Dell Storage Center with Solaris 11 Best Practices

Daniel Tan, Product Specialist
Dell Storage Applications Engineering

May 2015

Revisions

Date	Revision	Description	Author
Oct 2013	1.0	Initial release	Daniel Tan
May 2015	1.1	Procedures updated regarding fcp/fp timeout configuration	Daniel Tan

THIS WHITE PAPER IS FOR INFORMATIONAL PURPOSES ONLY, AND MAY CONTAIN TYPOGRAPHICAL ERRORS AND TECHNICAL INACCURACIES. THE CONTENT IS PROVIDED AS IS, WITHOUT EXPRESS OR IMPLIED WARRANTIES OF ANY KIND.

Table of contents

Executive summary

Oracle Solaris is an enterprise-class operating system; it can be extremely robust and scalable and with this also brings with it a large degree of complexity as well. There are many features of the Solaris OS which provides no relevance nor impact to its use with the Dell Storage Center.

This paper outlines, highlights and discusses the various topics where the Solaris OS can be configured and tuned to provide an optimized experience in use with the Dell Storage Center products.

Purpose

This paper provides an overview of Solaris tools and utilities and introduces best practice guidelines for configuring volume discovery, multipath, file system and queue depth management and more; when in use with the Dell Storage Center.

It is important to note that as is common with most UNIX, Linux and in this case, Solaris platforms, there are often various methods in which to accomplish the tasks as discussed in this paper. This guide does not contain every possible method nor is it intended to be the definitive guide for all scenarios. This document is intended as a starting point of reference for end users and system administrators.

Also note that this guide will focus almost exclusively on the CLI (Command Line Interface). Many UNIX and Linux distributions have created graphical tools to achieve many of these tasks; this guide simply focuses on the CLI because it is often the most universally applicable across UNIX distributions.

1 Overview

The Dell™ Storage Center™ provides Solaris-compatible and SCSI-3 compliant disk volumes to Solaris which removes much of the complexity of allocating and administering, using and protecting the mission critical data found on most Solaris-based hosts.

A properly configured Storage Center can remove the need for cumbersome physical disk configuration exercises and management along with complex RAID configuration mathematics. The use of the Storage Center removes the need to mirror volumes on the Solaris OS layer since the Storage Center is already providing RAID 10 speed and reliability at the storage array layer.

The full range of Solaris utilities such as mirroring, backup, multiple file system types, multipath, boot from SAN and disaster recovery can be used with Dell Storage Center volumes.

2 Useful Tools

The Dell Storage Center does not require the installation of any additional software or tools beyond what is provided with the base Operating System. There are several OS utilities that will be leveraged in order to manage volumes presented to a Solaris host. These commands/utilities are discussed below.

Please refer to the Oracle Solaris 11 Information Library URL below.
http://docs.oracle.com/cd/E23824_01/

2.1 format

This is the disk partitioning and maintenance utility for Solaris on SPARC. This tool is used to label the newly provisioned volumes and create partitions. For ZFS-based volumes, it is a Storage Center recommendation to use entire disk layouts (for ZFS pools and file systems) instead of creating partitions.

```
root:~# echo | format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
       0. c0t5000C50048697F5Fd0 <SUN300G cyl 46873 alt 2 hd 20 sec 625>  solaris
          /scsi_vhci/disk@g5000c50048697f5f
          /dev/chassis//SYS/HDD0/disk
Specify disk (enter its number): Specify disk (enter its number):
```

2.2 fcinfo

The fcinfo command is used to gather information about Fibre Channel HBA devices installed in the Solaris system. This command is useful when needing to cross-reference the WWN reported on the host with the WWN reported for the HBA in the Dell Storage Center.

```
root:~# fcinfo hba-port
HBA Port WWN: 21000024ff3ebd24
        Port Mode: Initiator
        Port ID: df2a00
        OS Device Name: /dev/cfg/c2
        Manufacturer: QLogic Corp.
        Model: 371-4325-02
        Firmware Version: 5.6.4
        FCode/BIOS Version:  BIOS: 2.02; fcode: 2.03; EFI: 2.01;
        Serial Number: 0402H00-1216023468
        Driver Name: qlc
        Driver Version: 20120717-4.01
        Type: N-port
        State: online
        Supported Speeds: 2Gb 4Gb 8Gb
        Current Speed: 4Gb
        Node WWN: 20000024ff3ebd24
        Max NPIV Ports: 255
        NPIV port list:
HBA Port WWN: 21000024ff3ebd25
        Port Mode: Initiator
        Port ID: dd1900
        OS Device Name: /dev/cfg/c3
        Manufacturer: QLogic Corp.
        Model: 371-4325-02
        Firmware Version: 5.6.4
        FCode/BIOS Version:  BIOS: 2.02; fcode: 2.03; EFI: 2.01;
        Serial Number: 0402H00-1216023468
        Driver Name: qlc
        Driver Version: 20120717-4.01
        Type: N-port
        State: online
        Supported Speeds: 2Gb 4Gb 8Gb
```

```
          Current Speed: 4Gb

          Node WWN: 20000024ff3ebd25

          Max NPIV Ports: 255

          NPIV port list:
```

2.3 devfsadm

The devfsadm command is used to manage the device tree listed in the /dev structure. This is used as part of volume management, specifically when it is needed to add and/or remove /dev links to volumes which have been removed from the system.

```
root:~# devfsadm [-Cv]
```

2.4 cfgadm

The cfgadm command is used to manage the configuration of devices in the Solaris system. In particular, when a volume is mapped or unmapped from the Dell Storage Center, cfgadm is used to scan the controller ports of the HBA to (un)discover volumes. The example below illustrates a typical command to scan the Fibre Channel HBA ports of a system to discover a newly mapped volume.

To discover any mapped volumes on device paths /dev/cfg/c2 and /dev/cfg/c3, run the following commands below.

```
root:~# cfgadm -c configure c2

root:~# cfgadm -c configure c3
```

2.5 luxadm

The luxadm command is used to manage specific aspects of a volume, such as to bring a volume offline or to gather detailed information about a given volume. The example below illustrates the usage of luxadm to bring a volume offline in preparation to have it removed from the Solaris host.

```
root:~# luxadm -e offline /dev/rdsk/c4t6000D310000069000000000000000F4Ed0s2
```

2.6 stmsboot

In Solaris 10U3 or newer and Solaris 11.x OS releases, the need of manually editing the /kernel/drv/fp.conf and /kernel/drv/mpt.conf files has been deprecated. These latter files may still be manually edited for advanced configuration needs; though it is now recommended that the following command toolset be used for administering and managing multipath on Solaris.

2.7 mpathadm

The mpathadm command is used to manage, administer and display information regarding the multipath state of the attached SAN volumes.

```
root:/kernel/drv# mpathadm list lu

        /dev/rdsk/c0t5000C50048697F5Fd0s2

                Total Path Count: 1

                Operational Path Count: 1

        /dev/rdsk/c0t6000D3100000650000000000000017C7d0s2

                Total Path Count: 4

                Operational Path Count: 4

        /dev/rdsk/c0t6000D3100000650000000000000017C6d0s2

                Total Path Count: 4

                Operational Path Count: 4

        /dev/rdsk/c0t6000D3100000650000000000000017C5d0s2

                Total Path Count: 4

                Operational Path Count: 4

        /dev/rdsk/c0t6000D3100000650000000000000017C4d0s2

                Total Path Count: 4

                Operational Path Count: 4
```

3 Fibre Channel and multipath (MPxIO)

The Dell Storage Center SAN supports both Fibre Channel protocol (FCP) and iSCSI attached volumes. Most Solaris SPARC environments have traditionally used FCP to connect Solaris hosts to the SAN. This section will focus on concepts pertaining directly to the FCP model.

The Dell SAN utilizes the native tools provided by the OS vendor to provide the interconnectivity between the host and the SAN. It is recommended to configure the environment with multipath whenever possible. For details on this, refer to the section Validating and activating multipath (MPxIO) below.

There are two Fibre Channel HBA brands (QLogic and Emulex) which are supported on the Solaris SPARC platform. These HBAs may come with either the native HBA vendor firmware or the Oracle/Sun-branded firmware. The QLogic and Oracle/Sun-QLogic adapters look the same physically. However, Oracle/Sun-QLogic adapters include Oracle/Sun firmware, the Oracle/Sun label, and a part number beginning with SG instead of QLA. The same is true with the Emulex and Oracle/Sun-Emulex adapters.

It is recommended for most environments to use the Oracle/Sun-branded firmware on Oracle/Sun-branded HBAs where feasible.

3.1 Drivers

With Solaris 11, the drivers for QLogic and Emulex HBAs are already incorporated into the native multipath IO stack.

There is no further need to install the SAN StorEdge Foundation Suite v4.4.x as was required with older Solaris OS releases.

These drivers can be validated with the following “pkg list” command as shown below.

```
root:~# pkg list | egrep -e 'fc|fp|qlc|emlxs'

driver/fc/emlxs                                   0.5.11-0.175.1.0.0.24.2    i--

driver/fc/qlc                                     0.5.11-0.175.1.0.0.24.2    i--

driver/fc/socal                                   0.5.11-0.175.1.0.0.24.2    i--

driver/network/ethernet/qlcnic                    0.5.11-0.175.1.0.0.24.2    i--

driver/storage/ifp                                0.5.11-0.175.1.0.0.24.2    i--

system/embedded-fcode-interpreter                 0.5.11-0.175.1.0.0.24.2    i--

system/io/fc/fc-fabric                            0.5.11-0.175.1.0.0.24.2    i--

system/io/fc/fc-port                              0.5.11-0.175.1.0.0.24.2    i--

system/io/fc/fc-san-management                    0.5.11-0.175.1.0.0.24.2    i--
```

```
system/io/fc/fc-scsi                              0.5.11-0.175.1.0.0.24.2    i--
system/io/fc/ip-over-fc                           0.5.11-0.175.1.0.0.24.2    i--
system/library/storage/libfcoe                    0.5.11-0.175.1.0.0.24.2    i--
system/storage/fc-utilities                       0.5.11-0.175.1.0.0.24.2    i--
system/storage/fcoe                               0.5.11-0.175.1.0.0.24.2    i--
system/storage/fcoe/fcoe-initiator                0.5.11-0.175.1.0.0.24.2    i--
```

3.2 Validating and activating multipath (MPxIO)

With Solaris (release 10U3 or newer and including 11.x OS releases), it has been the recommended best practice to manage multipath using the built-in multipath-utilities toolset (stmsboot, mpathadm etc.) instead of manually editing the /kernel/drv/fp.conf and mpt.conf files. These files may still be edited manually if required for an advanced configuration. There is no additional software required to connect the Solaris host to a Storage Center-based SAN. This greatly simplifies the connectivity and management of this layer of the IO stack.

This section below will discuss some of the tasks required to activate multipath on a Solaris host using these built-in utilities. The recommended best practice configurations for path and SCSI timeouts will also be discussed.

To validate that these built-in toolsets are installed, run the following commands shown below.

```
root:~# pkg list | grep multipath
system/storage/multipath-utilities                0.5.11-0.175.1.0.0.24.2    i–
root:~# pkg info system/storage/multipath-utilities
          Name: system/storage/multipath-utilities
       Summary: Solaris Multipathing CLI
   Description: Path Administration Utility for a Solaris Multipathing device
      Category: Applications/System Utilities
         State: Installed
     Publisher: solaris
       Version: 0.5.11
 Build Release: 5.11
        Branch: 0.175.1.0.0.24.2
```

```
Packaging Date: September 19, 2012 06:52:47 PM

          Size: 109.61 kB

          FMRI: pkg://solaris/system/storage/multipath-utilities@0.5.11,5.11-
0.175.1.0.0.24.2:20120919T185247Z
```

3.2.1 Activating multipath (MPxIO)

Activating multipath with previous versions of Solaris used to be a harrowing process, having to edit multiple files, run various commands etc. This process has since been greatly simplified with the introduction of the "stmsboot" and "mpathadm" commands as part of the multipath-utilities package as shown above. The following methods demonstrated below are presented as only one (1) of various ways to activate multipath, in this case, in its most basic form and configuration and to meet the functionality requirements when used with Dell Storage Center volumes.

```
root:~# stmsboot -D fp -e

root:~# cp /kernel/drv/scsi_vhci.conf /etc/driver/drv/scsi_vhci.conf

root:~# vi /etc/driver/drv/scsi_vhci.conf
```

Inside the FAILOVER_MODULE_BLOCK section of the /etc/driver/drv/scsi_vhci.conf file, add the following lines (1 per line) then save the file. These three (3) lines instruct the Solaris multipath IO stack to correctly identify and discover volumes presented from the Dell Storage Center.

```
scsi-vhci-failover-override =

"COMPELNTCompellent Vol", "f_sym";

symmetric-option = 0x1000000;
```

Finally, reboot the Solaris host to activate multipath (MPxIO).

3.3 Correlating Storage Center volumes to MPxIO device names

Once multipath (MPxIO) has been activated, the following command may be used to display all discovered Dell Storage Center volumes as shown below.

```
root:~# mpathadm list lu

        /dev/rdsk/c0t5000C50048697F5Fd0s2

                Total Path Count: 1

                Operational Path Count: 1

        /dev/rdsk/c0t6000D31000006500000000000000017C7d0s2

                Total Path Count: 4
                Operational Path Count: 4
        /dev/rdsk/c0t6000D3100000650000000000000017C6d0s2
                Total Path Count: 4
                Operational Path Count: 4
        /dev/rdsk/c0t6000D3100000650000000000000017C5d0s2
                Total Path Count: 4
                Operational Path Count: 4
        /dev/rdsk/c0t6000D3100000650000000000000017C4d0s2
                Total Path Count: 4
                Operational Path Count: 4
        /dev/rdsk/c0t6000D3100000650000000000000017C3d0s2
                Total Path Count: 4
                Operational Path Count: 4
```

All multipath devices in Solaris are reflected with a “c0t” prefixed designation; after this prefix, Solaris has appended the full string of the Storage Center volumes’ Serial Number as shown in RED above and finally postfixed with “d0s2”.

Using a simple script as shown below, a simple tabulated chart can be created correlating Solaris multipath devices to their respective Storage Center volumes.

```
root:~# for i in `echo | format | awk '{print $2}' | grep c0t6000D31`; do
CompelntSerial=`echo ${i} | cut -c4-35`; echo "${i}s2 = $CompelntSerial"; done

c0t6000D3100000650000000000000017C3d0s2 = 6000D3100000650000000000000017C3

c0t6000D3100000650000000000000017C4d0s2 = 6000D3100000650000000000000017C4

c0t6000D3100000650000000000000017C5d0s2 = 6000D3100000650000000000000017C5

c0t6000D3100000650000000000000017C6d0s2 = 6000D3100000650000000000000017C6

c0t6000D3100000650000000000000017C7d0s2 = 6000D3100000650000000000000017C7
```

As is shown below, the output from the “format” command is also changed to reflect the multipath device names instead of native Solaris cXtXdX devices.

```
root:~# echo | format

Searching for disks...done
```

```
AVAILABLE DISK SELECTIONS:

       0. c0t5000C50048697F5Fd0 <SUN300G cyl 46873 alt 2 hd 20 sec 625>  solaris

          /scsi_vhci/disk@g5000c50048697f5f

          /dev/chassis//SYS/HDD0/disk

       1. c0t6000D3100000650000000000000017C3d0 <COMPELNT-Compellent Vol-0604
cyl 31966 alt 2 hd 4 sec 164>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c3

       2. c0t6000D3100000650000000000000017C4d0 <COMPELNT-Compellent Vol-0604
cyl 31966 alt 2 hd 4 sec 164>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c4

       3. c0t6000D3100000650000000000000017C5d0 <COMPELNT-Compellent Vol-0604
cyl 31966 alt 2 hd 4 sec 164>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c5

       4. c0t6000D3100000650000000000000017C6d0 <COMPELNT-Compellent Vol-0604-
10.00GB>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c6

       5. c0t6000D3100000650000000000000017C7d0 <COMPELNT-Compellent Vol-0604-
10.00GB>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c7

Specify disk (enter its number): Specify disk (enter its number):
```

3.4 Correlating MPxIO device names to Solaris device names

There may be occasion where there is need to correlate the Solaris MPxIO device names to their respective and backing Solaris device names. This correlation can be achieved with the following command as shown below.

```
root:/kernel/drv# stmsboot -L | sort -k2 | egrep -v -e '^---|^non'

/dev/rdsk/c2t5000D31000006505d1 /dev/rdsk/c0t6000D3100000650000000000000017C3d0

/dev/rdsk/c2t5000D31000006506d1 /dev/rdsk/c0t6000D3100000650000000000000017C3d0
```

```
/dev/rdsk/c3t5000D31000006507d1 /dev/rdsk/c0t6000D3100000650000000000000017C3d0

/dev/rdsk/c3t5000D31000006508d1 /dev/rdsk/c0t6000D3100000650000000000000017C3d0

/dev/rdsk/c2t5000D31000006505d2 /dev/rdsk/c0t6000D3100000650000000000000017C4d0

/dev/rdsk/c2t5000D31000006506d2 /dev/rdsk/c0t6000D3100000650000000000000017C4d0

/dev/rdsk/c3t5000D31000006507d2 /dev/rdsk/c0t6000D3100000650000000000000017C4d0

/dev/rdsk/c3t5000D31000006508d2 /dev/rdsk/c0t6000D3100000650000000000000017C4d0

/dev/rdsk/c2t5000D31000006505d3 /dev/rdsk/c0t6000D3100000650000000000000017C5d0

/dev/rdsk/c2t5000D31000006506d3 /dev/rdsk/c0t6000D3100000650000000000000017C5d0

/dev/rdsk/c3t5000D31000006507d3 /dev/rdsk/c0t6000D3100000650000000000000017C5d0

/dev/rdsk/c3t5000D31000006508d3 /dev/rdsk/c0t6000D3100000650000000000000017C5d0

/dev/rdsk/c2t5000D31000006505d4 /dev/rdsk/c0t6000D3100000650000000000000017C6d0

/dev/rdsk/c2t5000D31000006506d4 /dev/rdsk/c0t6000D3100000650000000000000017C6d0

/dev/rdsk/c3t5000D31000006507d4 /dev/rdsk/c0t6000D3100000650000000000000017C6d0

/dev/rdsk/c3t5000D31000006508d4 /dev/rdsk/c0t6000D3100000650000000000000017C6d0

/dev/rdsk/c2t5000D31000006505d5 /dev/rdsk/c0t6000D3100000650000000000000017C7d0

/dev/rdsk/c2t5000D31000006506d5 /dev/rdsk/c0t6000D3100000650000000000000017C7d0

/dev/rdsk/c3t5000D31000006507d5 /dev/rdsk/c0t6000D3100000650000000000000017C7d0

/dev/rdsk/c3t5000D31000006508d5 /dev/rdsk/c0t6000D3100000650000000000000017C7d0
```

3.5 Configuring fcp_offline_delay and fp_offline_ticker

These two (2) parameters on Solaris dictate how the SCSI and HBA I/O layers behave when connectivity to the respective SCSI targets go missing for momentary or extended durations.

The first parameter fcp_offline_delay determines the wait time to offline a SCSI target, should the volume momentarily disappear. The value of this parameter defaults to 20 seconds. It is recommended best practice to change this to 60 seconds. Adjusting this parameter to 60 seconds allows the Storage Center up to 60 seconds to complete a controller failover event and present the same volumes on its' alternate (2nd controller) paths.

The second parameter fp_offline_ticker determines the wait time to offline a HBA port due to loss of "light". The value of this parameter defaults to 90 seconds. It is recommended best practice to leave this value at 90 seconds. This parameter should be configured to a value and a factor larger than the fcp_offline_delay parameter so as to avoid the HBA port being accidentally offlined during a controller failover event and/or interrupting business and production I/O needs.

3.5.1 Applying the configuration

These two (2) parameters are managed in the /kernel/drv/fcp.conf and /kernel/drv/fp.conf files respectively. Any changes made to these files only take effect after a reboot of the Solaris host. These files should be edited as shown below, appending the following lines to the end of the files' existing content.

```
root:~# vi /kernel/drv/fcp.conf

[snip]

fcp_offline_delay=60;


root:~# vi /kernel/drv/fp.conf

[snip]

fp_offline_ticker=90;


root:~# shutdown -y -i6 -g0
```

Shutdown and reboot the Solaris host where most appropriate to the business needs accordingly.

4 Software iSCSI

The Dell Storage Center supports both Fibre Channel and iSCSI SAN connectivity. The features of Storage Center are not locked into either protocol. Although one connectivity protocol may be more appropriate than the other depending on the situation, the Storage Center features are available and supported on both types of connectivity.

This section covers some of the basics regarding iSCSi connectivity between a Solaris host and a Dell Storage Center. The information presented below is intended as a starting point for utilizing iSCSI and is not all-encompassing, but rather equips the system administrator with enough information to get started with Solaris iSCSI usage with Storage Center.

Further information on iSCSI with Solaris can be found at the Oracle URL below.
http://docs.oracle.com/cd/E23824_01/html/821-1459/fncpc.html

With Solaris 11.1, the onboard Ethernet ports can be used to initiate an iSCSI connection. Several packages may need to be installed in order to use iSCSI on Solaris. First, the following packages need to be verified and installed on the Solaris host. These packages are already part of the default Solaris 11 installed catalog.

```
root:~# pkg list | grep iscsi

system/storage/iscsi/iscsi-initiator                0.5.11-0.175.1.0.0.24.2    i--

system/storage/iscsi/iscsi-iser                     0.5.11-0.175.1.0.0.24.2    i--
```

The Solaris iSCSI target software can use standard volumes from the Dell Storage Center as a backing store. All the Storage Center needs to provide is a standard volume to the server via normal device discovery methods, using built-in Solaris commands and utilities such as iscsiadm, devfsadm, format etc. No additional software is required.

4.1 Mapping an iSCSI volume

Verify the iSCSI control port IP addresses on the Storage Center controllers. This is located in the Controllers -> <Controller Name> -> IO Cards -> iSCSI subtree. Akin to Fibre Channel connectivity, multipath configuration is recommended for iSCSI as well. In the interest of conciseness and due to hardware limitations of the test environment, only a single path setup and configuration will be demonstrated below.

> Best practice recommendation is to have at least two control ports configured on the Storage Center, across multiple controllers and to have these on two (2) different subnets/VLANs within the network.

In this demonstration, iSCSI will be configured to ride on top of the primary network interface net0 as shown below.

```
root:~# ifconfig -a
```

```
lo0: flags=2001000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4,VIRTUAL> mtu 8232 index
1
        inet 127.0.0.1 netmask ff000000
net0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
        inet 172.16.26.88 netmask fffff000 broadcast 172.16.31.255
        ether 0:21:28:fa:d2:b2
[snip]
net6: flags=20002000840<RUNNING,MULTICAST,IPv6> mtu 1500 index 3
        inet6 ::/0
        ether 2:21:28:57:47:17
```

The first step is to setup the discovery address to each iSCSI control port on the Storage Center, then enable the sendtargets discovery method. This may take up to a few seconds to complete. In this scenario, the Storage Center iSCSI control port is located at IP address 172.16.26.180 on port 3260 (iSCSI default).

```
root:~# iscsiadm add discovery-address 172.16.26.180:3260
root:~# iscsiadm list discovery-address
Discovery Address: 172.16.26.180:3260
root:~# iscsiadm modify discovery --sendtargets enable
root:~# iscsiadm list discovery
Discovery:
        Static: disabled
        Send Targets: enabled
        iSNS: disabled
```

At this point, the Storage Center Server Object can then be created. Select the appropriate iqn.* HBA, name the Server Object and select the Operating System to be "Solaris 10". The "Solaris 10" object type is functionally equivalent to that for Solaris 11.x OS.

When this is complete, volumes can then be mapped from Storage Center to the new iSCSI Server Object. A new volume named maximus_Solaris11BP_DataVols_10g_00_iscsi is created and mapped to the Solaris host maximus. With iSCSI, new volumes are discovered and made available via the command as shown below.

```
root:~# devfsadm -i iscsi -v

devfsadm[29907]: verbose: symlink /dev/dsk/c1t7d0s0 ->
../../devices/iscsi/disk@0000iqn.2002-
03.com.compellent%3A5000d310000067800000,1:a

devfsadm[29907]: verbose: symlink /dev/dsk/c1t7d0s1 ->
../../devices/iscsi/disk@0000iqn.2002-
03.com.compellent%3A5000d310000067800000,1:b

[snip]

devfsadm[29907]: verbose: symlink /dev/rdsk/c1t8d0s4 ->
../../devices/iscsi/disk@0000iqn.2002-
03.com.compellent%3A5000d310000067820000,1:e,raw

devfsadm[29907]: verbose: symlink /dev/rdsk/c1t8d0s5 ->
../../devices/iscsi/disk@0000iqn.2002-
03.com.compellent%3A5000d310000067820000,1:f,raw

devfsadm[29907]: verbose: symlink /dev/rdsk/c1t8d0s6 ->
../../devices/iscsi/disk@0000iqn.2002-
03.com.compellent%3A5000d310000067820000,1:g,raw

devfsadm[29907]: verbose: symlink /dev/rdsk/c1t8d0s7 ->
../../devices/iscsi/disk@0000iqn.2002-
03.com.compellent%3A5000d310000067820000,1:h,raw

root:~# echo | format

Searching for disks...done


c1t7d0: configured with capacity of 10.00GB

c1t8d0: configured with capacity of 10.00GB



AVAILABLE DISK SELECTIONS:

       0. c0t5000CCA025442A24d0 <SUN300G cyl 46873 alt 2 hd 20 sec 625>  solaris

          /scsi_vhci/disk@g5000cca025442a24

          /dev/chassis//SYS/HDD0/disk

       1. c1t7d0 <COMPELNT-Compellent Vol-0505 cyl 31966 alt 2 hd 4 sec 164>
```

```
          /iscsi/disk@0000iqn.2002-03.com.compellent%3A5000d310000067800000,1

       2. c1t8d0 <COMPELNT-Compellent Vol-0505 cyl 31966 alt 2 hd 4 sec 164>

          /iscsi/disk@0000iqn.2002-03.com.compellent%3A5000d310000067820000,1

Specify disk (enter its number): Specify disk (enter its number):
```

In the above scenario, multipath had not been configured, though if multiple iSCSI paths were configured, the Solaris iSCSI stack will work natively with MPxIO to manage path naming, path use, redundancy and failover. With multiple paths configured, iSCSI performance and throughput will be noticeably improved as well. Additionally, it is a recommend best practice (where feasible) to isolate all storage and backup traffic from the primary network interfaces of the Solaris host by using either more physical cabling, VLANs or by other means. This would provide among many things, better security, reduced latency, improved IO performance etc.

4.1.1 Configuring jumbo frames

Jumbo frames (MTU size) can and should be used to improve IO performance over iSCSI networks. To enable jumbo frames, the appropriate network interface configuration file needs to be edited. In this scenario, the net0 device is represented by the file /kernel/drv/e1000g.conf. This latter file needs to be modified as shown below.

```
root:/kernel/drv# vi /kernel/drv/e1000g.conf
```

The key value pair MaxFrameSize=X,X,X,X,X,X,X,X,X,X,X,X,X,X,X,X; should be updated to the value as shown below.

```
MaxFrameSize=3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3;
```

A reboot of the Solaris host is then required to effect this change. After the reboot, the ifconfig output which used to show “mtu 1500” will now show “mtu 9216” as shown below.

```
root:~# ifconfig -a

lo0: flags=2001000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4,VIRTUAL> mtu 8232 index
1

        inet 127.0.0.1 netmask ff000000

net0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 9216 index 2

        inet 172.16.26.88 netmask fffff000 broadcast 172.16.31.255

        ether 0:21:28:fa:d2:b2

[snip]

net6: flags=20002000840<RUNNING,MULTICAST,IPv6> mtu 1500 index 3
```

```
        inet6 ::/0
        ether 2:21:28:57:47:17
```

To complete the process in enabling jumbo frames; any Ethernet switches providing connectivity from the Solaris host to Storage Center should also be configured accordingly to enable jumbo frames. Refer to respective switch vendor documentation for these instructions accordingly.

4.2 Removing an iSCSI volume

Removing one or multiple iSCSI volumes from a Solaris host is essentially similar to the process used to remove Fibre Channel volumes. In the interest of conciseness, refer to the section Removing volumes below.

4.3 Removing an iSCSI discovery address

If the Storage Center control port IP address were to change, it would then be required to remove the old address from the Solaris host as well; subsequently adding the new control port IP address to replace it. This is achieved with the use of the "iscsiadm" command as shown below.

```
root:~# iscsiadm remove discovery-address 172.16.26.180:3260
root:~# iscsiadm list discovery-address
root:~# iscsiadm add discovery-address <new IP address>:3260
```

5 Managing volumes

Understanding how volumes are managed in the Solaris file system requires a basic understanding of the Solaris device structure. The /devices directory contains a physical device file for every hardware device and device class attached to the Solaris host. In addition, there is the /dev directory, which contains both dsk and rdsk subdirectories. These subdirectories have symbolic links that reference the block and raw device paths listed in the /devices tree. Fibre Channel- attached devices will be found under the /devices/scsi_vhci device tree and will utilize the ssd device driver.

Any volumes mapped to the Solaris host will have multiple files located under this /devices/scsi_vhci path as a volume and be presented in the format: ssd@port,target:partition

To illustrate this, see the output from “mpathadm” and “ls -la” below.

```
root:~# mpathadm list lu

        /dev/rdsk/c0t5000C50048697F5Fd0s2

                Total Path Count: 1

                Operational Path Count: 1

        /dev/rdsk/c0t6000D3100000650000000000000017C7d0s2

                Total Path Count: 4

                Operational Path Count: 4

        [snip]

root:~# ls -la /devices/scsi_vhci/*6000d31*:[a-z]

[snip]

brw-r-----   1 root     sys      231,  0 Oct  7 11:44
/devices/scsi_vhci/ssd@g6000d3100000650000000000000017c7:a

brw-r-----   1 root     sys      231,  1 Oct 14 15:45
/devices/scsi_vhci/ssd@g6000d3100000650000000000000017c7:b

brw-r-----   1 root     sys      231,  2 Oct 14 15:45
/devices/scsi_vhci/ssd@g6000d3100000650000000000000017c7:c

brw-r-----   1 root     sys      231,  3 Oct 14 15:45
/devices/scsi_vhci/ssd@g6000d3100000650000000000000017c7:d

brw-r-----   1 root     sys      231,  4 Oct 14 15:45
/devices/scsi_vhci/ssd@g6000d3100000650000000000000017c7:e
```

```
brw-r-----   1 root     sys      231,  5 Oct 14 15:45
/devices/scsi_vhci/ssd@g6000d3100000650000000000000017c7:f

brw-r-----   1 root     sys      231,  6 Oct 14 15:45
/devices/scsi_vhci/ssd@g6000d3100000650000000000000017c7:g
```

5.1 Scanning for new volumes

After volumes have been presented from the Storage Center to the Solaris host, the Solaris host would then need to scan its' respective HBA controller ports to identify and discover these new volumes. This is achieved via the methods shown below and where needed, highlighted in RED for significance.

```
root:~# for i in `fcinfo hba-port | grep cfg | awk -F/ '{print $4}'`; do cfgadm
-c configure ${i}; done

root:~# devfsadm
```

Once this is complete and if multipath has already been activated, then refer to the section above "Correlating Storage Center volumes to MPxIO device names". If the Solaris host is operating in single-path mode or otherwise, then proceed to the section below.

5.2 Correlating Storage Center volumes to OS device names

Although Dell Storage Center volumes will most often be used in multipath environments, there may be certain circumstances where these volumes may not. In these latter circumstances, the "format" command output may still be used to correlate Solaris device names to their respective Dell Storage Center volumes as shown below.

```
root:~# cfgadm -c configure c6; cfgadm –c configure c7

root:~# echo | format

Searching for disks...done


c6t5000D31000006505d1: configured with capacity of 10.00GB

c6t5000D31000006506d1: configured with capacity of 10.00GB

c7t5000D31000006507d1: configured with capacity of 10.00GB

c7t5000D31000006508d1: configured with capacity of 10.00GB



AVAILABLE DISK SELECTIONS:
       0. c0t5000CCA025442A24d0 <SUN300G cyl 46873 alt 2 hd 20 sec 625>  solaris

          /scsi_vhci/disk@g5000cca025442a24

          /dev/chassis//SYS/HDD0/disk

       1. c6t5000D31000006505d1 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0/fp@0,0/ssd@w5000d31000006505,1

       2. c6t5000D31000006506d1 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0/fp@0,0/ssd@w5000d31000006506,1

       3. c7t5000D31000006507d1 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0,1/fp@0,0/ssd@w5000d31000006507,1

       4. c7t5000D31000006508d1 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0,1/fp@0,0/ssd@w5000d31000006508,1

Specify disk (enter its number): Specify disk (enter its number):
```

The output above represents a single 10GB volume presented to a Solaris host named maximus. Multipath is NOT enabled and this single volume is thus represented by four (4) Solaris device names c6t5000D31000006505d1, c6t5000D31000006506d1, c7t5000D31000006507d1 and c7t5000D31000006508d1 respectively. Each of these devices represents a single path by which the volume is presented from Storage Center; in this case and as shown in RED above, the Storage Center NPIV WWPNs are embedded into the device name. Additionally, the d1 postfix on the Solaris device name represents the LUN# by which the Storage Center presents it. An alternative to correlating each Solaris device name to its respective Storage Center volume is to issue the following command.

```
root:~# for i in `echo | format | awk '{print $2}' | egrep "5000D31"`; do echo
"$i = $(luxadm inq /dev/rdsk/${i}s2 | grep ^Serial)"; done | sort -k5

c6t5000D31000006505d1 = Serial Number              00000065-000017ca

c6t5000D31000006506d1 = Serial Number              00000065-000017ca

c7t5000D31000006507d1 = Serial Number              00000065-000017ca

c7t5000D31000006508d1 = Serial Number              00000065-000017ca

c6t5000D31000006505d2 = Serial Number              00000065-000017cb

c6t5000D31000006506d2 = Serial Number              00000065-000017cb
```

```
c7t5000D31000006507d2 = Serial Number                00000065-000017cb

c7t5000D31000006508d2 = Serial Number                00000065-000017cb
```

5.3 Removing volumes

When removing a volume from a Solaris host, it is important to execute this process in the proper order to ensure that the volume is cleanly removed from the system. The recommended best practice is to always remove the volume from the host before unmapping the volume from the Storage Center. Removing the volume in the opposite order may result in a Solaris host behaving oddly and impede the system administrator's ability to manage the host effectively. The methods demonstrated below represent the recommended best practices for removing a volume from the Solaris host.

First, unmount the volume from the Solaris host if it is a mounted filesystem. Since the volume is presented to the Solaris host on multiple paths (not to be confused with multipath), apply the "luxadm" command to take each respective path offline as shown below. In the example below, the *d2 (LUN# 2) will be removed from the Solaris host with a simple script (see below).

```
root:~# echo | format

Searching for disks...done


c6t5000D31000006506d1: configured with capacity of 10.00GB

c6t5000D31000006505d1: configured with capacity of 10.00GB

c6t5000D31000006505d2: configured with capacity of 10.00GB

c6t5000D31000006506d2: configured with capacity of 10.00GB

c7t5000D31000006507d1: configured with capacity of 10.00GB

c7t5000D31000006508d1: configured with capacity of 10.00GB

c7t5000D31000006507d2: configured with capacity of 10.00GB

c7t5000D31000006508d2: configured with capacity of 10.00GB




AVAILABLE DISK SELECTIONS:

       0. c0t5000CCA025442A24d0 <SUN300G cyl 46873 alt 2 hd 20 sec 625>  solaris

          /scsi_vhci/disk@g5000cca025442a24
          /dev/chassis//SYS/HDD0/disk

       1. c6t5000D31000006506d1 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0/fp@0,0/ssd@w5000d31000006506,1

       2. c6t5000D31000006505d1 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0/fp@0,0/ssd@w5000d31000006505,1

       3. c6t5000D31000006505d2 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0/fp@0,0/ssd@w5000d31000006505,2

       4. c6t5000D31000006506d2 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0/fp@0,0/ssd@w5000d31000006506,2

       5. c7t5000D31000006507d1 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0,1/fp@0,0/ssd@w5000d31000006507,1

       6. c7t5000D31000006508d1 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0,1/fp@0,0/ssd@w5000d31000006508,1

       7. c7t5000D31000006507d2 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0,1/fp@0,0/ssd@w5000d31000006507,2

       8. c7t5000D31000006508d2 <COMPELNT-Compellent Vol-0604 cyl 31966 alt 2 hd
4 sec 164>

          /pci@400/pci@1/pci@0/pci@8/SUNW,qlc@0,1/fp@0,0/ssd@w5000d31000006508,2

Specify disk (enter its number): Specify disk (enter its number):

root:~# for i in `echo | format | awk '{print $2}' | egrep "d2"`; do echo "===
going offline ${i}s2"; luxadm -e offline /dev/rdsk/${i}s2; echo; done
=== going offline c6t5000D31000006505d2s2



=== going offline c6t5000D31000006506d2s2
=== going offline c7t5000D31000006507d2s2


=== going offline c7t5000D31000006508d2s2
```

The “echo | format” command will no longer show these devices after their removal from the system. At this point, the mappings from Storage Center may be safely removed. The “cfgadm” command is then used to rescan the HBA controller ports on the Solaris host so it no longer references the removed volumes.

```
root:~# cfgadm -o force_update –c configure c6; cfgadm –o force_update \ -c
configure c7
```

Finally, apply the “devfsadm” command to clean up the device symbolic links and any references to the removed volumes from the Solaris host.

```
root:~# devfsadm -Cv
```

5.3.1 Removing multipath volumes

In a similar manner, removing multipath volumes will use essentially the same methods as shown above; applying the “luxadm -e offline” command to the multipath volume name instead of each individual Solaris device name/path. The “devfsadm” command should also be run to clean up any remaining unused device files.

```
root:~# echo | format

Searching for disks...done




AVAILABLE DISK SELECTIONS:

       0. c0t5000C50048697F5Fd0 <SUN300G cyl 46873 alt 2 hd 20 sec 625>  solaris

          /scsi_vhci/disk@g5000c50048697f5f

          /dev/chassis//SYS/HDD0/disk

       1. c0t6000D31000006500000000000000017C3d0 <COMPELNT-Compellent Vol-0604
cyl 31966 alt 2 hd 4 sec 164>

          /scsi_vhci/ssd@g6000d31000006500000000000000017c3

       2. c0t6000D31000006500000000000000017C4d0 <COMPELNT-Compellent Vol-0604
cyl 31966 alt 2 hd 4 sec 164>
```

```
          /scsi_vhci/ssd@g6000d3100000650000000000000017c4

       3. c0t6000D3100000650000000000000017C5d0 <COMPELNT-Compellent Vol-0604
cyl 31966 alt 2 hd 4 sec 164>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c5

       4. c0t6000D3100000650000000000000017C6d0 <COMPELNT-Compellent Vol-0604-
10.00GB>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c6

       5. c0t6000D3100000650000000000000017C7d0 <COMPELNT-Compellent Vol-0604-
10.00GB>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c7

Specify disk (enter its number): Specify disk (enter its number):

root:~#

root:~#

root:~# luxadm -e offline /dev/rdsk/c0t6000D3100000650000000000000017C3d0s2
```

6 Partitions and file systems

As a block level SAN, the Dell Storage Center will take any partition and file system scheme supported by the Solaris OS. However, there are a few things to take into consideration when designing a scheme.

6.1 Partitions (slices)

Traditionally, Solaris LUNs are provisioned with different partitions (or slices) for the function of the file system. For example, / is often on a separate partition from /usr or the /export/home partitions etc. Although this is appropriate for the primary boot volume, it is not recommended to create elaborate partition schemes when using volumes from Storage Center. Although multiple partitions are supported, it often leads to unnecessary complexity when it comes to managing volumes presented to the Solaris host from Storage Center. After mapping a volume to Solaris, it is recommended to only use a single partition on the volume. Dell recommends assigning volumes on a "per-purpose" basis. For example, the OS volume should be separate from a data volume (e.g. database data) and a separate volume for logging (e.g. database logs). For each volume presented, a single partition would be utilized on the volume; this single partition is typically denoted by a "s2" postfix after the device name, often termed "whole disk" or "slice 2". By allocating volumes based on a business need, the data and function of the volume can be logically administered for that purpose, which may have different retention, security and performance requirements etc. depending on its purpose.

6.2 Disk labels

When a volume is mapped to a Solaris host and subsequently discovered using the "cfgadm" command and having a disk label applied via the format tool; the label applied by default is an SMI label for any volume less than two (2) terabyte (TB) in size. Any volumes larger than 2TB needs to have an EFI label applied; an EFI label is applied using the "format -e" command. EFI labels are also the default label used when working with ZFS pools, file systems etc.

If a UFS file system is to be used, which may start below the 2TB ceiling, but has the potential to grow beyond 2TB, then it is recommended to use the EFI label instead. In most cases, it is considered a best practice to use the EFI label for all volumes except the boot volumes.

In traditional usage such as the boot volume in a boot-from-SAN configuration, using an SMI label is still the recommended approach; it is also recommended to keep the boot volume separate from data volumes and to keep it less than 2TB in size.

As of Solaris 11.1, booting the OS from a disk or volume with an EFI label is NOT supported.

6.2.1 SMI labels

A volume which is less than 2TB in size will use the SMI label by default, which will setup partitions (or slices) using the default system defined scheme with "slice 2" representing the whole disk. The traditional SMI label, which consists of eight (8) partitions/slices, numbered 0 to 7 and is limited to a 2TB maximum size.

6.2.2 EFI labels

A volume which is greater than 2TB, or has a ZFS pool created on it, will use the EFI label by default. The EFI label is required for ZFS and brings with it certain advanced capabilities. The EFI label consists of eight partitions, numbered 0 to 8, skipping 7 and is required for LUNs greater than 2TB in size. Please refer to the Oracle documentation for further details.

6.3 UFS file system

The traditional and most often used file system on Solaris is the UFS file system. This section discusses some recommendations when using this file system type on Solaris with Dell Storage Center.

The UFS file system is capable of files and a file system size of 16TB. Although there may be practical limitations that would prevent such a size from being implemented in production, the Dell Storage Center supports volume sizes up to the limits of the Solaris UFS implementation.

Once the volume is mapped from Storage Center to the Solaris host and discovered for use, a file system can then be placed on the volume and mounted for use. The "newfs" or "mkfs" commands may be used to apply the file system to the volume. The example below demonstrates the use of the "newfs" command, thereafter mounting the new file system on the Solaris host.

Note: UFS file systems are not Thin Reclamation aware and will not participate with the Storage Center Free Space Recovery algorithms.

```
root:~# mpathadm list lu

        /dev/rdsk/c0t5000C50048697F5Fd0s2

                Total Path Count: 1

                Operational Path Count: 1

        /dev/rdsk/c0t6000D3100000650000000000000017C7d0s2

                Total Path Count: 8

                Operational Path Count: 8

        /dev/rdsk/c0t6000D3100000650000000000000017C6d0s2

                Total Path Count: 4

                Operational Path Count: 4

        /dev/rdsk/c0t6000D3100000650000000000000017C5d0s2

                Total Path Count: 4

                Operational Path Count: 4
        /dev/rdsk/c0t6000D3100000650000000000000017C4d0s2

                Total Path Count: 4

                Operational Path Count: 4

        /dev/rdsk/c0t6000D3100000650000000000000017C3d0s2

                Total Path Count: 4

                Operational Path Count: 4
```

Applying the "newfs" command to the multipath device /dev/rdsk/c0t6000D3100000650000000000000017C5d0s2.

```
root:~# newfs /dev/rdsk/c0t6000D3100000650000000000000017C5d0s2

newfs: construct a new file system
/dev/rdsk/c0t6000D3100000650000000000000017C5d0s2: (y/n)? y

Warning: 5920 sector(s) in last cylinder unallocated

/dev/rdsk/c0t6000D3100000650000000000000017C5d0s2:      20969696 sectors in 3414
cylinders of 48 tracks, 128 sectors

        10239.1MB in 214 cyl groups (16 c/g, 48.00MB/g, 5824 i/g)

super-block backups (for fsck -F ufs -o b=#) at:

 32, 98464, 196896, 295328, 393760, 492192, 590624, 689056, 787488, 885920,

 20055584, 20154016, 20252448, 20350880, 20449312, 20547744, 20646176,

 20744608, 20843040, 20941472
```

And mounting the new file system to the directory structure /10g_UFS. Update the /etc/vfstab file accordingly if the mount point needs to be persistent across reboots of the Solaris host.

```
root:~# mount /dev/dsk/c0t6000D3100000650000000000000017C5d0s2 /10g_UFS

root:~# df -k

Filesystem           1024-blocks        Used   Available Capacity  Mounted on

rpool/ROOT/solaris     286949376     2295496   273879937     1%    /

[snip]

rpool/export           286949376          32   273879937     1%    /export

rpool/export/home      286949376          31   273879937     1%    /export/home
```

```
rpool                   286949376         73   273879937     1%    /rpool

/dev/dsk/c4t6d0s2          694700     694700           0   100%
/media/Oracle_Solaris-11_1-Text-SPARC

pool01                   10257408         32    10257277     1%    /pool01

pool01/2gZFS             10257408         31    10257277     1%    /pool01/2gZFS

172.16.26.131:/nfsroot

                        103212320   54758292    43211148    56%    /Tools

/dev/dsk/c0t6000D31000006500000000000000017C5d0s2

                         10325612      10257    10212099     1%    /10g_UFS
```

6.3.1 Expanding a UFS file system

A common administration task is to allocate additional space to an existing file system. The Dell Storage Center supports the expansion of a UFS based file system using Solaris native utilities to accomplish such a task.

The example below demonstrates the process of expanding a volume on Storage Center and then expanding the UFS file system to take advantage of this new space. The 10g volume will be expanded to a 30g volume accordingly.

There is always some risk to data loss whenever the underlying file system is modified. Before executing any of the steps below, make sure to take a backup of the volume. Also, it is very important to keep the same partition layout.

The first step in this process is to unmount the /10g_UFS file system.

```
root:~# umount /10g_UFS
```

At this point, the volume on the Storage Center can then be expanded from 10GB to 30GB via the GUI. After the volume size has been expanded, the Solaris host needs to be made aware of this change (i.e. disk geometry changes etc.). The “format” utility is started, select the volume which has been expanded (shown in RED below), select (ty)pe -> (0) and then (p)rint again to display the new partition table, then press (l)able to apply the new partition table. Notice that the Cylinders ceiling has changed from 31965 to 49771 for partition 2 (slice 2 or whole disk). Finally, press (q)uit two (2) time to exit from the “format” utility.

```
root:~# format

Searching for disks...done
```

```
AVAILABLE DISK SELECTIONS:

       0. c0t5000C50048697F5Fd0 <SUN300G cyl 46873 alt 2 hd 20 sec 625>  solaris

          /scsi_vhci/disk@g5000c50048697f5f

          /dev/chassis//SYS/HDD0/disk

       1. c0t6000D3100000650000000000000017C4d0 <COMPELNT-Compellent Vol-0604
cyl 31966 alt 2 hd 4 sec 164>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c4

       2. c0t6000D3100000650000000000000017C5d0 <COMPELNT-Compellent Vol-0604
cyl 31966 alt 2 hd 4 sec 164>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c5

       3. c0t6000D3100000650000000000000017C6d0 <COMPELNT-Compellent Vol-0604-
10.00GB>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c6

       4. c0t6000D3100000650000000000000017C7d0 <COMPELNT-Compellent Vol-0604-
10.00GB>

          /scsi_vhci/ssd@g6000d3100000650000000000000017c7

Specify disk (enter its number): 2

selecting c0t6000D3100000650000000000000017C5d0

[disk formatted]

Note: detected additional allowable expansion storage space that can be

added to current SMI label's computed capacity.

Select <partition> <expand> to adjust the label capacity.



FORMAT MENU:

        disk       - select a disk

        type       - select (define) a disk type

        partition  - select (define) a partition table
        current    - describe the current disk
        format     - format and analyze the disk
        repair     - repair a defective sector
        label      - write label to the disk
        analyze    - surface analysis
        defect     - defect list management
        backup     - search for backup labels
        verify     - read and display labels
        save       - save new disk/partition definitions
        inquiry    - show disk ID
        volname    - set 8-character volume name
        !<cmd>     - execute <cmd>, then return
        quit
format> ty


AVAILABLE DRIVE TYPES:
        0. Auto configure
        1. Quantum ProDrive 80S
        2. Quantum ProDrive 105S
        3. CDC Wren IV 94171-344
        4. SUN0104
        5. SUN0207
        6. SUN0327
        7. SUN0340
        8. SUN0424
        9. SUN0535
        10. SUN0669
        11. SUN1.0G
        12. SUN1.05
        13. SUN1.3G
        14. SUN2.1G
        15. SUN2.9G
        16. Zip 100
        17. Zip 250
        18. Peerless 10GB
        19. SUN300G
        20. COMPELNT-Compellent Vol-0604
        21. COMPELNT-Compellent Vol-0604
        22. other
Specify disk type (enter its number)[21]: 0
c0t6000D3100000650000000000000017C5d0: configured with capacity of 30.00GB
<COMPELNT-Compellent Vol-0604 cyl 49772 alt 2 hd 8 sec 158>
selecting c0t6000D3100000650000000000000017C5d0
[disk formatted]
format> p


PARTITION MENU:
        0      - change `0' partition
        1      - change `1' partition
        2      - change `2' partition
        3      - change `3' partition
        4      - change `4' partition
        5      - change `5' partition
        6      - change `6' partition
        7      - change `7' partition
        select - select a predefined table
        modify - modify a predefined partition table
        name   - name the current table
        print  - display the current table
        label  - write partition map and label to the disk
        !<cmd> - execute <cmd>, then return
        quit
partition> p
Current partition table (default):
Total disk cylinders available: 49772 + 2 (reserved cylinders)

Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm        0 -   207      128.38MB    (208/0/0)      262912
  1       swap    wu      208 -   415      128.38MB    (208/0/0)      262912
  2     backup    wu        0 - 49771       30.00GB    (49772/0/0) 62911808
  3 unassigned    wm        0                0          (0/0/0)            0
  4 unassigned    wm        0                0          (0/0/0)            0
  5 unassigned    wm        0                0          (0/0/0)            0
  6        usr    wm      416 - 49771       29.75GB    (49356/0/0) 62385984
  7 unassigned    wm        0                0          (0/0/0)            0

partition> l
Ready to label disk, continue? y
```

```
partition> q



FORMAT MENU:

        disk       - select a disk

        type       - select (define) a disk type

        partition  - select (define) a partition table

        current    - describe the current disk

        format     - format and analyze the disk

        repair     - repair a defective sector

        label      - write label to the disk

        analyze    - surface analysis

        defect     - defect list management

        backup     - search for backup labels

        verify     - read and display labels

        save       - save new disk/partition definitions

        inquiry    - show disk ID

        volname    - set 8-character volume name

        !<cmd>     - execute <cmd>, then return

        quit

format> q
```

The file system is then remounted, but it is observed below that it still shows a 10GB capacity.

```
root:~# mount /dev/dsk/c0t6000D3100000650000000000000017C5d0s2 /10g_UFS

root:~# df -k

Filesystem           1024-blocks        Used   Available Capacity  Mounted on

rpool/ROOT/solaris     286949376     2295007   273850029     1%    /

[snip]
```

```
rpool/export           286949376          32    273850029     1%    /export

rpool/export/home      286949376          31    273850029     1%    /export/home

rpool                  286949376          73    273850029     1%    /rpool

/dev/dsk/c4t6d0s2         694700      694700            0   100%
/media/Oracle_Solaris-11_1-Text-SPARC

pool01                  10257408          32     10257277     1%    /pool01

pool01/2gZFS            10257408          31     10257277     1%    /pool01/2gZFS

172.16.26.131:/nfsroot

                       103212320    54758292     43211148    56%    /Tools

/dev/dsk/c0t6000D31000000650000000000000017C5d0s2

                        10325612       10257     10212099     1%    /10g_UFS
```

At this time, it is then required to apply the "growfs" command to expand the UFS file system to the full capacity of the underlying volume as shown below.

```
root:~# growfs -M /10g_UFS /dev/rdsk/c0t6000D31000000650000000000000017C5d0s2

Warning: 2752 sector(s) in last cylinder unallocated

/dev/rdsk/c0t6000D31000000650000000000000017C5d0s2:      62911808 sectors in
10240 cylinders of 48 tracks, 128 sectors

        30718.7MB in 640 cyl groups (16 c/g, 48.00MB/g, 5824 i/g)

super-block backups (for fsck -F ufs -o b=#) at:

 32, 98464, 196896, 295328, 393760, 492192, 590624, 689056, 787488, 885920,

Initializing cylinder groups:

............

super-block backups for last 10 cylinder groups at:

 61938464, 62036896, 62135328, 62233760, 62332192, 62430624, 62529056,

 62627488, 62725920, 62824352
```

The new "df –k" command now shows the mounted /10g_UFS file system at its full 30GB capacity.

```
root:~# mount /dev/dsk/c0t6000D31000000650000000000000017C5d0s2 /10g_UFS

root:~# df -k
```

```
Filesystem           1024-blocks         Used   Available Capacity  Mounted on

rpool/ROOT/solaris     286949376     2295007   273850029     1%    /

[snip]

rpool/export           286949376          32   273850029     1%    /export

rpool/export/home      286949376          31   273850029     1%    /export/home

rpool                  286949376          73   273850029     1%    /rpool

/dev/dsk/c4t6d0s2         694700      694700           0   100%
/media/Oracle_Solaris-11_1-Text-SPARC

pool01                  10257408          32    10257277     1%    /pool01

pool01/2gZFS            10257408          31    10257277     1%    /pool01/2gZFS

172.16.26.131:/nfsroot

                       103212320    54758292    43211148    56%    /Tools

/dev/dsk/c0t6000D3100000650000000000000017C5d0s2

                        30979718       30737    30737292     1%    /10g_UFS
```

6.3.2 UFS and Data Instant Replay

A Storage Center Replay is an exact point-in-time copy of the volume at the time the Replay is captured. Likewise, a Replay of a volume which serves a UFS file system on a Solaris host is an exact copy of the UFS file system contents as well at that point-in-time. Therefore, it is important to understand how this “state” needs to be considered when later using the contents of the Replay. Below are some general best practice guidelines and considerations to be aware of when working with Replays of UFS volumes.

1. It is recommended to map the Replay View volume created from a Replay to a different Solaris host than the host in which the Replay was captured. If attempting to recover file(s) it is recommended to map the Replay View volume to a “recovery” host and then copy the necessary data back to the original host over the network. However, it is supported to map the Replay View volume to the original host, but care needs to be taken to not confuse the Production volume with the Replay View volume. For example, if volume names (set by the “format” command’s VOLNAME option) are used, a mapped Replay View volume will also present itself as the same VOLNAME and this could lead to confusion in use.
2. If the UFS file system is mounted when the Replay is taken, all mount flags are preserved in the Replay. When the Replay volume is then presented to another server, those UFS mount flags remain intact.
3. When mounting the Replay View volume of a UFS file system that was mounted at the time the Replay was captured, the system administrator may be prompted that a file system consistency check is required. This is normal, and should be run using the standard Solaris “fsck” utility.

The example below demonstrates the process used to capture a Replay of a UFS based volume, create a Replay View volume from this Replay, and present it to a recovery host also running Solaris. A Replay of a 30GB volume (as shown in the "df -k" output below) will be captured from the host hadrian. A Replay View volume of this Replay will then be made and mapped to the Solaris host maximus.

```
root:~# df -k

Filesystem             1024-blocks        Used   Available Capacity  Mounted on

rpool/ROOT/solaris       286949376     2294824   273864418      1%    /

/devices                         0           0           0      0%    /devices

[snip]

rpool/export             286949376          32   273864418      1%    /export

rpool/export/home        286949376          31   273864418      1%    /export/home

rpool                    286949376          73   273864418      1%    /rpool

/dev/dsk/c4t6d0s2           694700      694700           0    100%
/media/Oracle_Solaris-11_1-Text-SPARC

172.16.26.131:/nfsroot

                         103212320    54758292    43211148     56%    /Tools

/dev/dsk/c0t6000D31000006500000000000000017C5d0s2

                          30979718       30737    30639184      1%    /10g_UFS

Compelnt_pool01           30901248          32    30901104      1%
/Compelnt_pool01

Compelnt_pool01/fs01      30901248          31    30901104      1%
/Compelnt_pool01/fs01
```

When this Replay View volume is presented to the host maximus, the methods as shown in the section Scanning for new volumes is applied to discovery this new volume. Additionally, using the "fsck" utility should also be used as needed to ensure that the volume is in a consistent state before mounting the file system.

```
root:~# cfgadm -c configure c6; cfgadm -c configure c7

root:~# mpathadm list lu

        /dev/rdsk/c0t5000CCA025442A24d0s2

                Total Path Count: 1
                Operational Path Count: 1

[snip]

                Operational Path Count: 2

        /dev/rdsk/c0t6000D3100000650000000000000017CCd0s2

                Total Path Count: 4

                Operational Path Count: 4

        /dev/rdsk/c0t6000D3100000650000000000000017CDd0s2

                Total Path Count: 4

                Operational Path Count: 4

root:~# mkdir /30g_UFS_from_hadrian

root:~# fsck -y /dev/rdsk/c0t6000D3100000650000000000000017CDd0s2

** /dev/rdsk/c0t6000D3100000650000000000000017CDd0s2

** Last Mounted on /10g_UFS

** Phase 1 - Check Blocks and Sizes

** Phase 2 - Check Pathnames

** Phase 3a - Check Connectivity

** Phase 3b - Verify Shadows/ACLs

** Phase 4 - Check Reference Counts

** Phase 5 - Check Cylinder Groups

2 files, 9 used, 30948981 free (13 frags, 3868621 blocks, 0.0% fragmentation)

root:~# mount /dev/dsk/c0t6000D3100000650000000000000017CDd0s2
/30g_UFS_from_hadrian

root:~# df -k

Filesystem           1024-blocks        Used   Available Capacity  Mounted on

rpool/ROOT/solaris     286949376     2200462   273950560     1%    /

[snip]

rpool/export           286949376          32   273950560     1%    /export
```

```
rpool/export/home      286949376          31   273950560     1%    /export/home

rpool                  286949376          73   273950560     1%    /rpool

/dev/dsk/c2t6d0s2         694700      694700           0   100%
/media/Oracle_Solaris-11_1-Text-SPARC

Compelnt_pool01         30901248          32    30901104     1%
/Compelnt_pool01

Compelnt_pool01/fs01    30901248          31    30901104     1%
/Compelnt_pool01/fs01

/dev/dsk/c0t6000D31000006500000000000000017CDd0s2

                        30979718       30737    30639184     1%
/30g_UFS_from_hadrian
```

6.3.3 UFS performance tuning recommendations

Performance tuning is a black art when it comes to file systems. The UFS file system is no different. The performance of a file system is very highly dependent on the actual IO workload occurring on the file systems and detailed consideration must be taken when discussing ways to improve and tune IO performance of a file system.

This paper does not attempt to be the end-all guide to UFS performance tuning; but is instead meant to offer a starting point into understanding the various components which may contribute towards IO performance and how each of these components may be interact with each other on your Solaris host.

There are multiple Oracle and Solaris community guides that may be leveraged, which discuss performance considerations in significantly more detail. This URL provided a very good place to start. http://www.solarisinternals.com/si/reading/fs2/fs2.html

> Performance tuning should be approached in a systematic, methodical fashion; any changes should be applied a single change at a time, followed by a period of monitoring to determine the impact of such change and iteratively applying subsequent changes until the desired conclusions or performance is reached.

6.3.3.1 maxcontig

One of the tunable parameters to the UFS file system which can result in performance improvements is the “maxcontig” setting. This value determines the number of file system blocks which will be read or written in a single operation. The value is set automatically by the Solaris OS when the file system is created and is based on the device characteristics. Adjusting this value can result in improved performance when it is set to a value that is optimal to the IO workload.

The “maxcontig” value can be determined as shown via multiple commands below.

```
root:~# fstyp -v /dev/dsk/c0t6000D3100000650000000000000017C5d0s2 | grep
maxcontig

maxcontig 128   rotdelay 0ms    rps     250

root:~# mkfs -m /dev/rdsk/c0t6000D3100000650000000000000017C5d0s2

mkfs -F ufs -o
nsect=128,ntrack=48,bsize=8192,fragsize=1024,cgsize=16,free=1,rps=250,nbpi=8155,
opt=t,apc=0,gap=0,nrpos=8,maxcontig=128,mtb=n
/dev/rdsk/c0t6000D3100000650000000000000017C5d0s2 62911808
```

The “maxcontig” parameter may be changed as follows.

```
root:~# tunefs -a 256 /dev/dsk/c0t6000D3100000650000000000000017C5d0s2

maximum contiguous block count changes from 128 to 256

root:~# fstyp -v /dev/dsk/c0t6000D3100000650000000000000017C5d0s2 | grep
maxcontig

maxcontig 256   rotdelay 0ms    rps     250
```

6.3.3.2 Direct I/O

This refers to I/O that does not make use of the OS page cache. By default, Direct I/O is not enabled on any Solaris file system, including UFS. However, certain workloads may greatly benefit from using Direct I/O and thus enabling this may result in significant performance improvements. However, keep in mind that with Direct I/O enabled, certain file system features such as read pre-fetching, write coalescing and staged writes may not function, which in return may also impede performance.

For example, if the Solaris host is serving an Oracle RDBMS instance, using Direct I/O may lead to improved performance characteristics. Since Oracle uses its own caching mechanism within its SGA (Shared Global Area), read operations using the Solaris OS page cache can lead to double-cached situations. In addition, there is a higher CPU load when OS page cache is heavily used; thus in situations where a high I/O rate is occurring, the OS locking mechanisms used to maintain cache coherency may actually introduce a throttling effect that is far below what the underlying Storage Center SAN is capable of producing.

The URL below provides further supporting evidence that UFS with Direct I/O enabled may improve IO performance in certain IO workload scenarios.
http://www.solarisinternals.com/wiki/index.php/Direct_I/O

- It eliminates the write-breakup of large synchronous writes, which are important for the log-writer.
- It allows concurrent read/writes to a single file (eliminates the POSIX reader/writer lock)
- It eliminates double buffering
- It provides a small, efficient code path for reads and writes

- It removes memory pressure from the operating environment. Solaris uses the virtual memory system to implement file system caching and, as a result, many page-in operations result when reading data though the file system cache. With UFS direct I/O, database reads do not need to involve the virtual memory system.

The UFS Direct I/O option can be enabled on a per file system basis by appending the "-o forcedirectio" option to the "mount" command as shown below.

```
root:~# mount -o forcedirectio /dev/dsk/c0t6000D310000065000000000000000017C5d0s2
/10g_UFS
```

Every enterprise production workload and environment can vary. It is thus recommended that the implementation of Direct I/O is something which should be thoroughly tested prior to its introduction to any production environment.

6.3.3.3 /etc/system

The /etc/system file contains system wide kernel parameters. This file is read at boot time, therefore any changes made to this file will require a reboot to be enabled. There are a couple of exceptions to this, but generally rebooting the server is recommended when modifying this file. The Oracle Solaris Tuning Guide located at the following URL discusses this in great detail. http://download.oracle.com/docs/cd/E19253-01/817-0404/.

An example of a change which can affect IO performance is to change the maximum SCSI transfer size. To enable this change, add an entry as shown below is appended to the contents of the /etc/system file and a reboot of the Solaris host is then required to activate this change.

```
*

* Allow larger SCSI I/O transfers, parameter is bytes

*

set maxphys = 2097152
```

6.4 ZFS file system

ZFS is a combined file system as well as a logical volume manager developed by Sun. It is natively included in Solaris 10U6 and newer releases. ZFS brings with it the ability to create Exabyte (EB) size files and file systems. The stated maximums for ZFS are 16EB for a maximum file size as well as 16EB for maximum volume size. A ZFS file system is capable of containing a maximum of 2^{48} files. ZFS has a number of advanced file system features such as built-in compression, encryption, deduplication and more. Not all features are available depending on the actual version being used. In addition, some of the advanced features of ZFS are similar to features available in the Dell Storage Center. This section will discuss some of the recommended best practices which should be considered when using ZFS on Solaris host utilizing the Dell Storage Center SAN. For some greater detail on ZFS as well as general ZFS tuning and best practices, refer to the following URLs below.

- Oracle Solaris ZFS Administration Guide
  http://download.oracle.com/docs/cd/E19253-01/819-5461/index.html
- ZFS Best Practices Guide
  http://www.solarisinternals.com/wiki/index.php/ZFS_Best_Practices_Guide
- ZFS Tuning Guide
  http://www.solarisinternals.com/wiki/index.php/ZFS_Evil_Tuning_Guide

The Dell Storage Center stripes and mirrors all data across all drives in the array. Therefore, it is recommended to create large, single volumes for each ZFS zpool created, rather than creating multiple smaller volumes.

Additionally, adding multiple ZFS zpools, each backed by a dedicated large volume, balanced across multiple Storage Center controllers is the recommended approach in achieving optimal IO performance in such an environment.

Note: ZFS file systems are not Thin Reclamation aware and will not participate with the Storage Center Free Space Recovery algorithms.

6.4.1 Mapping a volume for ZFS

The process of mapping a Storage Center volume for ZFS use is identical to the section above. It is recommended to use a single and entire Storage Center volume per ZFS zpool rather than create multiple partitions (slices). A disk label is still required and ZFS requires the use of an EFI label type.

From the “mpathadm list lu” output shown below, the volume shown in RED will be used to create this new ZFS zpool.

```
root:~# mpathadm list lu

        /dev/rdsk/c0t5000C50048697F5Fd0s2

                Total Path Count: 1

                Operational Path Count: 1

        /dev/rdsk/c0t6000D3100000650000000000000017C7d0s2

                Total Path Count: 4

                Operational Path Count: 4

        /dev/rdsk/c0t6000D3100000650000000000000017C6d0s2

                Total Path Count: 4

                Operational Path Count: 4

        [snip]
```

There is no need to explicitly apply an EFI label to the volume, as the process of creating a zpool will automatically apply an EFI label by default.

```
root:~# zpool create –o autoexpand=on Compelnt_pool01
/dev/rdsk/c0t6000D3100000650000000000000017C7d0

root:~# zpool list

NAME              SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT

Compelnt_pool01  9.94G   126K  9.94G   0%  1.00x  ONLINE  -

rpool             278G  12.2G   266G   4%  1.00x  ONLINE  -
```

At this point, ZFS file systems may be created inside of this new zpool. The creation on any new ZFS file system is also automatically (thin) sized and mounted as default. These defaults may be changed with the use of additional command line parameters issued to the “zfs” command.

```
root:~# zfs create Compelnt_pool01/fs01

root:~# zfs list

NAME                     USED  AVAIL  REFER  MOUNTPOINT

Compelnt_pool01          124K  9.78G    32K  /Compelnt_pool01

Compelnt_pool01/fs01      31K  9.78G    31K  /Compelnt_pool01/fs01

root:~# df -k

Filesystem           1024-blocks        Used   Available Capacity  Mounted on

rpool/ROOT/solaris     286949376     2294563   273846614      1%    /

[snip]

rpool/export           286949376          32   273846614      1%    /export

rpool/export/home      286949376          31   273846614      1%    /export/home

rpool                  286949376          73   273846614      1%    /rpool

/dev/dsk/c4t6d0s2         694700      694700           0    100%
/media/Oracle_Solaris-11_1-Text-SPARC

172.16.26.131:/nfsroot

                       103212320    54758292    43211148     56%    /Tools

/dev/dsk/c0t6000D3100000650000000000000017C5d0s2

                        30979718       30737    30639184      1%    /10g_UFS
```

```
Compelnt_pool01        10257408          32     10257283     1%
/Compelnt_pool01

Compelnt_pool01/fs01   10257408          31     10257283     1%
/Compelnt_pool01/fs01
```

6.4.2 Expanding a ZFS file system

There are a few ways to expand a ZFS file system on a Solaris host. The current implementation of ZFS on Solaris (as of this writing) does not yet support the online expansion of a zpool device that has had its underlying volume expanded in size on the Dell Storage Center. Therefore, simply performing a volume expansion on the Storage Center volume and then expanding the file system on the Solaris host would not work. There are a couple of options available to expand the available storage of an existing ZFS pool. These methods are discussed below.

There is always certain risk of data loss whenever the underlying file system is modified. Before executing any of the steps below, make sure to have good and verifiable backups of your data.

6.4.2.1 zpool add

The first option for expanding a ZFS volume involves simply creating a new volume on the Storage Center, mapping it to the Solaris host and then adding the new volume to the existing ZFS pool.

In this example, the current zpool Compelnt_pool01 will be expanded from its current capacity of 10GB to 20GB. This will be achieved by adding the device c0t6000D31000006500000000000000017C6d0s2 (shown in RED in the mpathadm output) to this zpool.

```
root:~# mpathadm list lu

        /dev/rdsk/c0t5000C50048697F5Fd0s2

                Total Path Count: 1

                Operational Path Count: 1

        /dev/rdsk/c0t6000D31000006500000000000000017C7d0s2

                Total Path Count: 4

                Operational Path Count: 4

        /dev/rdsk/c0t6000D31000006500000000000000017C6d0s2

                Total Path Count: 4

                Operational Path Count: 4

         [snip]

root:~# zfs list
```

```
NAME                        USED  AVAIL  REFER  MOUNTPOINT

Compelnt_pool01             124K  9.78G    32K  /Compelnt_pool01

Compelnt_pool01/fs01         31K  9.78G    31K  /Compelnt_pool01/fs01

[snip]

root:~# zpool add Compelnt_pool01 c0t6000D3100000650000000000000017C6d0

root:~# zfs list

NAME                        USED  AVAIL  REFER  MOUNTPOINT

Compelnt_pool01             124K  19.6G    32K  /Compelnt_pool01

Compelnt_pool01/fs01         31K  19.6G    31K  /Compelnt_pool01/fs01

[snip]
```

It should be noted that with this approach, the zpool will become a concatenation of multiple Storage Center volumes backing it. The space in the zpool will be consumed one volume at a time instead of in a striped fashion. In contrast, any volumes presented from the Storage Center is striped across all drives within that same tier of storage.

Once the space is consumed on one of the volumes, all I/O will occur to the remaining volumes with free space. Therefore, when using multiple volumes in a single zpool, it is recommended to balance the volumes equally across both Storage Center controllers. This will help optimize performance as IO load can then be spread across both controllers.

6.4.2.2 zpool replace

Another option for allocating more space and expanding an existing ZFS file system is to create a new volume on Storage Center with the new capacity the ZFS file system is required to have. A "zpool replace" operation is then applied, which will migrate the data from the original volume to the new volume. This has the additional advantage of managing the number of volumes inside a ZFS zpool. The example below demonstrates this process.

Before attempting the "zpool replace" operation, the volume c0t6000D3100000650000000000000017C6d0 has been expanded from 10GB to 30GB on the Storage Center. Additionally, the Solaris host has been made aware of this capacity increase via the methods shown in section Expanding a UFS file system. This volume will be used to replace c0t6000D3100000650000000000000017C7d0 (10GB) which currently backs the ZFS zpool Compent_pool01.

```
root:~# zfs list

NAME                        USED  AVAIL  REFER  MOUNTPOINT

Compelnt_pool01             126K  9.78G    32K  /Compelnt_pool01
```

```
Compelnt_pool01/fs01      31K  9.78G    31K  /Compelnt_pool01/fs01

[snip]

root:~# zpool replace Compelnt_pool01 c0t6000D31000006500000000000000017C7d0
c0t6000D3100000   6500000000000000017C6d0
```

During the resilvering process, the status of the “zpool replace” operation can observed using the following command. In this scenario, and due to the lack of any actual data inside the zpool; the operation completed extremely quickly.

```
root:~# zpool status Compelnt_pool01

  pool: Compelnt_pool01

 state: ONLINE

  scan: resilvered 129K in 0h0m with 0 errors on Thu Oct 17 15:03:14 2013

config:


        NAME                                     STATE     READ WRITE CKSUM

        Compelnt_pool01                          ONLINE       0     0     0

          c0t6000D31000006500000000000000017C6d0  ONLINE       0     0     0


errors: No known data errors

root:~# zfs list

NAME                     USED  AVAIL  REFER  MOUNTPOINT

Compelnt_pool01          144K  29.5G    32K  /Compelnt_pool01

Compelnt_pool01/fs01      31K  29.5G    31K  /Compelnt_pool01/fs01

[snip]
```

It should be noted that the above process may potentially take a long time, should there be large amounts of data stored in the zpool. Additionally, this operation can be extremely IO intensive and due consideration should be made to accommodate any other ongoing production IO business needs.

Starting with Solaris 10U9, the ZFS software was updated to include the ability to expand zpool capacity dynamically and online. This method is achieved with the command “zpool online -e <Pool Name> <existing volume/device name>”.

6.4.3 ZFS and Data Instant Replay

Data Instant Replay (DIR) is a core feature of the Dell Storage Center and its usage can dramatically improve efficiency when it comes to facilitating backup/recovery tasks, sharing production data with other environments (e.g. development, system test, QA, etc.). However, due to the manner in which ZFS stores metadata pertaining to the zpool, it is not possible to remap a Replay View volume of a ZFS file system back to the originating system while the production zpool is active; thus complicating the usage of DIR with ZFS.

As with the UFS file system, the recommended practice is to map Replay View volumes to a host other than to the originating host. However, where it is possible to mount a UFS based Replay View volume back to the originating host (with care), this is not possible with ZFS based zpools due to the metadata which ZFS embeds into each volume which makes up the zpool.

When using multiple Dell Storage Center volumes to make a ZFS zpool, configure these volumes to be contained within a logical Storage Center Consistency Group. This will ensure that all volumes in the zpool are successfully snapped (imaged) with Data Instant Replay at the same point-in-time so as to maintain a consistent zpool state. Refer to the Storage Center Administrator's Guide for further details on Consistency Groups and its proper usage.

6.4.3.1 Mapping a Replay View of a ZFS file system

The example below demonstrates the process used to capture a Replay of a ZFS zpool on the Solaris host hadrian. A Replay View volume is then created and presented to an alternate Solaris host name maximus. Since this zpool currently consists of a single Storage Center volume, the use of Consistency Groups will not be shown in this instance.

On the Solaris host maximus, this Replay View volume will be discovered (as shown in RED below). The ZFS import command will (by default) warn that the inbound ZFS zpool may be potentially owned by a different host; this is normal protocol and behavior of the ZFS file system. This warning message may be bypassed with the use of the "-f" parameter to the "zpool import" command as shown below.

The Replay View volume is identified by its Storage Center Serial Number as shown below.

```
root:~# mpathadm list lu

        /dev/rdsk/c0t5000CCA025442A24d0s2

                Total Path Count: 1

                Operational Path Count: 1

        /dev/rdsk/c0t6000D3100000650000000000000017CAd0s2

                Total Path Count: 4

                Operational Path Count: 4

        /dev/rdsk/c0t6000D310000067000000000000000B32d0s2
```

```
                Total Path Count: 2

                Operational Path Count: 2

        /dev/rdsk/c0t6000D31000006500000000000000017CCd0s2

                Total Path Count: 4

                Operational Path Count: 4
```

At this point, the "zpool import" command is issued; once to demonstrate the ZFS warning message, then once more with the "-f" parameter given.

```
root:~# zpool import Compelnt_pool01

cannot import 'Compelnt_pool01': pool may be in use from other system,

it was last accessed by hadrian (hostid: 0x860cb606) on Thu Oct 17 15:03:20 2013

: pool may already be in use

root:~# zpool import -f Compelnt_pool01

root:~# zfs list

NAME                     USED  AVAIL  REFER  MOUNTPOINT

Compelnt_pool01          144K  29.5G    32K  /Compelnt_pool01

Compelnt_pool01/fs01      31K  29.5G    31K  /Compelnt_pool01/fs01

[snip]

root:~# df -k

Filesystem           1024-blocks        Used   Available Capacity  Mounted on

rpool/ROOT/solaris     286949376     2200356   273955659      1%    /

[snip]

rpool/export           286949376          32   273955659      1%    /export

rpool/export/home      286949376          31   273955659      1%    /export/home

rpool                  286949376          73   273955659      1%    /rpool

/dev/dsk/c2t6d0s2         694700      694700           0    100%
/media/Oracle_Solaris-11_1-Text-SPARC

Compelnt_pool01         30901248          32    30901104      1%
/Compelnt_pool01
```

```
Compelnt_pool01/fs01     30901248           31     30901104      1%
/Compelnt_pool01/fs01
```

6.4.4 ZFS performance tuning recommendations

ZFS does an incredibly good job of auto-tuning itself and using its built-in algorithms to provide optimal IO performance. There are several places in which advanced performance tuning of ZFS is discussed in greater detail, one such location is shown in the URL below.
http://www.solarisinternals.com/wiki/index.php/ZFS_Evil_Tuning_Guide

Before making any changes to a ZFS device and/or file system, it is always prudent to verify that good and recoverable backups of the data exists.

Some areas of consideration from the perspective of ZFS on Dell Storage Center is discussed below. These are starting points of discussion, and as such, any modifications made should be thoroughly tested before implementing it in production environment.

Performance tuning should be approached in a systematic, methodical fashion; any changes should be applied a single change at a time, followed by a period of monitoring to determine the impact of such change and iteratively applying subsequent changes until the desired conclusions or performance is reached.

6.4.4.1 ZFS Intent Log (ZIL)

Traditionally, it has been recommended with other storage platforms/designs to dedicate spindles for ZIL. With Storage Center it is not necessary to store ZIL on separate storage. The Dell Storage Center writes all new data to Tier 1 by default (if using the Recommended Tier Storage Profile). In addition all data is striped across all drives in the tier. Therefore, placing ZIL data on separate volumes does not lead to any notable performance improvements. Allowing Storage Center to automatically manage the best placement for new data is the recommended best practice.

6.4.4.2 Cache Flush

Storage Center will always return acknowledgements from writes to stable storage.

6.4.4.3 Multiple Storage Center volumes in ZFS pool

There may be certain performance benefits gained when configuring multiple Storage Center volumes balanced across both Storage Center controllers. By using multiple volumes across both controllers, more IO paths are created. System administrators should also note that care be taken not to set Fibre Channel timeouts too low. If these thresholds are set too low (about 30 seconds or less) when a controller failover event occurs, the volumes on that downed controller may not have sufficient time to relocate themselves to the other online controller; this may create upstream impact by way of SCSI timeouts and errors and therefrom cause ZFS based IO to fail or even the potential for ZFS pools to go offline.

7 Boot from SAN

This section will briefly discusses some of the considerations around Solaris boot-from-SAN with the Dell Storage Center.

Beginning with SolarisU2 and newer releases, boot-from-SAN functionality is supported using the UFS file system only. The HBAs also need to have the necessary firmware release which supports boot-from-SAN as well.

Beginning with Solaris 10U6 and newer releases, both UFS and ZFS file systems are available as root file system options for the Solaris host. Either will work as expected with the Dell Storage Center. Refer to the UFS and ZFS sections in this paper for specific considerations with regard to each of these file system types.

Prior to beginning the installation, verify that the QLogic or Emulex HBAs used is patched to required versions of their driver and firmware code levels in order to support boot-from-SAN.

Either single path or multipath can be used to install the Solaris OS via boot-from-SAN. If single path is used during installation, multipath can be enabled by adding the proper Dell device entries to the /etc/driver/drv/scsi_vhci.conf and rebooting accordingly. Afterwhich, add the additional path and it will be added by the Solaris MPxIO management upon discovery. It is considered best practice to configure multipath during OS installation whenever possible.

Prior to starting the installation it is necessary to create the Server Object for the Solaris host with the Dell Storage Center GUI. In order for the WWPNs to appear, the HBA ports must be activated so that Storage Center can be made aware of them. There are a couple of options for accomplishing this and are discussed below.

- Use the Open Boot Prom commands "show-devs", "dev <device>", and ".properties" to display the WWPN information for the HBA ports. Open the Storage Center GUI and create the Server Objects referencing the WWPNs discovered.
- Boot the Solaris host using either the Installation DVD or a jumpstart server. During the boot process, the server will scan for all devices on all HBA ports. When the scan is complete, go to the Storage Center GUI and create the Server Object using the WWPNs discovered.

After creating the Server Object, create the volume on Storage Center, it must be mapped as LUN 0 in order for it to be properly discovered by the Solaris host as a bootable device.

It is recommended to have only the boot/root OS volume mapped to the host during the boot-from-SAN installation. This will reduce confusion and remove the risk of accidentally partitioning the incorrect device which, in a worst case scenario, could cause accidental data loss.

When the Solaris installation reaches the point where it will want partition the drive, the installation program will fail with the error message.

```
One or more disks are found,
but one of the following
problems exists:

Hardware failure
Unformatted disk.
```

This error is due to the fact that the volume does not yet have a proper disk label. The Solaris installer will only interact with devices that have been labeled with a Solaris disk label. The Solaris installation program will exit to a shell prompt. Use the “format” utility at this point to label the device.

In a multipath installation there will be two devices listed. You will only need to select and label either one of the two devices. Once labeled, exit out of the “format” utility and then resume the Solaris installation by issuing the “suninstall” command and then pressing enter. The Soalris interactive installer will then resume where it left off.

8 All about CompCU

The Dell Storage Center SAN can have many of its daily functions managed through a remote command utility called the Dell Compellent Command Utility (CompCU). This allows for scripting and automation integration of SAN tasks between the Solaris OS and Storage Center. CompCU is a java-packaged application and thus requires the installation of Java on the Solaris host. CompCU can be used to script common administrative tasks which can be tremendous time savers and provide a consistent framework for managing Storage Center volumes and Replays.

CompCU requires the host to have the proper Java release installed. Refer to the Command Utility User Guide (in Appendix A) for more details. The CompCU.jar object can be downloaded from the Dell support site. Once installed on the Solaris host, this tool can be used to perform Storage Center tasks from the shell prompt, which can be incorporated into new and/or existing user management scripts. Outlined below are some common use cases for CompCU.

- Creating Volumes, mapping to the server.
- Taking replays, recovering replays, etc.

The examples below do not cover the full breadth of the usefulness of CompCU by any means; they are designed to give an initial insight into the types of tasks which may be automated with CompCU.

8.1 Verifying Java, configuring and testing CompCU functions

The first task is to install the required (or newer) version of Java onto the Solaris host. CompCU requires a Java RTE v1.6.x or newer. The Java runtime may have already been installed with the OS and can be verified with the command shown below.

```
root:~# /usr/bin/java -version

java version "1.7.0_07"

Java(TM) SE Runtime Environment (build 1.7.0_07-b10)

Java HotSpot(TM) Server VM (build 23.3-b01, mixed mode)
```

Download the CompCU package from the Dell support site. The package will include a PDF User Guide as well as the CompCU.jar file required. Locate this CompCU.jar in a logical file system location. Verify that CompCU is working with Java by executing the command below to display the help and usage syntax.

```
root:/Tools/CompCU# /usr/bin/java -jar ./CompCU.jar -h

Compellent Command Utility (CompCU) 6.4.1.1



usage: java -jar CompCU.jar [Options] "<Command>"
```

```
 -c <arg>                Run a single command (option must be within
                         quotes)
 -default                Saves host, user, and password to encrypted file
 -defaultname <arg>      File name to save default host, user, and password
                         encrypted file to
 -file <arg>             Save output to a file
 -h                      Show help message
 -host <arg>             IP Address/Host Name of Storage Center Management
                         IP
 -password <arg>         Password of user
 -s <arg>                Run a script of commands
 -user <arg>             User to log into Storage Center with
 -verbose                Show debug output
 -xmloutputfile <arg>    File name to save the CompCU return code in xml
                         format.  Default is cu_output.xml.
[snip]
```

To facilitate the ease of access in using CompCU, the tool can be initially run with the "-default" switch to configure an encrypted password file as shown below. A file named "default.cli" is created in the local directory; this file may be renamed as required for clarity and usage.

```
root:/Tools/CompCU# /usr/bin/java -jar ./CompCU.jar  -default -host 172.16.2.109
-user Admin -password XXX

Compellent Command Utility (CompCU) 6.4.1.1


================================================================================
=================

User Name:               Admin

Host/IP Address:         172.16.2.109
```

```
================================================================================
=================

Connecting to Storage Center: 172.16.2.109 with user: Admin

java.lang.IllegalStateException: TrustManagerFactoryImpl is not initialized

Saving CompCu Defaults to file [default.cli]...
```

The “default.cli” file may then be referenced in other commands to login to the same Storage Center and perform tasks. A separate .cli file may be created for each Storage Center under management with each containing the appropriate login credentials for the respective Storage Center array. The example below demonstrates a “volume show” command applied to the Storage Center located at IP address 172.16.2.109.

```
root:/Tools/CompCU# /usr/bin/java -jar ./CompCU.jar  -defaultname default.cli -
host 172.16.2.109 -user Admin -password XXX -c "volume show"

Compellent Command Utility (CompCU) 6.4.1.1


================================================================================
=================

User Name:              Admin

Host/IP Address:        172.16.2.109

Single Command:         volume show

================================================================================
=================

Connecting to Storage Center: 172.16.2.109 with user: Admin

java.lang.IllegalStateException: TrustManagerFactoryImpl is not initialized

Running Command: volume show


Index  Name                             Status   ConfigSize      ActiveSize
ReplaySize      Folder
StorageProfile                   DeviceI              D
SerialNumber      ConfigSizeBlock ActiveSizeBlock ReplaySizeBlock
MaxWriteSizeBlo ReadCache  WriteCache

------ -------------------------------- -------- --------------- ---------------
--------------- ----------------------------------------------------------------
```

```
-------------------------------- -------              -------------------------
----------------- --------------- --------------- --------------- --------------
- ---------- ----------

283    Fal-asm-mirror-test-failgroup1   Up       100.00 GB       31.33 GB
0.00 KB         Oracle/11gR2/ASM-Mirror
Recommended                      6000d31              000006500000000000000012f
00000065-0000012f 209715200       65712128        0               0
Enabled    Enabled

290    Fal-asm-mirror-sp-fg1            Up       100.00 GB       7.94 GB
0.00 KB         Oracle/11gR2/ASM-Mirror
Recommended                      6000d31              0000065000000000000000136
00000065-00000136 209715200       16658432        0               0
Enabled    Enabled

824    ibmsvc00-managed-mdisk1          Up       500.00 GB       98.02 GB
0.00 KB         IBMSVC
Recommended                      6000d31              000006500000000000000034c
00000065-0000034c 1048576000      205561856       0               0
Enabled    Enabled

[snip]

Successfully finished running Compellent Command Utility (CompCU) application.
```

8.2 Using CompCU to automate common tasks

This section illustrates some use cases with managing Storage Center tasks with CompCU on Solaris. As mentioned above, these examples are indicative of the types of tasks which can be easily accomplished from the Solaris shell prompt using CompCU and only meant as a starting point to familiarize the system administrator with this powerful tool set.

8.2.1 Creating a single volume with CompCU

This example demonstrates using CompCU to create a single 100GB named “hadrian_100g_00” Storage Center volume from the Solaris host, placed in the Storage Center folder named “Solaris” and then subsequently mapped to the Solaris host named “hadrian”.

```
root:/Tools/CompCU# /usr/bin/java -jar ./CompCU.jar  -defaultname default.cli -
host 172.16.2.109 -user Admin -password XXX -c "volume create -name
hadrian_100g_00 -folder Solaris -server hadrian -size 100g"
```

8.2.2 Creating a Replay and a Replay View with CompCU

This example demonstrates creating a Replay “hadrian_100g_00_Replay” of an existing volume “hadrian_100g_00” on Storage Center, creating a Replay View named “hadrian_100g_00_View” from this mentioned Replay and finally mapping the Replay View to a Solaris host named “maximus”, all with a single CompCU command.

```
root:/Tools/CompCU# /usr/bin/java -jar ./CompCU.jar  -defaultname default.cli -
host 172.16.2.109 -user Admin -password XXX -c "replay create -volume
'hadrian_100g_00' -name 'hadrian_100g_00_Replay' -view 'hadrian_100g_00_RpView'
-server 'maximus'”
```

8.2.3 Rapid deployment of multiple volumes with CompCU

This final example demonstrates using CompCU for rapid volume creation and deployment from Storage Center and mapping these volumes to the Solaris host “maximus”.

```
root:/Tools/CompCU# for i in 0 1 2 3 4 5 6 7 8 9; do /usr/bin/java -jar
./CompCU.jar  -defaultname default.cli -host 172.16.2.109 -user Admin -password
XXX -c "volume create –name maximus_10g_0${i} –folder Solaris -server 'maximus'
–size 10g”; done
```

A Additional resources

Dell Storage Center 6.3 Administrator's Guide
http://kcint.compellent.com/Published%20Documents/680-019-013.pdf

SC8000 Connectivity Guide
http://kcint.compellent.com/Knowledge%20Center%20Documents/680-027-013.pdf

Dell Compellent Command Utility (CompCU) Reference Guide
http://kcint.compellent.com/Knowledge%20Center%20Documents/680-018-007.pdf

Oracle Solaris 11 (11/11 Release) Information Library
http://docs.oracle.com/cd/E23824_01/

Oracle Solaris Tunable Parameters Reference Guide
http://docs.oracle.com/cd/E19253-01/817-0404/

ILOM and ALOM Command Comparison
http://docs.oracle.com/cd/E19745-01/820-7145-11/z400018e1008787.html

B Configuration details

Table 1 Component table

Component	Description
Operating system	Oracle Solaris 11.1 SPARC
Driver version	driver/fc/qlc Build Release 5.11
Firmware version	0.5.11
Application	Oracle Solaris 11.1 SPARC
Cabling	4GB Fibre Channel
Server	Oracle Sun SPARC T4-1
Storage	Dell Compellent Storage Center SCOS 6.4.10, Virtual port mode
Switch	Cisco MDS 9148