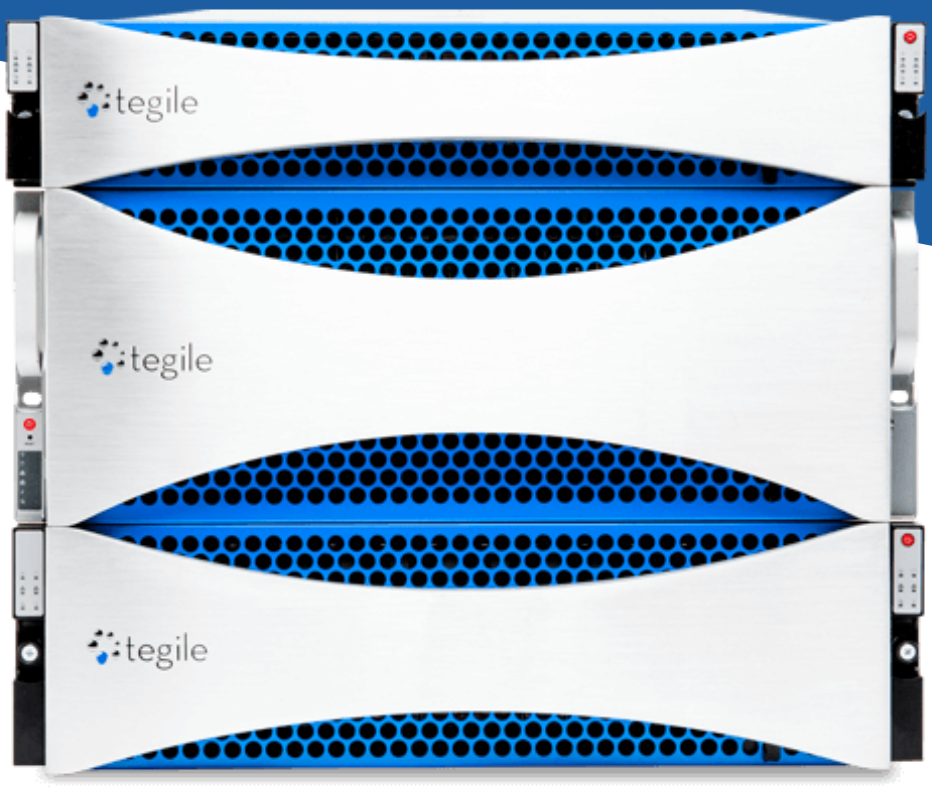
tegile
a Western Digital brand

Best Practices Guide

Oracle Database with Tegile IntelliFlash™ Storage



Overview

This document describes best practices for deploying Oracle Database with Oracle Automatic Storage Manager (ASM) and Oracle Linux on Tegile IntelliFlash arrays. For purposes of this document, Oracle Database 12*c* and Oracle Linux 7 with Unbreakable Enterprise Kernel (UEK) were used. However, these recommendations also apply to Oracle Database 11g and earlier versions of Oracle Linux. Any version-specific deviations in procedure are called out in the document.

Best practices are provided for the following areas:

- Tegile IntelliFlash Storage Array Provisioning
- Fiber Channel Network Configuration
- Oracle Linux Operating System Configuration
- Multipathing Customization
- Oracle Database Installation
- Oracle Database Grid Infrastructure Installation
- Oracle Automatic Storage Manager (ASM) Configuration
- Tegile Snapshots and Clones

Target Audience

This guide is intended for system administrators, storage administrators, database administrators, and implementation engineers who manage, monitor, or deploy database servers and storage solutions in the datacenter. It is recommended that those who will implement these best practices have familiarity with networking, storage systems, and Oracle database concepts and functionality.

Disclaimers

This document contains recommendations for building a generic Oracle system with a single database instance. It does not take into account requirements for security, performance, resilience, and other operational aspects that may vary for individual customer deployments. If recommendations in this document conflict with current operational guidelines, those existing policies should be given higher priority. Tegile accepts no liability for any issues resulting from following these recommendations.

Test Environment

The recommendations and settings in the guide were tested using the following configuration. If running an IntelliFlash version earlier that 3.7.x, please contact Tegile Support to verify that these best practices are compatible.

Software Environment

- Oracle Linux 7.3 x86_64 with the Unbreakable Enterprise Kernel 4.1.12-94.3.7.el7uek.x86_64
- Oracle Database 12*c* Enterprise Edition Release 12.2.0.1.0 – 64 bit Production
- Oracle Grid Infrastructure 12*c* Release 1 (12.2.0.1.0) for Linux x86-64

Hardware Environment

Figure 1 - Tegile IntelliFlash T4700 All-Flash Storage Array

1 x Tegile IntelliFlash T4700 All-Flash Storage Array with

- 4 x Intel® Xeon® E5 CPUs
- 464 GB memory
- 24 x 1 TB SSDs
- IntelliFlash version 3.7.0.0.170808(GA)

Tegile IntelliFlash Storage Array Provisioning

The following recommendations are made for optimal performance when provisioning Tegile storage for Oracle Database environments:

1. Use 2-way mirrored storage pools of equal size with active/active configuration. This provides redundancy with greater performance by taking advantage of the full capabilities of both controllers in the array.

2. Enable Tegile data compression. Tegile compression technology is a powerful way to reduce the overall storage capacity requirements for Oracle environments. The LZ4 compression algorithm is recommended for database workloads.

3. Due to the nature of Oracle Database blocks and the underlying data, Oracle deployments are not well suited for data deduplication. Significant savings are not realized in Oracle environments, so data deduplication is not recommended.

4. Larger block sizes can yield better performance. At a minimum, an 8 KB block size is recommended. However, this recommendation is workload-specific. Depending on the workload, a 16 KB or 32 KB block size may be considered (especially for REDO logs). Modify the Oracle DB_BLOCK_SIZE parameter to match the LUN block size selected. For purposes of our testing, a 16 KB block size was found to produce optimal performance for an OLTP workload.

5. Use thin provisioning when creating LUNs for Oracle ASM disk groups. Thin provisioning dynamically allocates storage capacity on the array as it is needed. This avoids performance problems associated with wasted capacity when over-allocating storage for future needs.

Listed below are the Tegile IntelliFlash configuration screens to implement these recommended settings.

Pool Configuration:

- Active/Active
- 2-way Mirror Redundancy
- Use Half of Total Disks

Project Configuration:

- Generic Purpose
- Fibre Channel Access Protocol
- Data Deduplication Off
- LZ4 Data Compression

LUN Configuration:

- Thin Provisioning
- Database Purpose
- 16 KB Block Size
- Fibre Channel Access Protocol

Two pools should be created with Active/Active configuration using 2-way mirroring. Each pool is assigned half of the total number of disks.

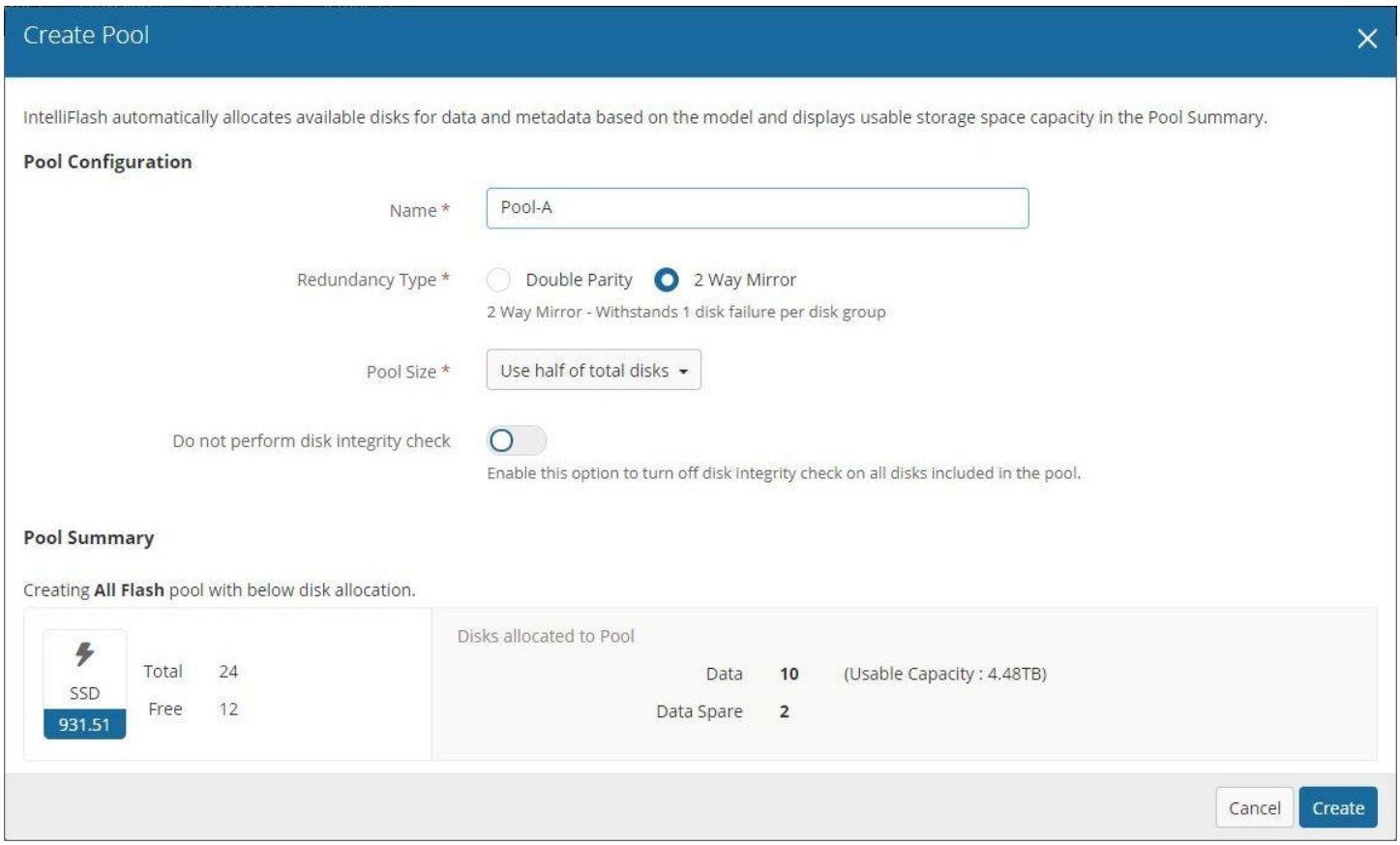


Figure 2 - Pool Configuration Settings

Each project is configured as Generic Purpose, using the Fibre Channel access protocol.

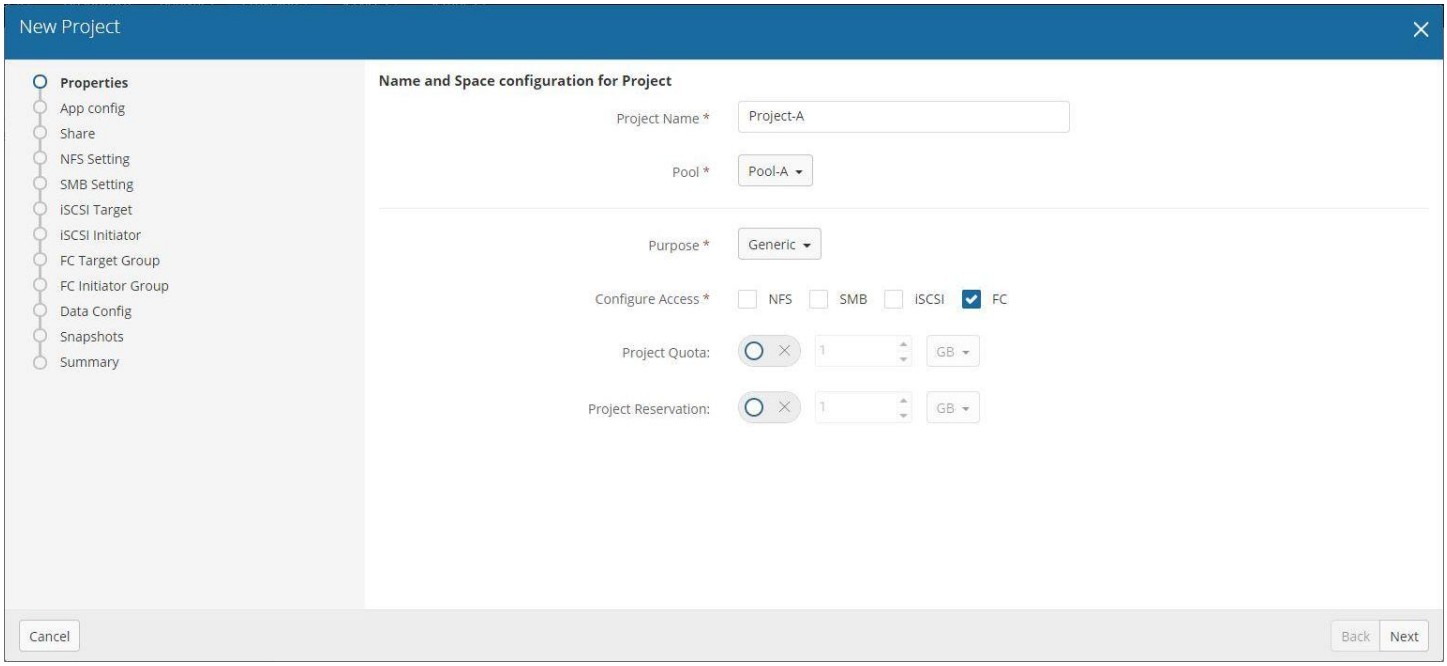


Figure 3 - Project Configuration Settings

Each project is configured with LZ4 compression. Deduplication is not enabled for Oracle Database workloads.

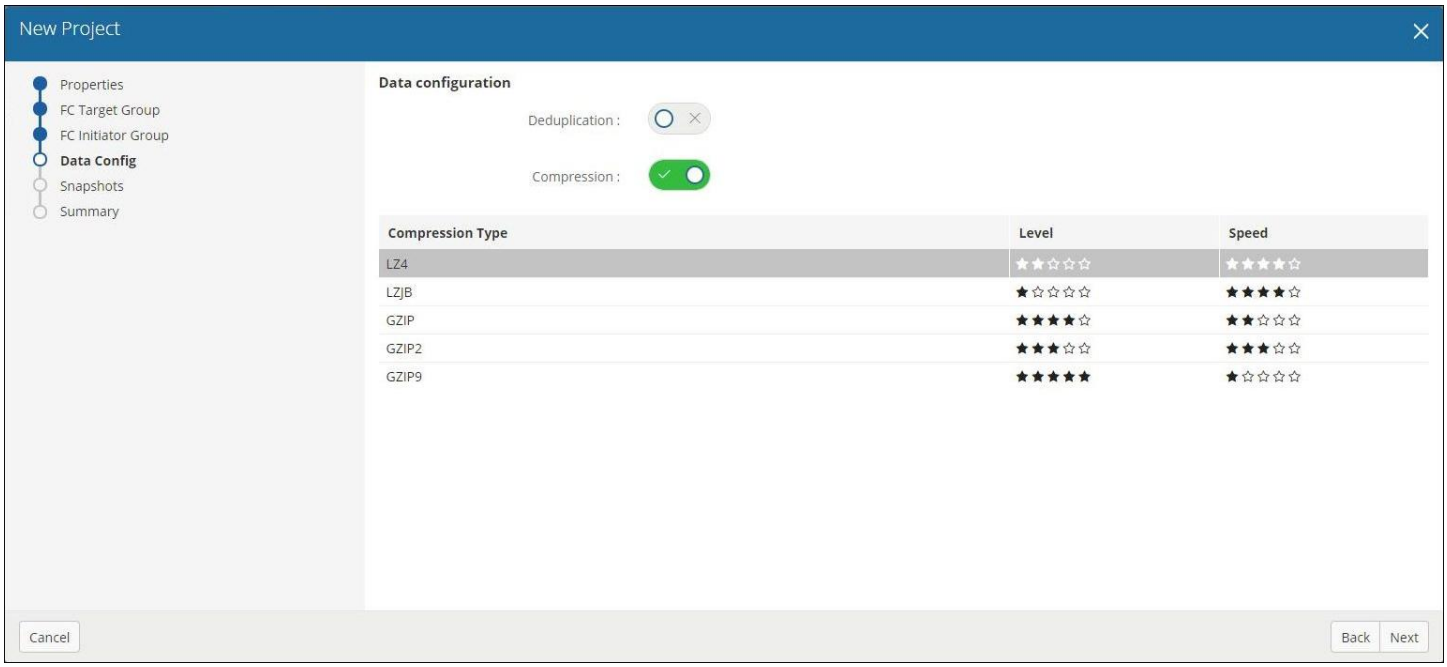


Figure 4 - Project Configuration Settings

Each LUN is created with thin provisioning, using with a 16 KB block size and the fibre channel access protocol.

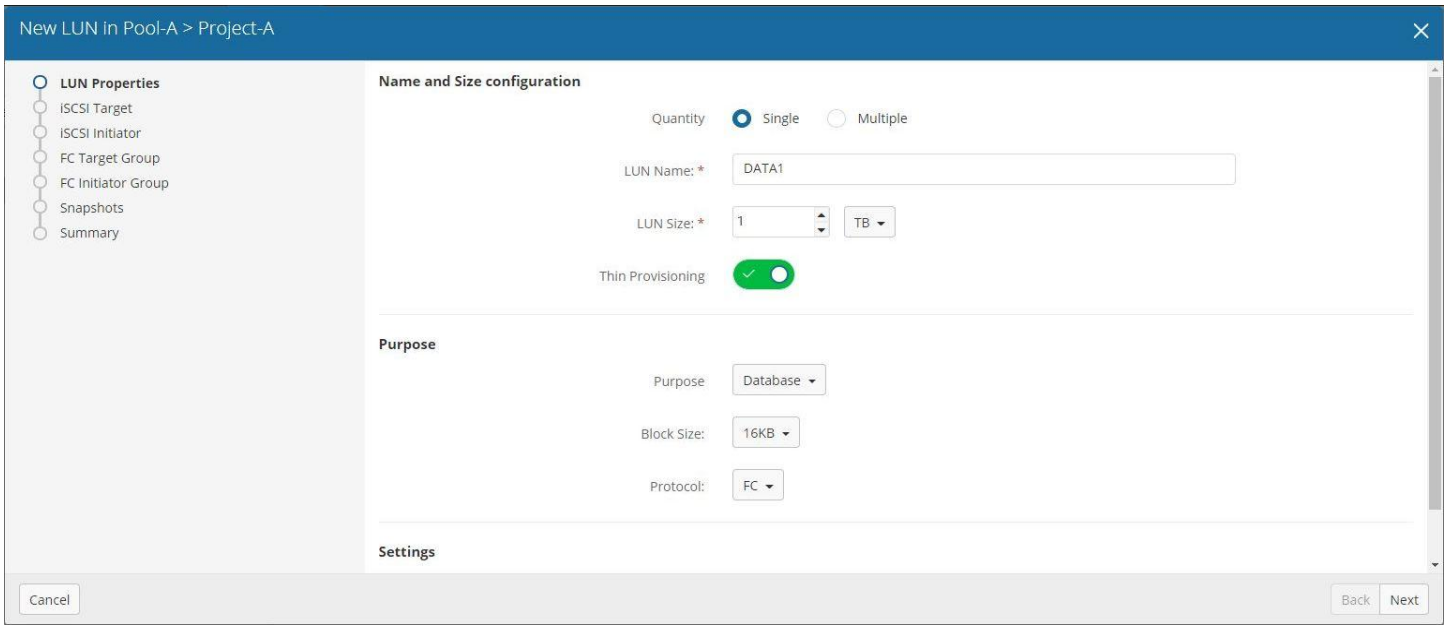


Figure 5 - LUN Configuration Settings

Fiber Channel Network Configuration

Tegile recommends that a database server connected to an array with fibre channel always utilize redundant fabrics to protect against hardware failure, while providing load balancing and superior performance. In the diagram below, a FC configuration with redundant fabrics is shown connecting a database server to a Tegile all-flash array.

Redundant FC fabrics can be used with a Tegile array in both an Active/Passive and Active/Active pool configuration. The fabric configuration of a Tegile storage array in an Active/Active configuration is generally the same as the Active/Passive.

With Active/Passive configuration, one of the array controllers hosts all I/O traffic as the 'Active' controller until it is either manually failed over to the 'Standby' controller, or automatically failed over due to a hardware or link failure.

Active/Active is a common configuration to use with Tegile All-Flash arrays to obtain maximum performance as well as high density. In this configuration, all paths of the server's dual HBAs are utilized in connecting to the storage array. Should a hardware failure occur on either controller in the array, all Pools will be migrated to the surviving controller with all four ports in the database server host still being utilized for I/O traffic.

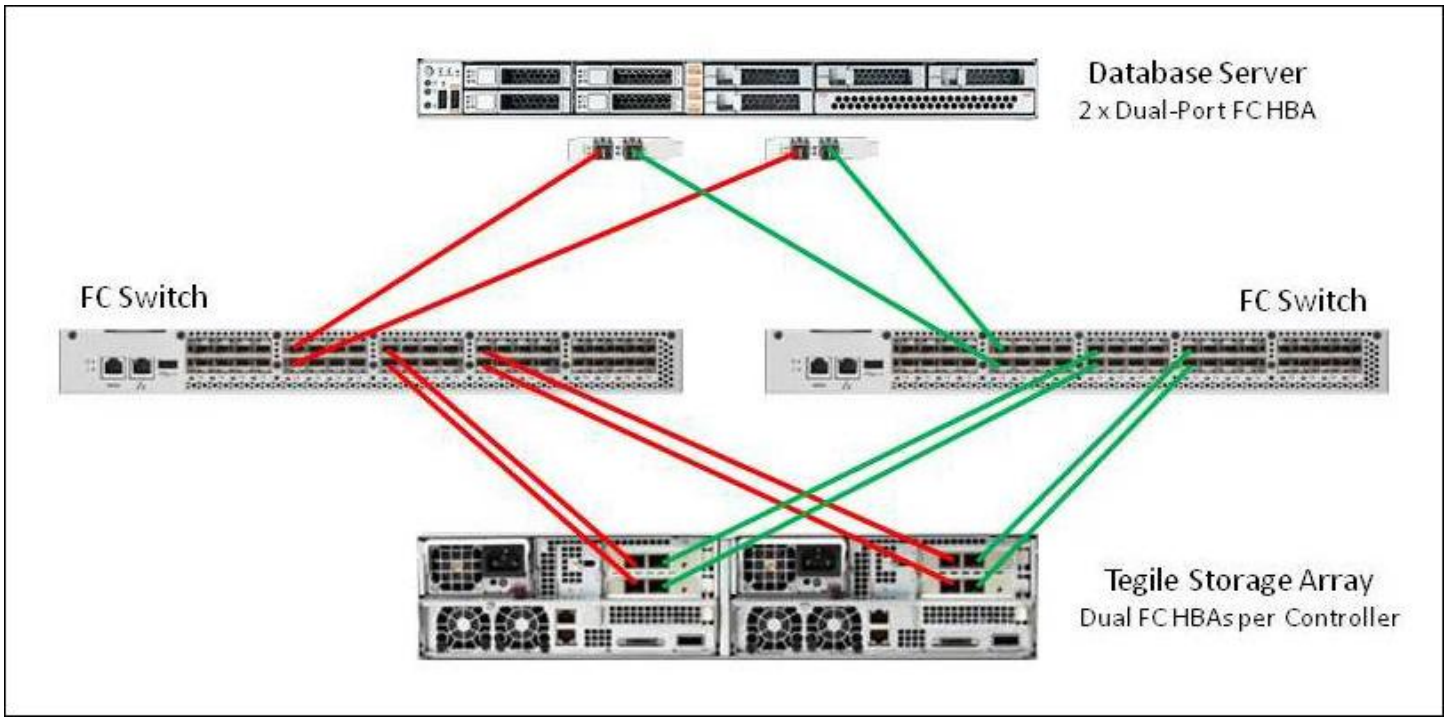


Figure 2 - Fibre Channel Network Configuration

Tegile recommends a FC switch configuration with 1:1 (one-to-one) zoning when connecting an array to a FC infrastructure. With 1:1 zoning, there is a single target WWPN and a single initiator WWPN in each zone configured on the FC switch. This type of zoning necessitates many zones being created, but reduces the impact of RSCNs (Registered State Change Notifications) and makes troubleshooting link problems easier.

With 1:1 zoning, each FC Initiator port on the database server is zoned with a single corresponding FC target port on the Tegile array. Using the above Active/Active FC Configuration as an example, each FC Initiator port in the database server is connected to an FC switch that is also directly connected to 4 FC Target ports on the Tegile array. In this case, each Initiator port would require 4 zones, one to each of the target ports connected to the same switch. A total of 16 zones would be required for a full 1:1 zoning configuration for a single host.

Oracle Linux Operating System Configuration

Tegile recommends the following steps to install and configure the Oracle Linux operating system.

Download the Oracle Linux installation package from the Oracle Software Delivery Cloud (http://edelivery.oracle.com). For purposes of this document, Oracle Linux Release 7 Update 3 for x86 (64 bit) was used.

Install Oracle Linux by booting the system from DVD or the ISO installation file, and follow the prompts. Consult the Oracle Linux Installation Guide for Release 7 document for detailed installation steps.

After the Oracle Linux operating system installation is complete, the following Oracle-specific packages should also be installed:

```
oracle-database-server-12cR2-preinstall
oracleasm
oracleasm-support
kmod-oracleasm
```

Note:

For Oracle Database 12cR2, install package oracle-database-server-12cR2-preinstall
For Oracle Database 12cR1, install package oracle-rdbms-server-12cR1-preinstall
For Oracle Database 11gR2, install package oracle-rdbms-server-11gR2-preinstall

The following additional packages are also required:

```
binutils
compat-libcap1
compat-libstdc++
device-mapper-multipath
gcc
gcc-c++
glibc
glibc-devel
ksh
libgcc
libstdc++
libstdc++-devel
libaio
libaio-devel
libXScrnSaver
make
nfs-utils
ntp
openssh-server
sysstat
telnet
telnet-server
xinetd
xorg-x11*
```

For optimal performance, Oracle Linux should be configured with the following kernel settings in /etc/sysctl.conf as follows. These setting may vary depending on the physical memory of your database server.

```
fs.file-max = 6815744
kernel.sem = 250 32000 100 128
kernel.shmmni = 4096
kernel.shmall = 1073741824
kernel.shmmax = 4398046511104
kernel.panic_on_oops = 1
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
net.ipv4.conf.all.rp_filter = 2
net.ipv4.conf.default.rp_filter = 2
fs.aio-max-nr = 1048576
net.ipv4.ip_local_port_range = 9000 65500
```

The following lines should be added to the file /etc/security/limits.conf to set shell limits for user oracle:

```
# shell limits for user oracle 12gR2
oracle   soft      nofile    1024
oracle   hard      nofile    65536
oracle   soft      nproc     2047
oracle   hard      nproc     16384
oracle   soft      stack     10240
oracle   hard      stack     32768
oracle   soft   memlock   3145728
oracle   hard  memlock   3145728
```

Multipathing Customization

Tegile recommends that Linux multipathing be customized as follows:

Verify that Linux multipathing is installed.

As user root:

```
# yum list device-mapper
```

Verify that Linux multipathing is enabled.

As user root:

```
# multipath -ll
```

If needed, start Linux multipathing agent.

As user root:

```
# service multipathd start
```

Create the /etc/multipath.conf file with the following lines:

```
defaults {
        polling_interval        5
        path_grouping_policy    multibus
        failback                immediate
        user_friendly_names     yes
        max_fds                 8192
        }
devices {
        device {
                vendor  "TEGILE"
                product "INTELLIFLASH"
                hardware_handler        "1 alua"
                path_selector           "round-robin 0"
                path_grouping_policy    "group_by_prio"
                no_path_retry           10
                dev_loss_tmo            50
                path_checker            tur
                prio                    alua
                failback                30
                rr_min_io               128
                }
        }
mutipaths {
        multipath {
                wwid xxxxxxxxxx (substitute WWID of specific LUN here)
                alias DATA1
                }
        multipath {
                wwid xxxxxxxxxx (substitute WWID of specific LUN here)
                alias DATA2
                }
        }
```

After creating the /etc/multipath.conf file, flush the device mapper and rescan.

As user root:

```
# multipath -F
# multipath -v2
```

-or-

```
# reboot
```

Verify that LUNs appear with correct multipathing and aliases as specified in /etc/multipath.conf file.

As user root:

```
# multipath -ll
```

Oracle Database Installation

The following are recommended steps to install and configure Oracle Database software.

Download the Oracle Database installation package from the Oracle Software Delivery Cloud (http://edelivery.oracle.com). For purposes of this document, Oracle Database Enterprise Edition 12.2.0.1.0 for Linux x86-64 was used.

If they do not already exists, run the following commands to create the Oracle user and groups.

As user root:

```
# groupadd dba
# groupadd oinstall
# useradd -m -g oinstall -G dba oracle
# passwd oracle
```

Create directories for the Oracle Database installation. For purposes of this document,

ORACLE_BASE=/oracle/12c

ORACLE_HOME=/oracle/12c/database

As user root:

```
# mkdir -p /oracle/12c/database
# chown -R oracle:oinstall /oracle
# chmod -R 775 /oracle
```

Install the Oracle Database software by first unzipping the installation file. Then run the Oracle Database installer and follow the prompts. Consult the Oracle Database 12c Release 2 Installation Guide document for detailed installation steps.

As user oracle, set Oracle environment variables to Oracle home. Then run the Oracle Database installer.

```
# ./runInstaller.sh
```

Oracle Database Grid Infrastructure Installation

The following steps are recommended to install and configure Oracle Grid Infrastructure software.

Download the Oracle Database Grid Infrastructure installation package from the Oracle Software Delivery Cloud (http://edelivery.oracle.com). For purposes of this document, Oracle Database Grid Infrastructure 12.2.0.1.0 for Linux x86-64 was used.

Create directories for the Oracle Database Grid Infrastructure installation. For purposes of this document,

ORACLE_BASE=/oracle/12c

ORACLE_HOME=/oracle/grid

As user root:

```
# mkdir -p /oracle/grid
# chown -R oracle:oinstall /oracle
# chmod -R 775 /oracle
```

Install the Oracle Database software by first unzipping the installation file. Then run the Oracle Database Grid Infrastructure installer and follow the prompts. Consult the Oracle Grid Infrastructure Installation Guide for detailed installation steps.

As user oracle, set the Oracle environment variables to Oracle Grid home. Then run the Oracle Grid installer.

```
# ./gridSetup.sh
```

Oracle Automatic Storage Manager (ASM) Configuration

For simplicity and performance, it is recommended that Oracle ASM be used for deploying Tegile storage with an Oracle Database environment. The following configuration recommendations are made:

1. Create Oracle ASM disk groups specifying external redundancy. Tegile storage pools are created with built-in redundancy. Therefore, additional overhead from specifying Oracle ASM redundancy should be avoided.

2. Create separate Oracle ASM disk groups for DATA (for tablespaces and temp), LOGS (for redo logs), and FRA (for Fast Recovery Area). Redo logs store all changes made to the database as they occur. The Fast Recovery Area is a Oracle-managed central storage location for back and recovery files. Separate Oracle ASM disk groups for redo logs and Fast Recovery Area should be used to provide for the segregation of disks and redundancy in case a database recovery is required.

3. Add LUNs to Oracle ASM disk groups in pairs, balanced between controllers and storage pools. When adding pairs of LUNs to an Oracle ASM disk group, create one LUN from Pool A and one LUN from Pool B to take advantage of the performance benefits of Tegile active/active controller technology.

4. When creating an Oracle ASM disk group, eight LUNs of equal size per disk group is recommended. Tests show that this can result in greater performance and lower latency.

5. To avoid the unnecessary overhead of Oracle ASM rebalancing, increase the size of existing LUNs rather than add new LUNs to Oracle ASM disk groups.

For deployment of Oracle ASM with Tegile storage, first verify that Oracle ASM is installed.
As user root:

```
# yum list oracleasm
```

If needed, install Oracle ASM packages.
As user root:

```
# yum install oracleasm
# yum install oracleasm-support
# yum install kmod-oracleasm
```

Assign ASM labels to LUNs using multipathing aliases.

As user root:

```
# /etc/oracleasm createdisk DATA1 /dev/mapper/DATA1
# /etc/oracleasm createdisk DATA2 /dev/mapper/DATA2
```

As user oracle, set the Oracle environment variables to Oracle Grid home.

Then run the ASM Configuration Assistant to create ASM disk groups.

```
# asmca
```

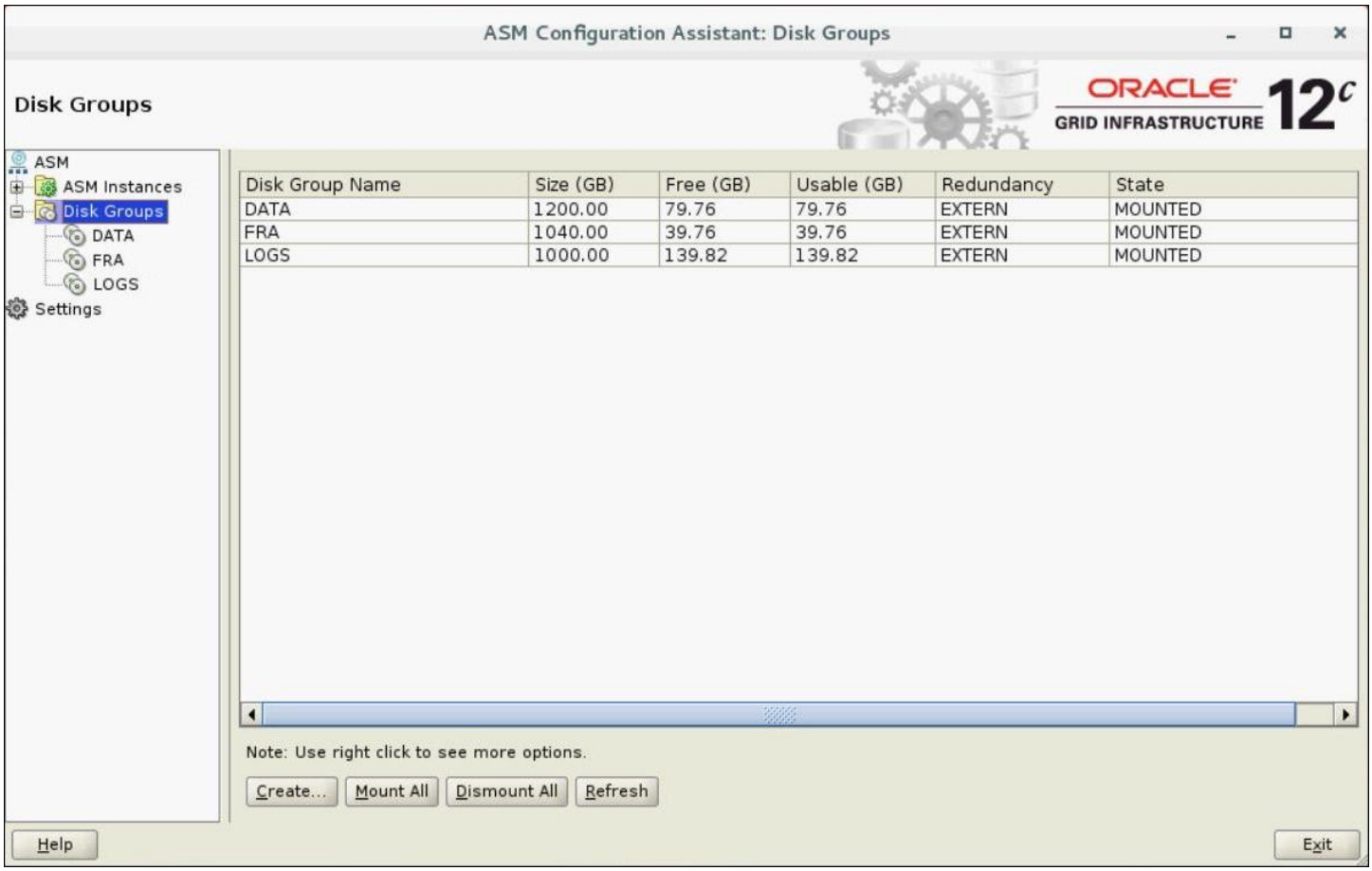


Figure 6 - ASM Configuration Assistant

Note: For Oracle Database versions below 12.1, a known issue exists with ASM on Solaris that could be observed when cloning Tegile snapshots. This is related to Oracle bug 12382627 Solaris: ORA-27063 "number of bytes read/written is incorrect". The fix for this problem is to upgrade Oracle Grid Infrastructure to version 11.2.0.4 or higher. This is relevant to Solaris 64-bit (SPARC) and Sun Solaris x86-64 environments only.

Snapshots and Clones

This section discusses best practices for managing Tegile snapshots and clones for an Oracle Database environment. This should not be confused with Oracle snapshots.

Tegile snapshots are widely used to quickly create point-in-time virtual copies of data. However, it is incorrect and dangerous to think of snapshots as a backup solution. Unless copied to different media such as a second storage array, Tegile snapshots do not protect against media failure.

Tegile snapshots can be helpful for implementing mass deployments or offline computing to ease compute and network loads. The typical use for Tegile snapshots in an Oracle Database environment is to replicate a target master (i.e. production) to a non-master (i.e. development), or to offload and cache expensive queries in a read-only data warehouse.

Another reason why Tegile snapshots are such a valuable administrative tool is because they take up virtually no additional space due to redirect on write technology. Space is only consumed by new data or pointers to old data when existing data is changed.

Before creating a Tegile snapshot of an Oracle Database environment, the database must first be put in backup mode.
As user oracle:

```
# sqlplus / as sysdba
SQL> alter database begin backup;
```

Note that Oracle backup mode is possible only with ARCHIVELOG enabled. NOARCHIVELOG is the default setting, but it does not provide for point-in-time snapshots. If NOARCHIVELOG is enabled, the database must be shutdown cleanly before creating a snapshot to ensure consistency.

The following commands will show the log mode (either ARCHIVELOG or NOARCHIVELOG) for a database.
As user oracle:

```
# sqlplus /nolog
SQL> connect / as sysdba;
connected.
SQL> archive log list;
SQL> select log_mode from v$database;
```

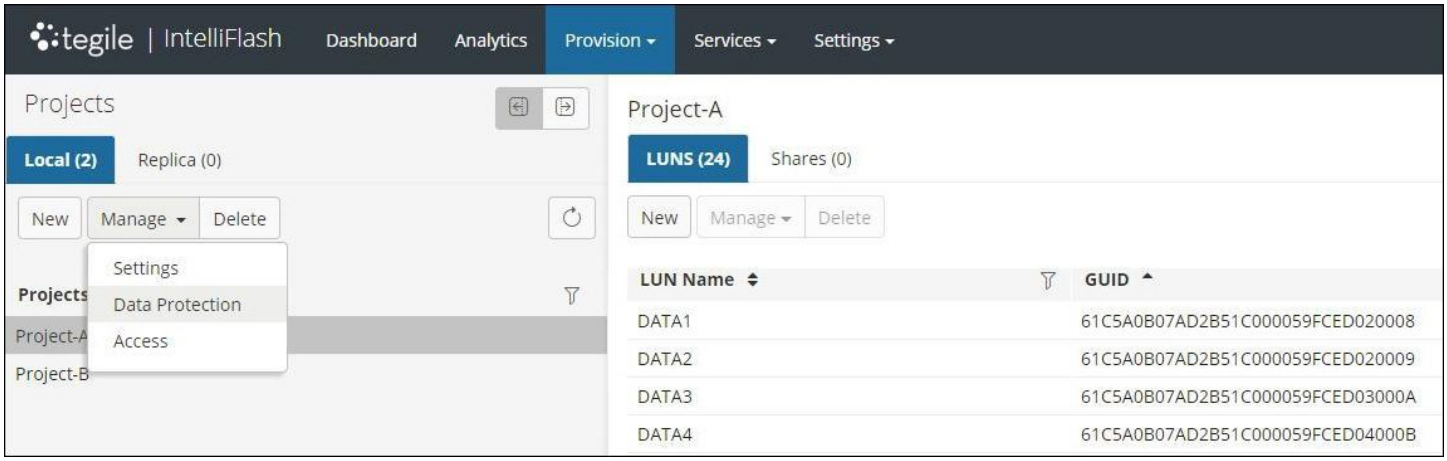
The log setting for a database can be changed from NOARCHIVELOG to ARCHIVELOG with the following commands. However, this requires that the database be shutdown first.

As user oracle:

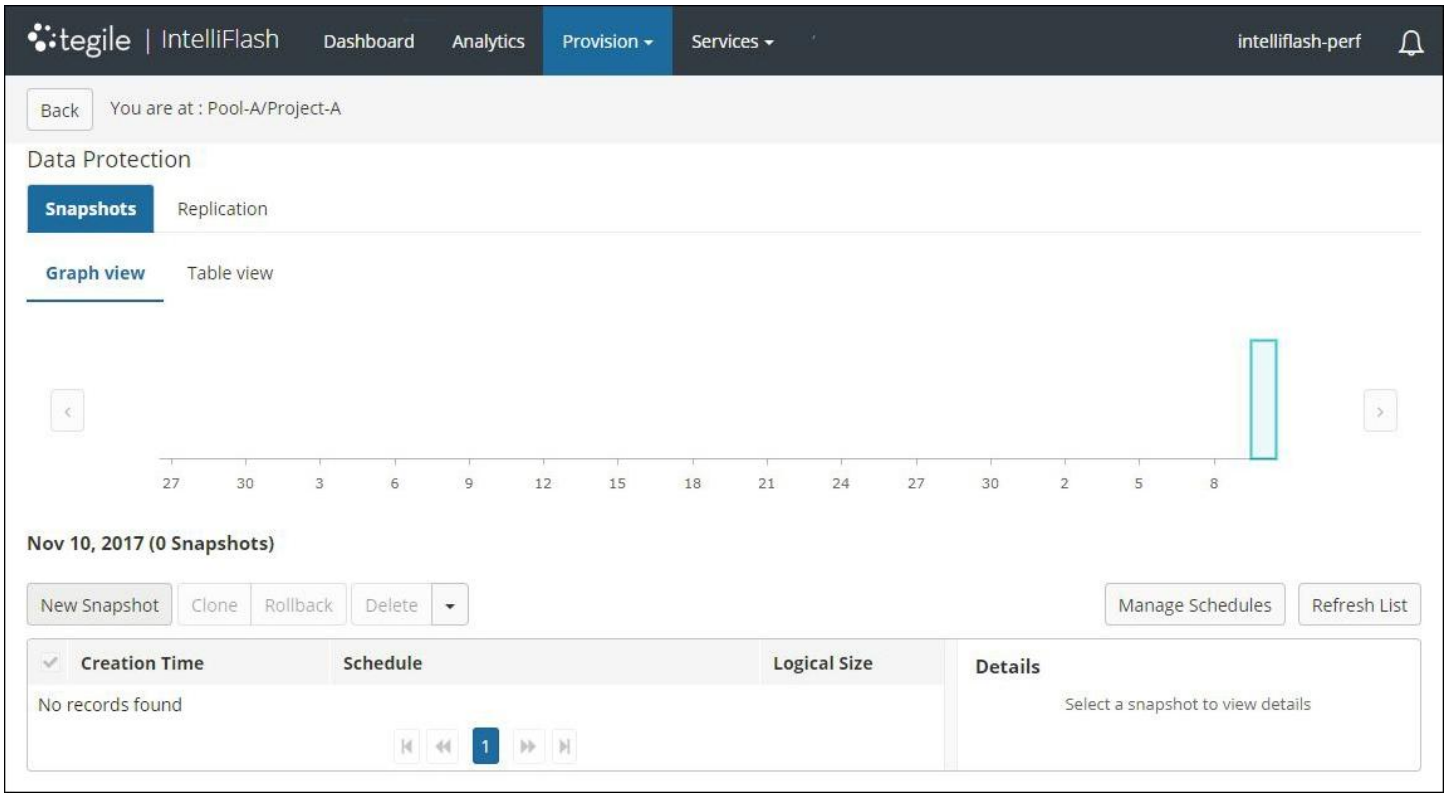
```
SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup mount;
ORACLE instance started.
Database mounted.
SQL> alter database archivelog;
Database altered.
SQL> alter database open;
Database altered.
```

After the database has been put in backup mode (with ARCHIVELOG) or shutdown (with NOARCHIVELOG), a Tegile snapshot can then be created. Tegile projects are convenient organizational structures that holds multiple LUNs. Snapshots are created at the project level and contain a point-in-time virtual copy of **all data** in that project.

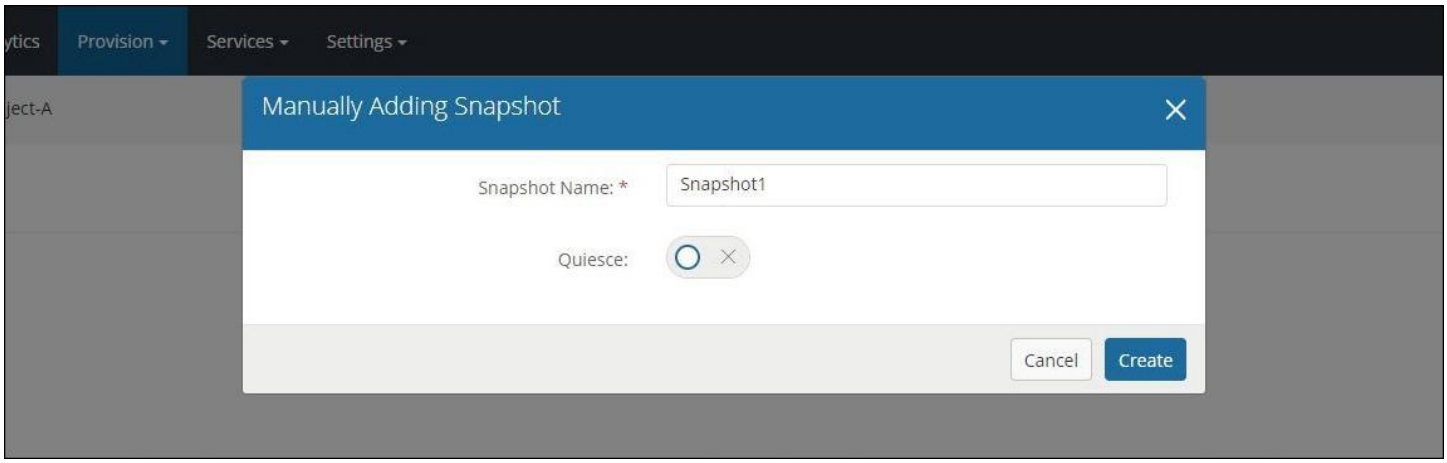
The following screens illustrate the simple process for creating a Tegile snapshot of *ProjectA*.



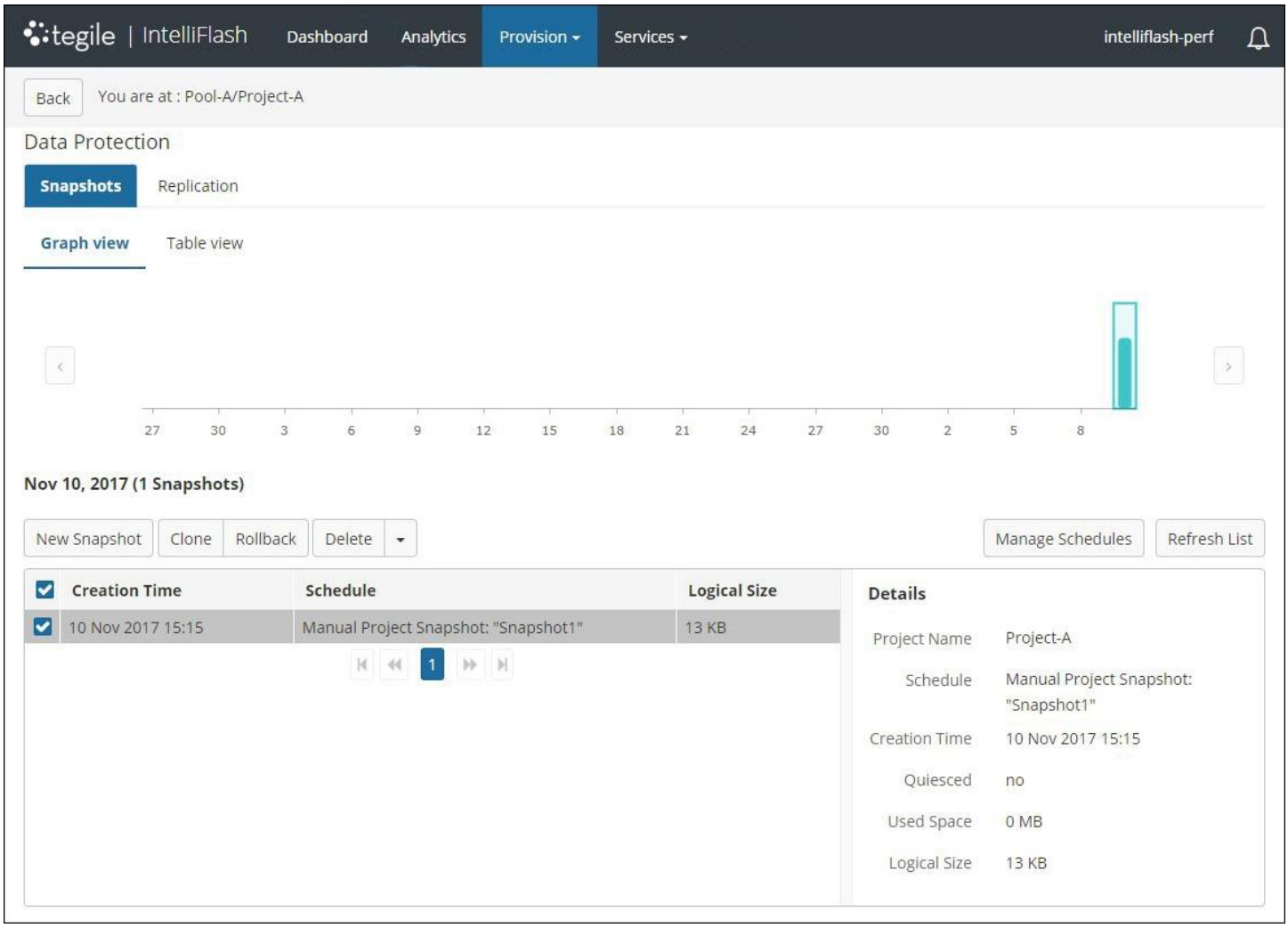
Provision --> Projects --> Manage --> Data Protection



New Snapshot



Enter Snapshot Name



New Snapshot is Created

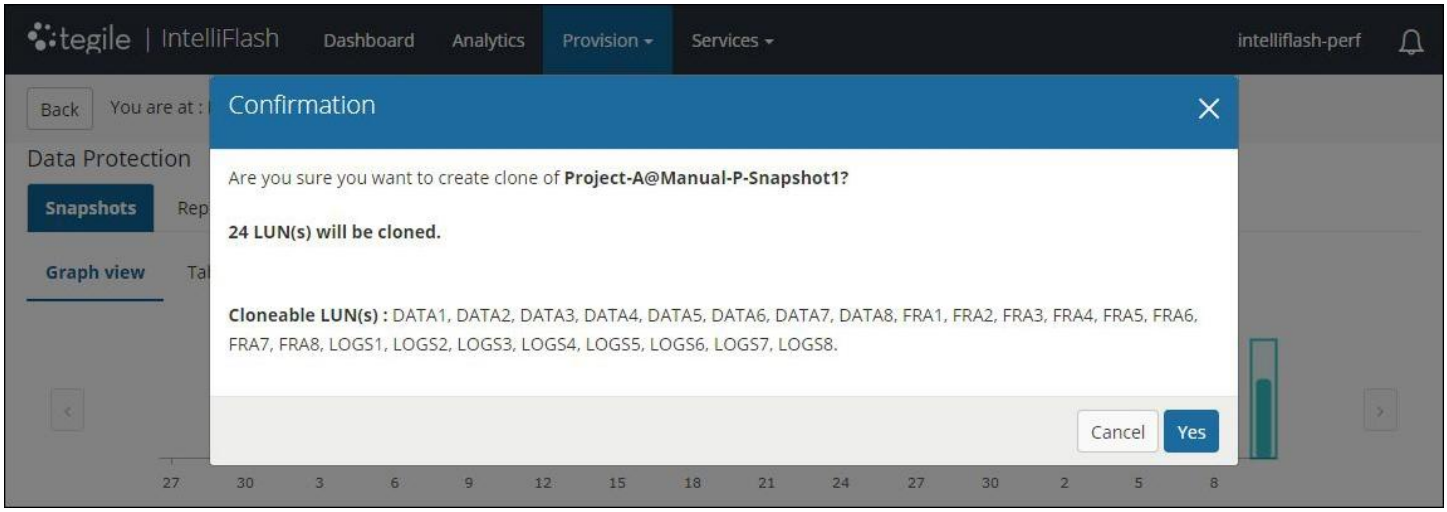
Creation of a Tegile snapshot is a very fast operation. After the snapshot has been created, the database should then be taken out of backup mode to resume normal operation.

As user oracle:

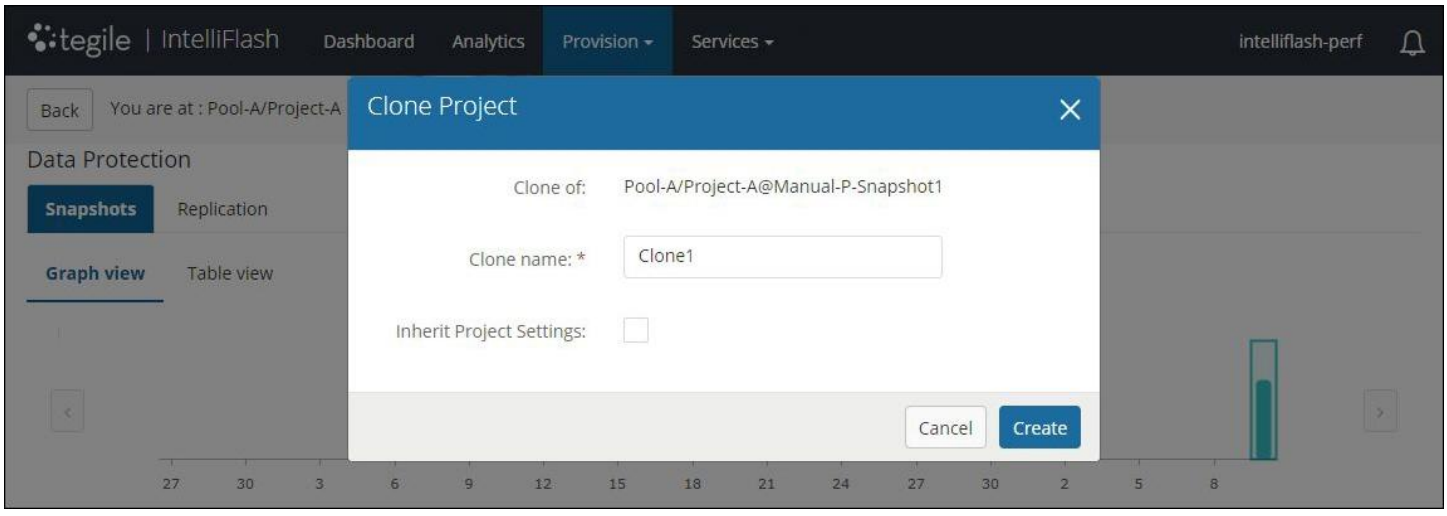
```
# sqlplus / as sysdba
SQL> alter database end backup;
```

After a Tegile snapshot has been created, it can be cloned to replicate every LUN in the project while maintaining the original LUNs for continuous use. The cloned LUNs can be assigned alternate access for recovery, data mining, or test and development purposes. When finished, the clone can be deleted without any disruption to the original data and LUNs.

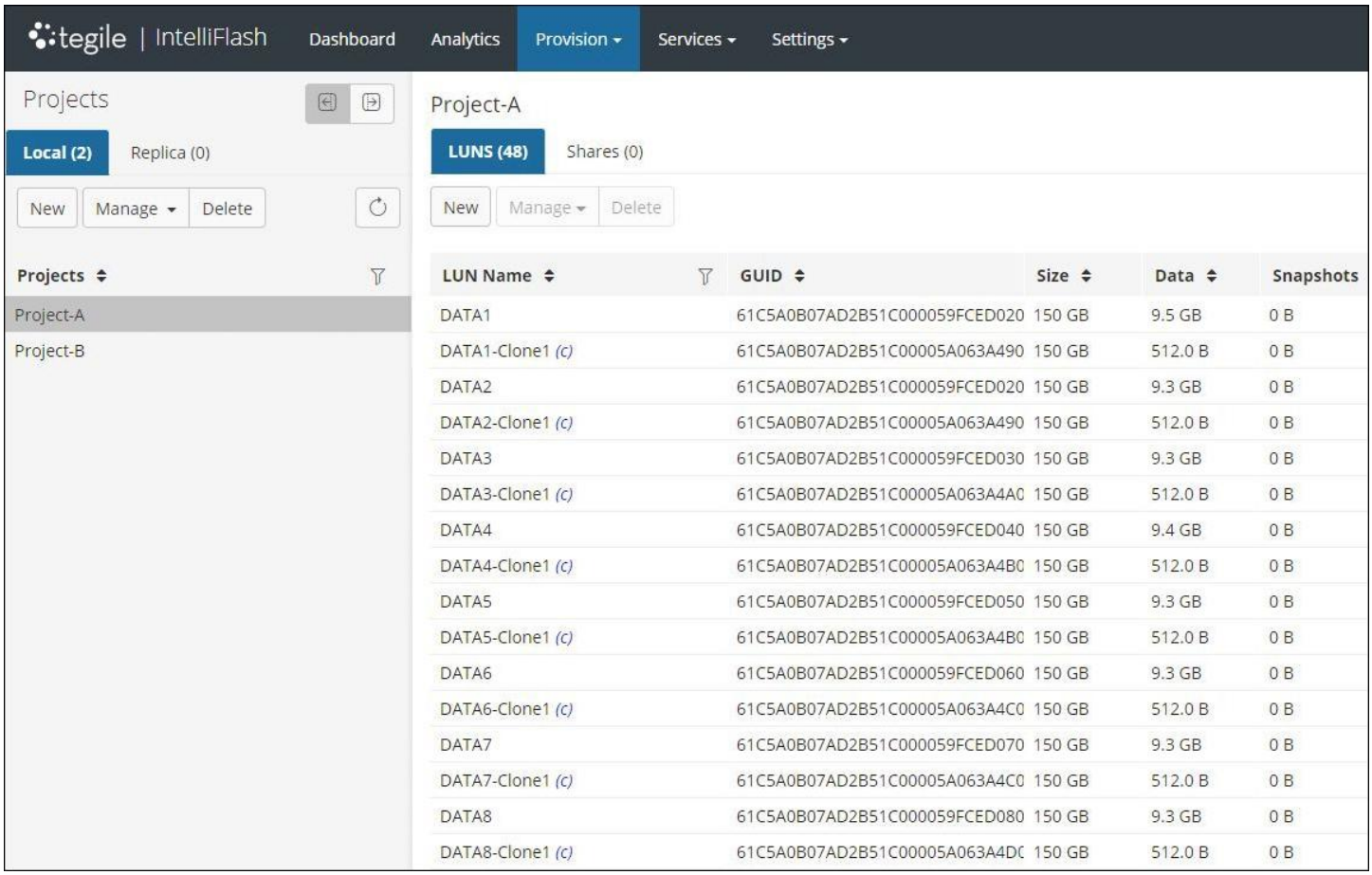
The following screens illustrate the simple process for creating a clone of *Snapshot1*.



New Clone



Enter Clone Name



Cloned LUNs are Created

tegile
a Western Digital brand