updates that depend on the needs of the database.

1.2 Compression and Deduplication

Each SolidFire node includes a PCIe NVRAM card that serves as a write cache. When a host sends writes, they are divided into 4KB blocks that are immediately compressed, hashed, and stored in the NVRAM of the two storage nodes before an acknowledgment is returned. The resulting value serves as a block ID that determines the block placement and that is randomly distributed across all nodes to create an even load distribution.

The SolidFire deduplication block service identifies blocks that have previously been written based on the block ID. If a block already exists, metadata is updated accordingly, and the duplicate is discarded. The whole process is inline and global to the storage cluster.

The combination of inline compression and global deduplication has the following advantages:

- Reduced repetitive writes to media, increasing drive life
- Increased system performance by minimizing system resources
- Evenly distributed capacity and performance loads across the system, eliminating hot spots

1.3 Quality of Service (QoS)

NetApp SolidFire storage arrays present performance and capacity as dynamic, independent pools. This feature enables administrators to set the performance requirements for all the databases or tenants that are hosted on the same cluster. The minimum, maximum, and burst control settings in QoS guarantee the required performance and can be dynamically changed anytime. If the SolidFire hardware resources are pushed to their physical limits, more nodes can be added to the existing cluster. SolidFire redundant data protection automatically redistributes data for optimal load balancing over all hardware resources. This process is transparent to upstream applications.

2 Application Use Cases

NetApp SolidFire can support a wide range of Oracle Database application use cases, including OLTP; decision support systems (DSSs); database consolidation; data protection; disaster recovery; and development, testing, and quality assurance. This section shows how to identify when these applications are a good fit for SolidFire. It also reviews the innovative benefits of SolidFire to these applications.

2.1 Database Consolidation

NetApp SolidFire provides an optimal storage system for database consolidation. The per-volume QoS controls of SolidFire help individual databases get the I/O throughput that they need without being affected by other databases that run in parallel on the same storage system. With QoS and data reduction efficiencies, you can achieve higher database density within the shared storage infrastructure. The use case of deploying hundreds of individual databases is an excellent fit for SolidFire.

With SolidFire, you can use a single LUN to provision all the data files if the overall throughput requirement is under 600MBps. This configuration can achieve the required performance level without spreading data files across multiple volumes, controllers, and arrays. SolidFire QoS provides guaranteed performance for each database. For use cases that need of throughput above 600MBps, NetApp recommends striping the data across multiple LUNS.

2.2 Data Protection and Disaster Recovery

The combination of horizontal scale-out architecture and high-performance flash drives in SolidFire provides high throughput to minimize backup windows by using Recovery Manager (RMAN) incremental-apply operations. Guaranteed QoS allows you to adhere to recovery point objectives (RPOs) and recovery time objectives (RTOs). Having an RMAN copy on traditional storage can be a challenge if the backup system cannot keep up with the performance requirements of the production system. Placing the RMAN target on a SolidFire cluster allows you to shrink the backup window by adjusting the throughput requirement using the QoS feature. Also, the storage efficiency features built into SolidFire reduce the amount of storage space used.

SolidFire redundant data protection is a distributed replication algorithm that spreads two redundant copies of data across all drives within the cluster. The shared-nothing architecture of SolidFire creates no single point of failure and can absorb multiple failures across all levels of the solution.

The combined storage efficiency and QoS of SolidFire provide a compelling Oracle disaster recovery solution. A SolidFire Oracle disaster recovery solution allows the sharing of the same storage resources for disaster recovery and testing and development without performance penalties.

2.3 Development and Testing

SolidFire deduplication is a good match for RMAN clones taken from active databases, standby databases, and image-copy backups. This feature is especially useful when many databases are stored in the same ASM disk group because RMAN can clone individual databases. This approach compares favorably to storage system copy technologies that clone every database in the ASM disk group. This

process can be coupled with SolidFire QoS control to make sure that database clones are provisioned to meet the performance requirements of upstream applications.

The CopyVolume feature of NetApp SolidFire allows you to refresh an existing cloned copy of a database without performing file system remount operations. In this use case, you frequently refresh a copy of the database by only taking changes from the production copy.

3   Storage Configuration

This section shows how to configure NetApp SolidFire volumes to support an Oracle Database application. The configuration described in this document is for an Oracle database using an Oracle ASM layout. You can use an LVM configuration as well on Linux to configure the Oracle application.

NetApp recommends having all the Oracle data files, redo logs, archive logs, and cluster-ready services (CRS) for Real Application Clusters (RAC) configured on the SolidFire storage array. SolidFire supports presenting the storage in a 4K sector size (native mode) and in a traditional 512-byte sector size (512e). Testing shows no performance effect from choosing emulation mode as long as there is no partition misalignment at the host level.

3.1   Create an Account

1. Log in to the NetApp Element software UI.
2. Select Management → Accounts. The Account List window opens.
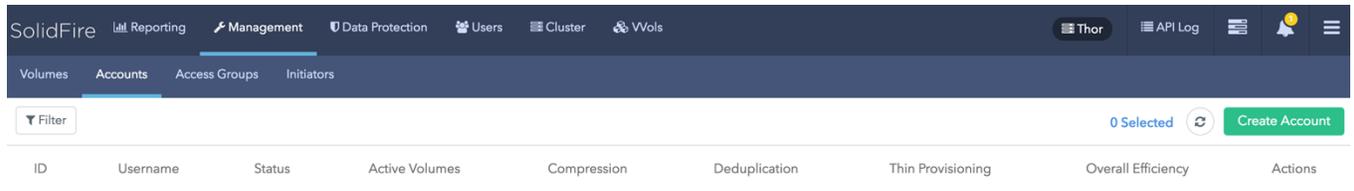
3. Click Create Account. The Create a New Account window opens.
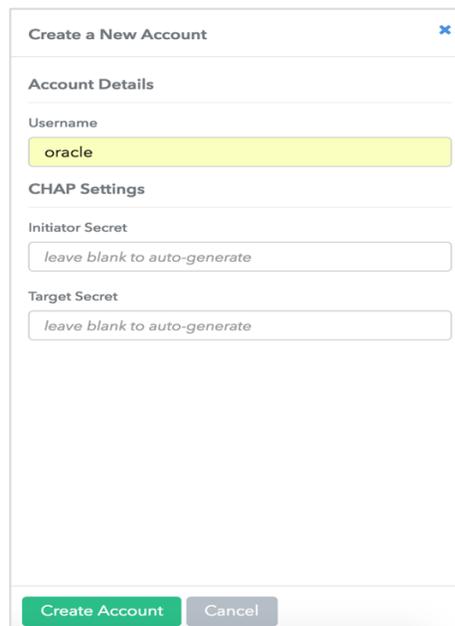
4. Enter a user name.

5. In the CHAP Settings section, fill in the following fields:
   - Initiator secret for CHAP node session authentication
   - Target secret for CHAP node session authentication

   Note: Leave the credentials field blank if you want the passwords to be generated automatically.

6. Click Create Account.

   Note: If an account with the same name exists, an error message will appear.

3.2   Create a QoS Policy

1. Log in to the NetApp SolidFire Element UI.
2. Select Management → QoS Policies. QoS policy window opens.
3. Click Create QoS Policy.
4. Enter the QoS policies as required.

Create a New QoS Policy			✖
Policy Name			
oracle			
Quality of Service			
IO Size	Min IOPS	Max IOPS	Burst IOPS
4 KB	15000	10000	10000
8 KB	9375 IOPS	62500 IOPS	62500 IOPS
16 KB	5556 IOPS	37037 IOPS	37037 IOPS
262 KB	385 IOPS	2564 IOPS	2564 IOPS
Max Bandwidth		699.05 MB/sec	699.05 MB/sec

3.3   Create a Volume

1. Log in to the Element UI.
2. Select Management → Volumes. The Volumes List window opens.
3. Click Create Volume. The Create a New Volume window opens.

4. Enter the volume name (1 to 64 characters in length). For example, provide the name as `ORADATA1`.

5. Enter the size of the volume.

6. Click the Account drop-down list and select the account that should have access to the volume. In this case, select oracle.

7. In Quality of Service options, select Policy and select the `oracle` policy created previously.

8. Click Create Volume.

9. Repeat steps 1 through 7 for all the remaining volumes (CRS1, CRS2, CRS3, ORADATA2, ORADATA3, and ORALOG1).

3.4   Create Volume Access Groups

Volume access groups limit connectivity from designated host servers based on a unique identifier, whereas CHAP authentication uses secret keys for unidirectional or bidirectional authentication. In this document, initiator iSCSI qualified names (IQNs) are used to access the volumes.

Volume access groups have the following system limits:

- They can have a maximum of 64 IQNs.
- An IQN can belong to only one access group.
- A single volume can belong to a maximum of four access groups.

To create volume access groups, complete the following steps:

1. Log in to the Element.

2. Select Management → Access Groups. The Access Group window opens.

3. Click Create Access Group. The Create a New Access Group window opens.

Create a New Access Group

Volume Access Group Details

Name

oracleag

Add Initiators

Initiators

iqn.2015-1.ucs.lab:client:57 ▼   Add Initiator   Create Initiator?

Initiators   0 ∧

Attach Volumes

Volumes

Select a Volume ▼   Attach Volume

Attached Volumes   0 ∧

Create Access Group   Cancel

4. Enter a name for the volume access group.

5. Select the IQN from the initiator drop-down list or click Create Initiator.

   Note: Multiple IQNs are listed for Oracle RAC clusters, depending on the number of nodes.

6. Click Create Access Group.

7. Add the volumes to the access group by selecting Management→Volumes.

8. Select the checkbox to the left of each volume.

9. Near the Create Volume button, click the Bulk Actions drop-down list.

10. Select Add to Volume Access Group.
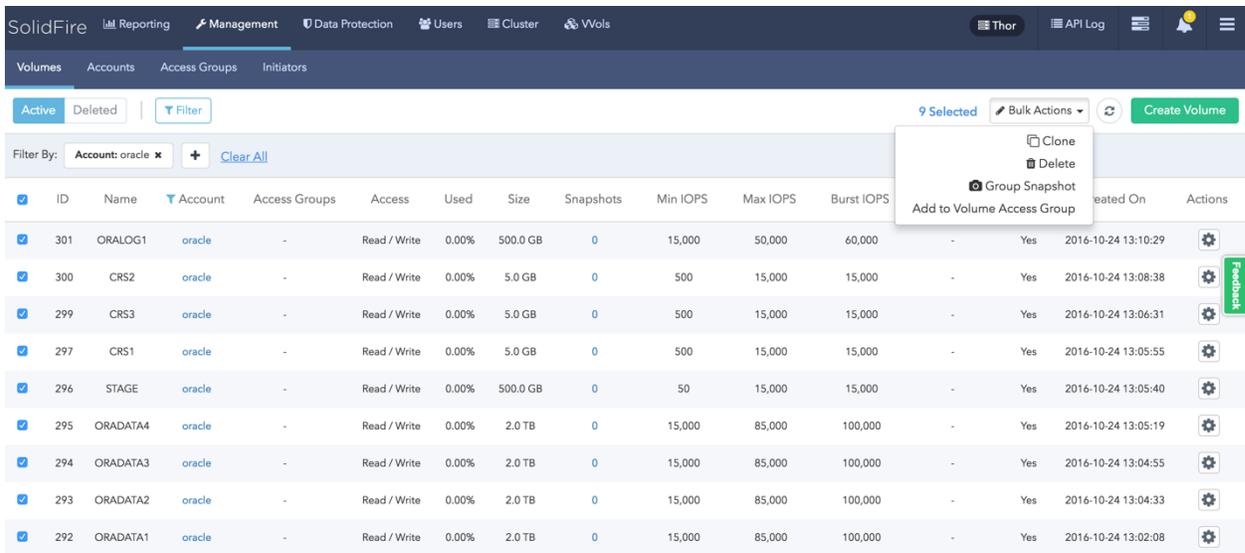
11. Scroll to the bottom and click Add to Volume Access Group. The Add to Volume Access Group window opens.

12. Select the previously created volume access group from the drop-down list.

13. Click Add to join the selected volumes to the target group.

The Oracle volumes are now listed as part of the selected volume access group. They are now ready for presentation to the host operating system after the host identifier is allowed into the volume access group configuration.

Note: For this configuration, four SolidFire volumes were chosen to distribute data across all four volumes in the ASM disk group. For a web-scale deployment, with hundreds of databases, NetApp recommends having just one volume for data for each individual database and that you control performance with the QoS settings.

4 Operating System Configuration

Before beginning the configuration procedures, implement all the operating system recommendations listed in the SolidFire Linux best practices guide. The implementation guidelines in this document apply to Red Hat Enterprise Linux 7.x, distributions of Oracle Enterprise Linux (OEL), and Community Enterprise Operating System (CentOS). These implementation guidelines assume a default installation of a software development workstation. Alternate distributions can be used, assuming full compatibility with the Oracle Grid Infrastructure and Oracle Database software.

4.1 Add Additional Packages

After installing the base operating system, you might need to update the operating system to meet Oracle Grid Infrastructure and Oracle Database installation requirements. Refer to the Oracle documentation to meet these requirements, depending on the Oracle release. See the section "Install Prerequisite Packages" for details.

4.2 Update Kernel Parameters

1. Update the kernel parameters for your host operating system to the following values:

```
net.ipv4.ip_local_port_range = 9000 65500
net.core.rmem_default = 4194304
net.core.rmem_max = 16777216
```

```
net.core.wmem_default = 262144
net.core.wmem_max = 16777216
net.ipv4.ipfrag_high_thresh = 524288
net.ipv4.ipfrag_low_thresh = 393216
net.ipv4.tcp_rmem = 4096 524288 16777216
net.ipv4.tcp_wmem = 4096 524288 16777216
net.ipv4.tcp_timestamps = 0
net.ipv4.tcp_sack = 0
net.ipv4.tcp_window_scaling = 1
net.core.optmem_max = 524287
net.core.netdev_max_backlog = 2500
net.ipv4.tcp_mem = 16384 16384 16384
fs.aio-max-nr = 1048576
net.ipv4.tcp_no_metrics_save = 1
net.ipv4.tcp_moderate_rcvbuf = 0
vm.min_free_kbytes=262144
vm.swappiness=10
```

4.3 Optimize Network Performance

Consider the following guidelines for optimal network performance:

- Enable jumbo frames for all host network interfaces.
- The RAC interconnect should be configured with a different subnet than the RAC VIP so that cluster interconnect traffic is fully isolated.

4.4 Optimize iSCSI Parameters

The Linux iSCSI initiator configuration works with NetApp SolidFire volumes in its default configuration. To maximize system throughput, increase the number of sessions per target (`nr_sessions`) from the default of 1 to 8.

1. Make the following changes to the iSCSI daemon in the `etc/iscsi/iscsid.conf` file:

```
iscsid.startup = /etc/rc.d/init.d/iscsid force-start
node.startup = automatic
node.leading_login = No
node.session.timeo.replacement_timeout = 120
node.conn[0].timeo.login_timeout = 15
node.conn[0].timeo.logout_timeout = 15
node.conn[0].timeo.noop_out_interval = 5
node.conn[0].timeo.noop_out_timeout = 5
node.session.err_timeo.abort_timeout = 15
node.session.err_timeo.lu_reset_timeout = 30
node.session.err_timeo.tgt_reset_timeout = 30
node.session.initial_login_retry_max = 8
node.session.cmds_max = 128
node.session.queue_depth = 32
node.session.xmit_thread_priority = -20
node.session.iscsi.InitialR2T = No
node.session.iscsi.ImmediateData = Yes
node.session.iscsi.FirstBurstLength = 262144
node.session.iscsi.MaxBurstLength = 16776192
node.conn[0].iscsi.MaxRecvDataSegmentLength = 262144
node.conn[0].iscsi.MaxXmitDataSegmentLength = 0
discovery.sendtargets.iscsi.MaxRecvDataSegmentLength = 32768
node.conn[0].iscsi.HeaderDigest = None
node.session.iscsi.FastAbort = Yes
node.startup = automatic
node.session.nr_sessions = 8
```

4.5 Set Ulimits

Oracle recommends setting the user limits to an optimal number, which affects application performance. For this solution, the following user limits were applied for both the `grid` and `oracle` users. The same

settings were applied on all RAC nodes. See the Oracle documentation for any updates to the following settings:

```
grid soft nproc 2047
grid hard nproc 16384
grid soft nofile 1024
grid hard nofile 65536
oracle soft nproc 2047
oracle hard nproc 16384
oracle soft nofile 1024
oracle hard nofile 65536
```

4.6 Permissions

Before the Oracle installation, you should set appropriate permissions so that Oracle can access the underlying devices on the SolidFire storage. You can set the device permissions on multipath devices by creating a udev rule file that allows the owner of the Oracle Database software to access the devices.

5 Oracle Configuration

5.1 Install Prerequisite Packages

NetApp recommends running the `runcluvfy.sh` script that is part of the Oracle installation package under the grid directory. The script does a thorough verification of all the required packages for each RAC node. You can run the script from the first RAC node, and it validates all the other nodes that are part of the cluster. The script can be executed as follows and is available in the installer directory of the grid installation software:

```
./runcluvfy.sh stage -pre crsinst -n rac1,rac2 -fixup -verbose -asm -asmdev
/dev/mapper/crs1,/dev/mapper/crs2,/dev/mapper/crs3
```

5.2 Oracle ASM Configuration

The storage nodes automatically distribute and protects Oracle Database data across all nodes in a SolidFire cluster. For optimal distribution of Oracle Database data on SolidFire storage, use ASM external redundancy to stripe data across storage volumes. This implementation makes sure that all nodes can process Oracle writes and that Oracle Database I/O requests are uniformly distributed over all SolidFire nodes. Storage volumes can be dynamically created and added to Oracle ASM disk groups as capacity and performance requirements for that disk group increase.

NetApp recommends creating more than one storage volume if the overall throughput requirement of the database goes beyond 600MBps. The ASM software should be further configured to use large allocation unit (64M). The larger allocation unit reduces the amount of metadata and metadata updates that ASM requires to support a given amount of storage.

The storage nodes provide complete fault isolation through a comprehensive shared-nothing architecture. Consequently, ASM mirroring within a single cluster is not appropriate. For extreme fault tolerance, ASM mirroring can be used between clusters.

Note: ASMlib is a requirement on Linux when 4KB native volume is used. Any volumes created with default 512e are accessible by Oracle ASM.

5.3 ASMlib Configuration

Oracle ASMlib simplifies storage deployment and management for Linux operating systems. ASMlib is an optional component if SolidFire devices are configured with 512e, but ASMlib is mandatory if SolidFire devices are configured with a native 4KB sector size. Before assigning any LUN or volume to ASMlib, the parameter `ORACLEASM_USE_LOGICAL_ BLOCK_SIZE` must be set to `true`.

6 Virtualization

NetApp SolidFire supports customers who would like to virtualize databases with VMware ESX, Oracle OVM, or KVM. Customers considering virtualization of their databases should base their storage decisions on their business needs. Regarding storage presentation, a storage resource should be managed directly by the virtual machine (VM) guest. Therefore, NetApp recommends using one of the following storage configurations:

- iSCSI LUNs managed by the iSCSI initiator on the VM, not the hypervisor
- Separate datastore/storage LUNS for Oracle data files, redo logs, archive logs

As a rule, avoid using datastores for Oracle files, which will provide better:

- **Transparency.** When a guest VM owns the file systems, it is easier for a database administrator or a system administrator to identify the source of the file systems for their data.
- **Manageability.** When a guest VM owns the file systems, manageability of provisioning, monitoring, and data protection across the environment becomes easier.
- **Portability.** When a VM owns its file systems, the process of moving an Oracle environment becomes simpler. File systems can easily be moved between virtualized and nonvirtualized guests and to other hypervisors.

7 Microsoft Windows Configuration

This section describes how to configure the Windows operating system to use SolidFire. The Windows operating system automatically aligns disk partitions to SolidFire sector boundaries. No specific tuning of the partition table layout or device queue depth settings is required with Windows on SolidFire. To configure Microsoft Windows, complete the following tasks:

1. Enable jumbo frames on the network interface cards (NICs).
2. Enable multipath I/O (MPIO) and configure the Microsoft device-specific module (DSM) to recognize SolidFire devices.
3. Configure at least four iSCSI sessions per SolidFire device.

# 8 Backup and Recovery by Using Storage Snapshot Copies

You can take point-in-time (PiT) snapshots on the NetApp SolidFire array for the volumes that are part of the Oracle Database. The application-consistent snapshots can be used to do database recovery in the event of data corruption or media failure. SolidFire supports having multiple volumes for the database, and all volumes that are part of the database have the same consistency point during the group snapshot.

8.1 Create Snapshots (Backup)

1. Log in to the Oracle instance. To put the database into backup mode, run the following commands:

```
alter system archive log current;
alter database begin backup;
```

2. Log in to the Element UI.
3. Select Management → Volumes. The Volumes List window opens.
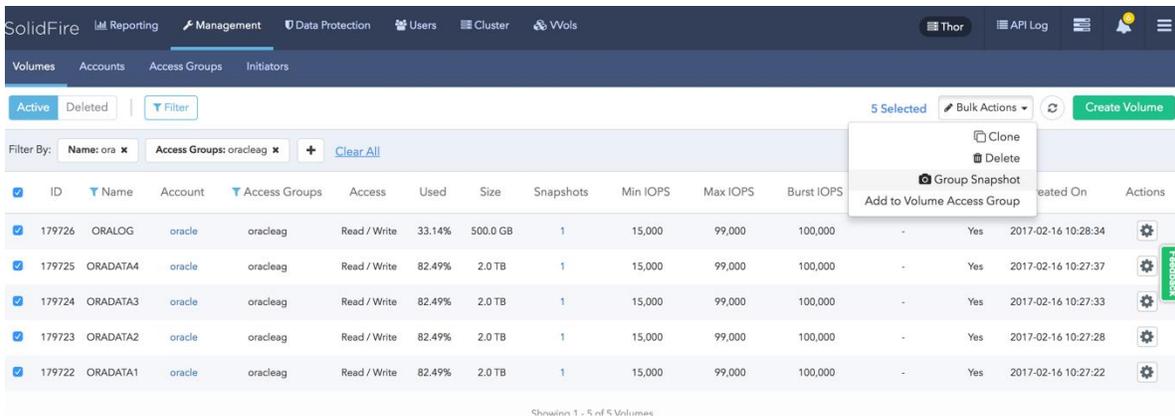
4. Select the volumes that are part of the database.
5. Select Bulk Actions → Group Snapshot.
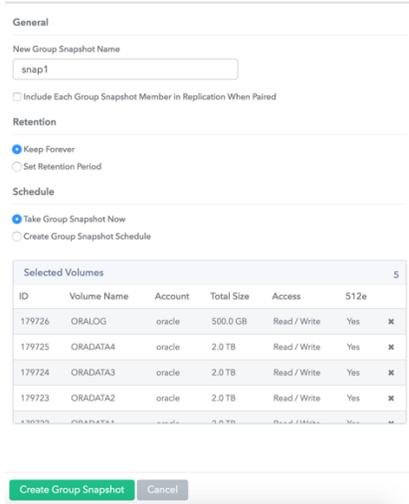
6. In the Create Group Snapshot window, enter a name for the snapshot (Snap1 in this case).
7. Set the desired retention time.
8. Select Take Group Snapshot Now or select Create Group Snapshot Schedule.
9. Scroll to the bottom and click Create Group Snapshot.
10. Take the database out of backup mode.

```
alter database end backup;
```

8.2   Restore Snapshots (Recovery)

Database recovery can be performed by reverting the volumes on the SolidFire array using a snapshot.
To recover the database, complete the following steps:

1. Log in to the Oracle Database server, stop the Oracle instance, and unmount the file systems that are part of the database. If the database is provisioned using ASM, stop the ASM instance.
2. Log in to the Element UI.
3. Select Data Protection→ Group Snapshots. The Group Snapshot list opens.

	ID	▼ Name	Account	▼ Access Groups	Access	Used	Size	Snapshots	Min IOPS	Max IOPS	Burst IOPS	Attributes	512e	Created On	Actions
☑	179725	ORADATA4	oracle	oracleag	Read / Write	82.49%	2.0 TB	0	15,000	99,000	100,000	-	Yes	2017-02-16 10:27:37	⚙
☑	179724	ORADATA3	oracle	oracleag	Read / Write	82.49%	2.0 TB	0	15,000	99,000	100,000	-	Yes	2017-02-16 10:27:33	⚙
☑	179723	ORADATA2	oracle	oracleag	Read / Write	82.49%	2.0 TB	0	15,000	99,000	100,000	-	Yes	2017-02-16 10:27:28	⚙
☑	179722	ORADATA1	oracle	oracleag	Read / Write	82.49%	2.0 TB	0	15,000	99,000	100,000	-	Yes	2017-02-16 10:27:22	⚙

4. Select the volumes that are part of the database.
5. Select Bulk Actions → Group Snapshot.
6. Click Actions.
7. Select Rollback Volumes to Group Snapshot. The Rollback to Group Snapshot window opens.

Rollback To Group Snapshot ✕

Group Snapshot Details
ID: 20   Name: snap1
UUID: c4bff3fc-80fb-47e9-9d5c-f5dcc75e278d
Created On: 2017-03-14 14:01:05

☑ Save volumes' current state as a group snapshot

New Group Snapshot Name
optional

Group Snapshot Members

ID	Name	Size	Volume ID	Volume Name	Account	Access	Retain Until	Replication Enabled	Replicated
182717	snap1	2.0 TB	179723	ORADATA2	oracle	Read / Write	-	No	-
182718	snap1	2.0 TB	179724	ORADATA3	oracle	Read / Write	-	No	-
182716	snap1	2.0 TB	179722	ORADATA1	oracle	Read / Write	-	No	-
182719	snap1	2.0 TB	179725	ORADATA4	oracle	Read / Write	-	No	-
182720	snap1	500.0 GB	179726	ORALOG	oracle	Read / Write	-	No	-

[Rollback Group Snapshot] [Cancel]

8. Click Rollback Group Snapshot.
9. After the volume is reverted successfully, log in to the Oracle instance.
10. Rescan the iSCSI devices.
11. Mount the file systems.
12. If the database is provisioned using ASM, start the ASM instance.
13. Log in to the Oracle instance and run these commands:

```
startup mount;
recover database;
alter database open;
```

9 RMAN Merge

RMAN merged incremental backups can reduce the time for Oracle Database recovery and are suitable primarily for disk-based backup and recovery. This method is widely used by database administrators. As part of the RMAN setup, a level 0 (full image) copy is created. Afterward, the subsequent RMAN runs and creates a level 1 incremental backup of the data file by using the system change number (SCN) as a reference. RMAN then applies all level 1 backups taken since the last backup to the image. The RMAN process uses a tag to link the incremental backups to the image copy. The configuration for this

document is done with level 0 and incremental backups using two different locations on the RMAN target file system.

The target file system of the RMAN dump is created on the SolidFire array and is mapped to the database server with permission set so that an Oracle user can perform reads and writes. Four SolidFire volumes were used for this configuration, and the QoS settings were configured to meet the appropriate throughput requirements. On the Linux host, LVM was used to stripe data across multiple volumes.

9.1 Enable Block Change Tracking

Enabling the block change tracking feature improves the performance of incremental backups by recording changed blocks in the block change tracking file. During a backup operation, RMAN uses this file to identify the changed blocks that need to be backed up instead of scanning all data blocks to identify which blocks have changed. To enable block change tracking, connect to the RMAN target database instance and run the following commands:

```
ALTER SYSTEM SET DB_CREATE_FILE_DEST = '+ORADATA'; #The destination of the file changes depending
on the your setup
ALTER DATABASE ENABLE BLOCK CHANGE TRACKING;
```

9.2 Create RMAN Incremental

To perform the RMAN merge process incorporating SolidFire snapshots, complete the following steps:

1. Set the RMAN configuration parameters for the level 0 image.

```
CONFIGURE DEFAULT DEVICE TYPE TO DISK;
CONFIGURE CONTROLFILE AUTOBACKUP ON;
CONFIGURE BACKUP OPTIMIZATION ON;
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO
'/rmanfs/controlfile/control.%F.ctl';
CONFIGURE DEVICE TYPE DISK PARALLELISM 8 BACKUP TYPE TO BACKUPSET;
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT '/rmanfs/dbimage/full/%U.dbf';
```

2. Run the RMAN backup with the following commands. The first backup is a level 0 backup and takes some time, depending on the amount of data to be copied.

```
RUN {
  BACKUP INCREMENTAL LEVEL 1 FOR RECOVER OF COPY WITH TAG 'ORADB_INCR' DATABASE;
  RECOVER COPY OF DATABASE WITH TAG 'ORADB_INCR';
}
```

3. After a successful level 0 backup, create a snapshot on the SolidFire storage for all the volumes that are part of the RMAN file system.

4. Configure the RMAN parameter to update the location for a level 1 backup.

```
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT '/rmanfs/dbimage/incremental/%U.dbf';
```

5. Create a level 1 backup image by running the same commands from step 2. To simplify this step, you can save the commands to a script file to use on the RMAN command line. You can take the subsequent level 1 backups, depending on the backup schedule.

6. Back up the archive logs. The following command backs up all the archive logs currently known in the control file that have not been backed up to disk yet. After the backup of the archive logs is finished, the control file is updated.

```
BACKUP ARCHIVELOG ALL NOT BACKED UP 1 TIMES;
```

7. Schedule the snapshot on the SolidFire storage to run right after step 6 for all the volumes that are part of the RMAN file system.

Use Case for RMAN Merge

- Users can switch to the RMAN merge file, recover the database to a desired point in time, and significantly reduce the recovery time.
- Users can also use a disk-based image copy.
- The snapshots that are taken on the RMAN image can be used to create clone copies of the database easily. The clone copy of the database is created using the final merge image, which has all the changes from the last level 1 backup operation.
- The archive logs can also be applied to the clone copy to apply the latest changes from production.

A sample output of the RMAN target file system is provided in the appendix.

10 Database Cloning

Creating usable, space-efficient, and time-efficient database copies quickly and with virtually no effect on the production system is important. SolidFire volume cloning is a proven technology that helps database and system administrators deliver a near-instantaneous, space-efficient, point-in-time copy of the production database. Traditional methods of copy processing pose various challenges, including system downtime and degraded performance during the cloning process. Also, a large amount of storage space is required to store each clone. The SolidFire volume cloning process is completed quickly, with virtually no performance effect on the production system.

10.1 Clone a Volume

To clone a volume, complete the following steps:

1. Log in to the Element UI.
2. Select Data Protection → Group Snapshots. Right-click the Actions button for the snapshots. In this case, we select Snap1, the snapshot taken in section 8.1.
3. The Clone Volumes from Group Snapshot window opens.

4. Click the gear icon and then select Clone Volumes from Group Snapshot.
5. Enter a prefix (`CL` in this example).
6. Provide the account and access info, depending on what level of access you require for the clone volumes.

Clone Volumes From Group Snapshot ✕

New Volume Clone Details

New Volume Name Prefix

CL

Account

Copy From Volume ⇕

Access

Read / Write ⇕

Group Snapshot Members

ID	Name	Size	Volume ID	Volume Name	Account	Access	Retain Until	Replication Enabled	Replicated
182717	snap1	2.0 TB	179723	ORADATA2	oracle	Read / Write	-	No	-
182718	snap1	2.0 TB	179724	ORADATA3	oracle	Read / Write	-	No	-
182716	snap1	2.0 TB	179722	ORADATA1	oracle	Read / Write	-	No	-
182719	snap1	2.0 TB	179725	ORADATA4	oracle	Read / Write	-	No	-
182720	snap1	500.0 GB	179726	ORALOG	oracle	Read / Write	-	No	-

Start Cloning   Cancel

7. Click Start Cloning.

8. The cloned volumes can be viewed in the volumes list.

Filter By: Name: CLO ✕   ＋   Clear All

	ID	▼Name	Account	Access Groups	Access	Used	Size	Snapshots	Min IOPS	Max IOPS	Burst IOPS
☐	179733	CLORALOG	oracle	oraVM-CL	Read / Write	32.51%	500.0 GB	0	15,000	99,000	100,000
☐	179732	CLORADATA4	oracle	oraVM-CL	Read / Write	82.49%	2.0 TB	0	15,000	99,000	100,000
☐	179731	CLORADATA3	oracle	oraVM-CL	Read / Write	82.49%	2.0 TB	0	15,000	99,000	100,000
☐	179730	CLORADATA2	oracle	oraVM-CL	Read / Write	82.49%	2.0 TB	0	15,000	99,000	100,000
☐	179729	CLORADATA1	oracle	oraVM-CL	Read / Write	82.49%	2.0 TB	0	15,000	99,000	100,000

9. After the clone process is complete, add the newly created volumes to the volume access group of the database server (target) where you are planning to host the cloned copy.

10. Log in to the target host.

11. To rescan the iSCSI devices to present the newly cloned volumes, run the following commands:

```
iscsiadm -m discovery -t sendtargets -p <SolidFire SVIP> --op update –n node.session.nr_sessions
-v 2 -> where "n" is the number of paths
iscsiadm -m node -L all
multipath -r
```

12. Set the permission for the Oracle user or the grid user.

13. Mount the file systems or start the ASM instance.

14. Log in to the Oracle instance.

```
startup mount;
recover database;
alter database open;
```

Summary

NetApp SolidFire offers an optimal storage system for Oracle Database applications that is particularly well suited for web-scale architectures. SolidFire all-flash storage arrays coupled with an innovative architecture provide significant efficiency and agility to system planners who want to deploy database applications. By following the application use cases shown in this document, IT administrators can find the best SolidFire system for their business needs.

Appendix: RMAN Merge Output

```
-rw-r-----. 1 oracle oinstall 6597069774848 Mar 13 09:33 data_D-ORADB_I-2689361952_TS-SLOB_FNO-
5_09ruru1n.dbf
-rw-r-----. 1 oracle oinstall      17080320 Mar 13 09:33 0orv12rp_1_1.dbf
-rw-r-----. 1 oracle oinstall         40960 Mar 13 09:33 0prv12rp_1_1.dbf
-rw-r-----. 1 oracle oinstall       8839168 Mar 13 09:33 0nrv12rp_1_1.dbf
-rw-r-----. 1 oracle oinstall      56811520 Mar 13 09:33 0mrv12rp_1_1.dbf
-rw-r-----. 1 oracle oinstall       1671168 Mar 13 10:14 0trv157p_1_1.dbf
-rw-r-----. 1 oracle oinstall         40960 Mar 13 10:14 0urv157p_1_1.dbf
-rw-r-----. 1 oracle oinstall        286720 Mar 13 10:14 0srv157p_1_1.dbf
-rw-r-----. 1 oracle oinstall       3702784 Mar 13 10:14 0rrv157p_1_1.dbf
-rw-r-----. 1 oracle oinstall       5251072 Mar 13 11:43 data_D-ORADB_I-2689361952_TS-USERS_FNO-
6_0druru1n.dbf
-rw-r-----. 1 oracle oinstall     157294592 Mar 13 11:43 data_D-ORADB_I-2689361952_TS-
UNDOTBS1_FNO-4_0cruru1n.dbf
-rw-r-----. 1 oracle oinstall     838868992 Mar 13 11:43 data_D-ORADB_I-2689361952_TS-SYSTEM_FNO-
1_0bruru1n.dbf
-rw-r-----. 1 oracle oinstall    1027612672 Mar 13 11:43 data_D-ORADB_I-2689361952_TS-SYSAUX_FNO-
3_0aruru1n.dbf
-rw-r-----. 1 oracle oinstall       1900544 Mar 13 11:43 12rv1aep_1_1.dbf
-rw-r-----. 1 oracle oinstall         40960 Mar 13 11:43 13rv1aep_1_1.dbf
-rw-r-----. 1 oracle oinstall        294912 Mar 13 11:43 11rv1aeo_1_1.dbf
-rw-r-----. 1 oracle oinstall       4153344 Mar 13 11:43 10rv1aeo_1_1.dbf
-rw-r-----. 1 oracle oinstall         40960 Mar 13 13:08 0lrv12rp_1_1.dbf
-rw-r-----. 1 oracle oinstall         40960 Mar 13 14:02 0qrv157p_1_1.dbf
-rw-r-----. 1 oracle oinstall         40960 Mar 13 15:07 0vrv1aeo_1_1.dbf
```

Where to Find Additional Information

To learn more about the information that is described in this document, see the following documents and websites:

- Configuring SolidFire on Linux for Element
  https://www.netapp.com/us/media/tr-4639.pdf

- NetApp Product Documentation
  http://docs.netapp.com

- NetApp SolidFire Resources
  http://mysupport.netapp.com/solidfire/resources

Version History

Version	Date		Document Version History
Version 1.0	June 2014		Initial version
Version 2.0	June 2017		Refresh for Element 9.x
Version 3.0	January 2019		Refresh for Element 11.x

Refer to the Interoperability Matrix Tool (IMT) on the NetApp Support site to validate that the exact product and feature versions described in this document are supported for your specific environment. The NetApp IMT defines the product components and versions that can be used to construct configurations that are supported by NetApp. Specific results depend on each customer's installation in accordance with published specifications.