



Technical Report

NetApp Snap Creator Framework Integration with SAP IQ

João Aragão and Neto from Brazil, NetApp
April 2017 | TR-4583



TABLE OF CONTENTS

1	Executive Summary.....	4
1.1	Document Organization	4
1.2	Audience	4
1.3	Snap Creator Framework Overview.....	4
2	Architecture Details.....	6
3	Recommendations.....	7
3.1	Hardware and Software Requirements	7
3.2	IMT	7
3.3	Test Preparation	8
4	Basic Infrastructure Validation	8
4.1	License Check	8
4.2	SAP IQ Client Requirements.....	9
5	Database Backup.....	12
5.1	Snapshot Copy–Based Backup of SAP IQ Database	12
5.2	Snapshot Copy–Based Backup of SAP IQ Database During Reads and Writes	25
6	Integrity Check of Snapshot Copy–Based Backup During Read-Write Process	29
6.1	Test Objectives	29
6.2	Test Procedure	29
6.3	Test Results	54
7	Integrity and Security Check During Simultaneous Reads and Writes Between Source and Cloned Databases.....	54
7.1	Snapshot Copy of SAP IQ Cloned Database	55
8	Verification of Snapshot Copy Creation Time and Size	58
8.1	Test Objectives	58
8.2	Test Procedure	58
9	Compression and Deduplication	60
9.1	Test Objective	60
9.2	Test Procedure	60

References.....	63
Version History	63

LIST OF TABLES

Table 1) Software versions.....	7
---------------------------------	---

LIST OF FIGURES

Figure 1) Snap Creator Framework architecture.	5
Figure 2) NetApp's backup environment for SAP IQ.....	6

1 Executive Summary

This report identifies the NetApp® ONTAP® resources such NetApp Snap Creator® Framework, NetApp SnapDrive® and NetApp SnapVault®, and NetApp Snapshot®, deduplication and compression, which integrate with the SAP IQ database (data warehouse) to minimize volumetry and the backup time. It describes in detail the test cases associated with each of the resources. By completing the test cases, the users can declare that ONTAP fits their application requirements and is qualified for use in real-time deployments. Although most test cases are covered, specific environmental or operational tests might require more personalization.

The tests described in document are valid for all versions of ONTAP and newer versions of the FAS systems. You can verify the compatibility of hardware and software in your setup by using the [NetApp Interoperability Matrix Tool](#).

A version of this document is also available for reference at the [SAP blog](#) page.

1.1 Document Organization

The document is organized to provide prescriptive guidance for each feature and its associated test cases, which can be executed in a qualification environment. It allows you the flexibility to pick and choose the test cases and build a qualification framework as desired by the users. The document also covers the architectural details of the given solution and NetApp and third-party components.

1.2 Audience

This document can be used as a reference by partner engineers and customers who are associated with IT qualifications projects.

1.3 Snap Creator Framework Overview

Snap Creator Framework helps to automate storage provisioning and simplify storage management in NetApp storage environments. It offers the following key benefits:

- Simplified provisioning from the host with a protocol-agnostic approach
- Automated backup and restore
- Snapshot management with OS/application-consistent Snapshot copies

Snap Creator Framework provides a layer of abstraction between an application running on the host operating system and the underlying NetApp storage systems. It has the unique advantage of understanding the host operating system, the file system, and NetApp storage. Thus, Snap Creator Framework has the ability to coordinate operations between these core layers and use SnapDrive in this process. One notable feature of SnapDrive lies in the area of Snapshot copy management. SnapDrive can provide host-/application-consistent Snapshot copies and crash-consistent Snapshot copies. When Snapshot copies of an application dataset that spans multiple volumes or storage systems are required, SnapDrive provides consistency by synchronizing the data in the file system cache and freezing I/O operations to the requested LUNs.

What makes SnapDrive interesting is the mechanism with which it suspends I/O. SnapDrive freezes I/O operations to the LUNs at the host OS/file system layer or directly within the NetApp storage system. If a storage-based I/O fencing mechanism is elected, then SnapDrive relies on the native consistency group feature of ONTAP.

Starting from SnapDrive for UNIX 2.2, SnapDrive supports the feature of consistency groups provided by ONTAP (beginning with version 7.2 and later). This feature is necessary for creating a consistent Snapshot copy across multiple controllers/volumes. In an environment where all participating controllers support consistency groups, SnapDrive uses a ONTAP consistency group as the preferred (default) method to capture multicontroller/multivolume Snapshot copies.

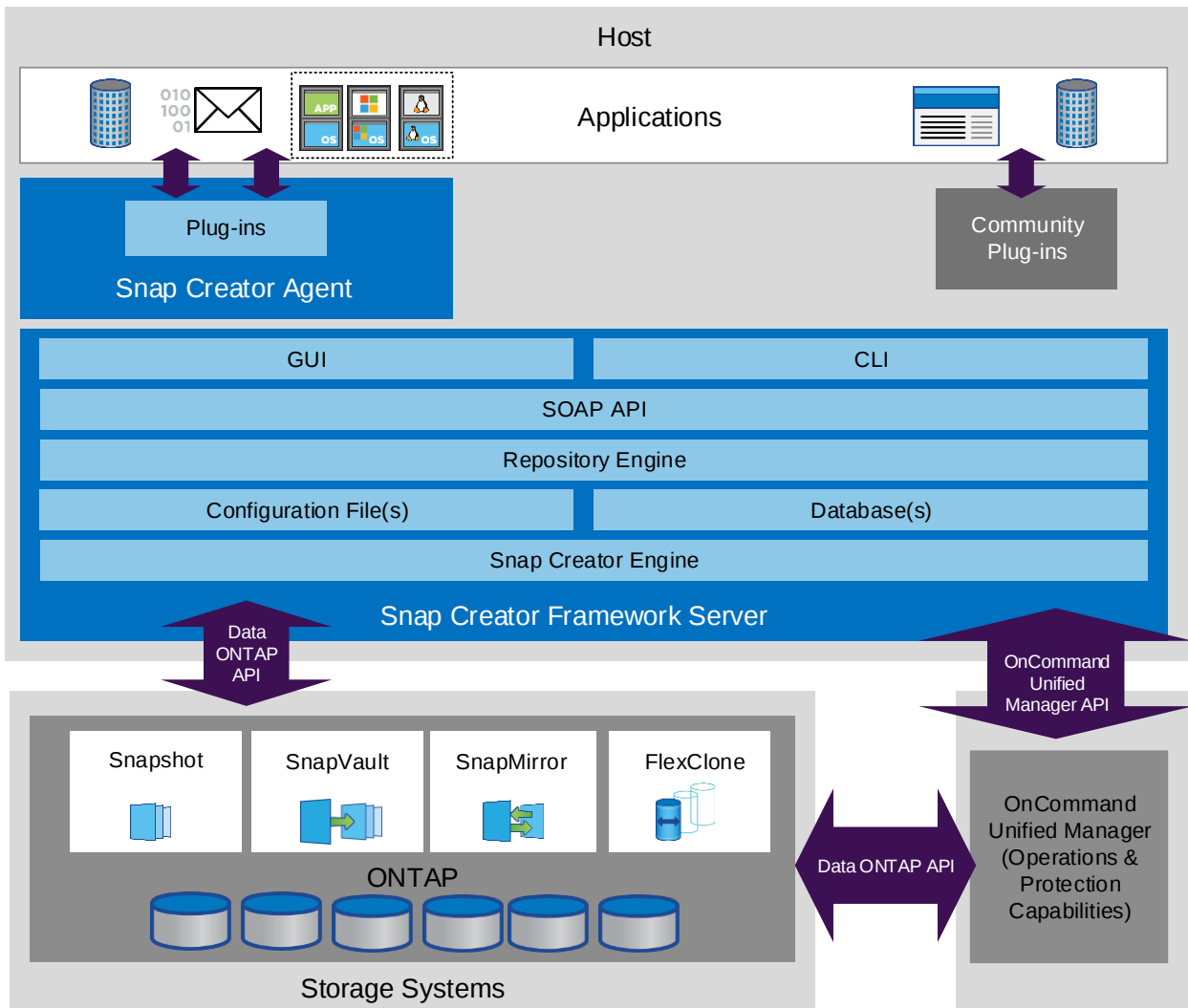
If using a consistency group is not possible or is perhaps undesirable, such as when one of the controllers does not support consistency groups or more than one application is sharing LUNs off a single volume where volume I/O fencing might influence other applications, then SnapDrive resorts to the I/O suspension capability offered at the host level. In such a scenario, SnapDrive attempts to use the lowest-level freezing mechanism available on the host to enforce consistency, which might require direct interaction with the file systems or volume managers. This approach suggests that consistency should always be enforced at the lowest possible level of the storage stack to minimize the duration of I/O suspension.

SnapDrive dramatically simplifies the use of consistency groups. When the file specs or file system dictates a Snapshot copy that spans multiple volumes and controllers, and all target controllers support consistency groups, SnapDrive automatically recognizes this requirement and creates consistency groups to enable crash-consistent Snapshot copies. No change to the SnapDrive syntax is necessary to take advantage of the consistency groups.

To facilitate these tests in this scenario, the tests used Python scripts based on NetApp SDK APIs. The same functionality can be accomplished using the NetApp Snap Creator Framework.

Figure 1 shows the logical architecture of the NetApp Snap Creator Framework.

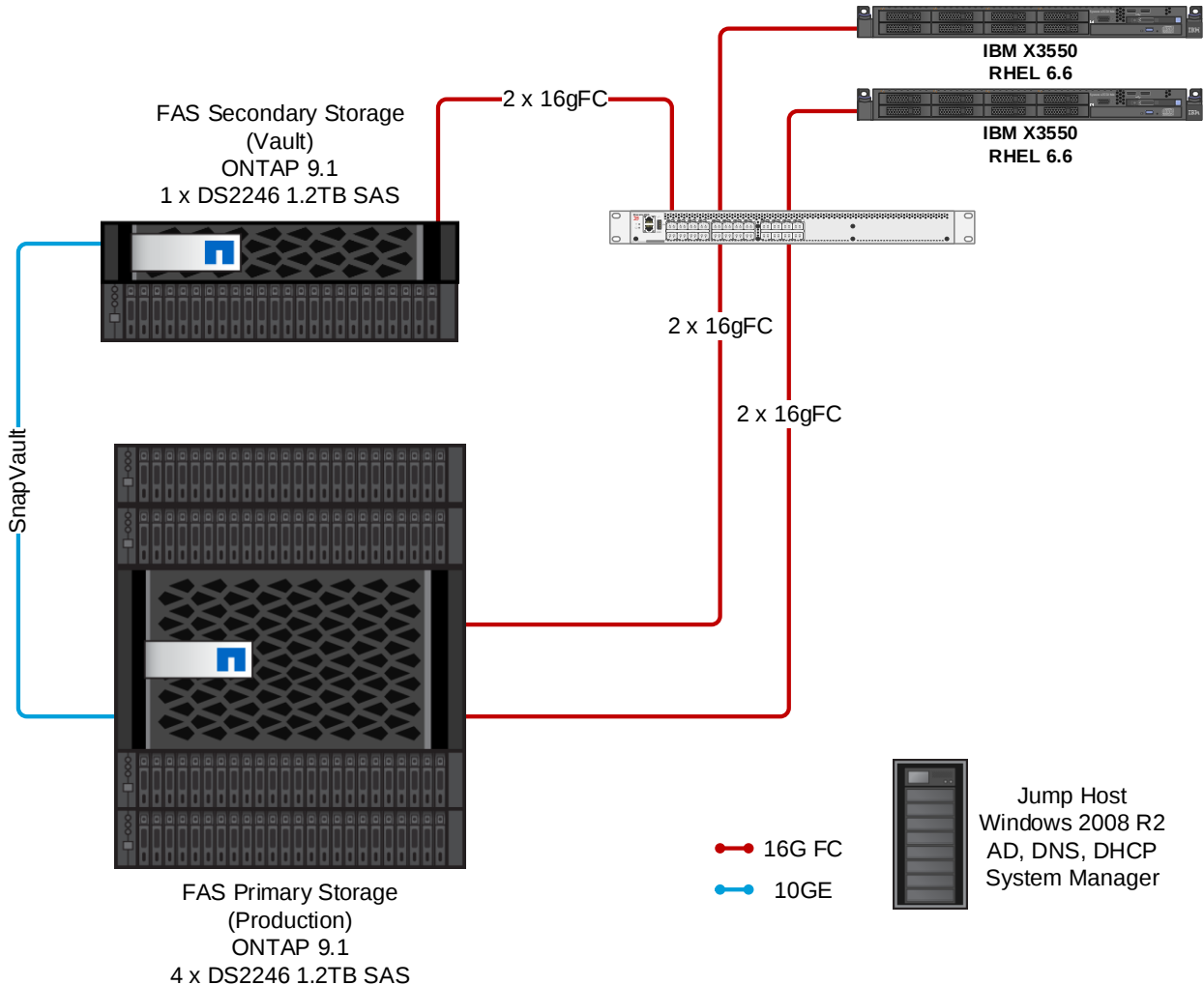
Figure 1) Snap Creator Framework architecture.



2 Architecture Details

Figure 2 illustrates the SAP IQ v16 architecture in terms of hardware, storage and database layout, and configuration. As shown in the figure, the SAP IQ databases are stored in a NetApp SAN environment. The SAP IQ database software is deployed on the RHEL 6.6 servers. The NetApp SnapDrive and NetApp Snap Creator Framework for UNIX software is deployed on each SAP node for backup, restore, and clone operation on the SAP IQ database.

Figure 2) NetApp's backup environment for SAP IQ.



Our test setup uses the following hardware:

- FAS8060HA (10.61.110.100)
- FAS3270 (10.61.110.110)
- Server 1 (S1): IBM X3550 running RHEL6.6 and SAP IQ v16, 24xCPU, 96GB RAM
- Server 2 (S2): IBM X3550 running RHEL6.6 and SAP IQ v16, 24xCPU, 96GB RAM

The infrastructure and all tests were performed in our Customer Proof of Concept (CPOC) Labs facility in RTP (Research Triangle Park), NC – USA.

3 Recommendations

This section details the recommended hardware and software configurations that are necessary to validate the test framework. It also lists the supported hardware and software compatibility between NetApp and third-party components.

Note: The tests described in document are valid for all versions of ONTAP and newer versions of the FAS systems. You can verify compatibility by using the [NetApp Interoperability Matrix Tool](#).

3.1 Hardware and Software Requirements

The following hardware is required to set up the basic configuration for the test:

- Two nodes, one for production and the other to receive data from SnapVault, installed with ONTAP 9.1 on supported hardware with required licenses.
- Our test setup uses 30TB usable space in the first controller (FAS8060HA) and 2TB in the secondary SnapVault controller (FAS3270).
- One database with 20TB (testen2) and another with 1TB (testen1) in both server 1 and controller 1 (FAS8060HA). Only the smallest database is replicated to the second NetApp controller (FAS3270).
- FC switches and port connectivity to storage and hosts.
- SAP IQ v16 installed on two physical hosts with RHEL 6.6 (bare-metal installation).
- FC HBA initiators on host (for SAN environment).

Table 1 lists the software used in the test setup.

Table 1) Software versions.

Software	Version
ONTAP	9.1
SnapDrive for UNIX	5.3
Snap Creator Framework	4.3
SAP IQ	V16 SP11
RHEL	6.6

3.2 IMT

The [NetApp Interoperability Matrix Tool](#) was used to validate the supportability of the following combinations:

- ONTAP (software)
- RHEL OS version 6.6
- SAP IQ database (software)
- SnapDrive and Snap Creator Framework for UNIX
- FC HBA model, driver, and firmware
- FC switch mode and firmware version

3.3 Test Preparation

Table 2) Test prerequisites.

Prerequisites
• ONTAP 9.1 installed on all controllers • All the required licenses installed (FCP, SnapVault, NetApp FlexClone®) • Root and data aggregates created • Data storage virtual machines (SVMs, formerly called Vservers) with required volumes/LUNs • Data LIFs created for client accessibility • Redundant network paths set up with required zoning configuration • LUN masking in storage array • Servers installed with RHEL 6.6 and required patches • Servers installed with SAP IQ v16 SP11 • Databases and instances created on SAP IQ servers • Snap Creator Framework and SnapDrive software installed on SAP IQ servers

4 Basic Infrastructure Validation

This section consists of test cases required to check the status of Snap Creator Framework for SAP IQ environment (SAP IQ database, storage licenses, Snap Creator Framework) prior to demonstration of backup, restore, and clone capability using Snap Creator Framework for SAP IQ.

4.1 License Check

Test Objective

Verify if the required licenses are installed on the ONTAP storage system.

Test Procedure

1. Run the following command from the controller CLI:

superman::> system license show			
Serial Number: 1-80-000099			
Owner: superman			
Package	Type	Description	Expiration
Base	license	Cluster Base License	-
NFS	site	NFS License	-
CIFS	site	CIFS License	-
iSCSI	site	iSCSI License	-
FCP	site	FCP License	-
SnapRestore	site	SnapRestore License	-
SnapMirror	site	SnapMirror License	-
SnapVault	site	SnapVault License	-
FlexClone	site	FlexClone License	-

Results

The required licenses and its validity are displayed.

4.2 SAP IQ Client Requirements

This section provides the test cases to be executed at the client (SAP IQ) side.

Test Objective

Check the environment variables and accessibility of the SAP IQ database.

Test Procedure

1. Check all the environment variables in both servers.

A specific user account created named `sybase_user` was used instead of root user.

```
[sybase_user@sX ~]$ cat .bash_profile
# .bash_profile
# Get the aliases and functions
if [ -f ~/.bashrc ]; then
. ~/.bashrc
fi
# User specific environment and startup programs
PATH=$PATH:$HOME/bin
export PATH
. /database/asiql60/IQ-16_0/IQ-16_0.sh
[sybase_user@sX ~]$ export
declare -x COCKPIT JAVA_HOME="/database/asiql60/shared/SAPJRE-8_1_008_64BIT" ...
declare -x HOME="/home/sybase_user"
declare -x HOSTNAME="s1"
declare -x INCLUDE="/database/asiql60/OCS-15_0/include:" declare -x
IQDIR16="/database/asiql60/IQ-16_0"
...
declare -x LD_LIBRARY_PATH="/database/asiql60/IQ-16_0/lib64:/database/asiql60/OCS-
15_0/lib:/database/asiql60/OCS-15_0/lib3p64:/database/asiql60/OCS-15_0/lib3p:"
...
declare -x LIB="/database/asiql60/OCS-15_0/lib:"
...
declare -x LOGNAME="sybase_user"
..
declare -x MAIL="/var/spool/mail/sybase_user"
...
declare -x PATH="/database/asiql60/IQ-16_0/bin64:/database/asiql60/COCKPIT-
4/bin:/database/asiql60/OCS-15_0/bin:/usr/lib64/qt-
3.3/bin:/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/sbin:/home/sybase_user/bin"
declare -x PWD="/home/sybase_user"
...
declare -x SAP_JRE7="/database/asiql60/shared/SAPJRE-7_1_015_64BIT" declare -x
SAP_JRE7_32="/database/asiql60/shared/SAPJRE-7_1_015_32BIT" declare -x
SAP_JRE7_64="/database/asiql60/shared/SAPJRE-7_1_015_64BIT" declare -x
SAP_JRE8="/database/asiql60/shared/SAPJRE-8_1_008_64BIT" declare -x
SAP_JRE8_32="/database/asiql60/shared/SAPJRE-8_1_008_32BIT" declare -x
SAP_JRE8_64="/database/asiql60/shared/SAPJRE-8_1_008_64BIT" declare -x SHELL="/bin/bash"
...
declare -x SSH_CLIENT="10.27.84.91 65084 22"
declare -x SSH_CONNECTION="10.27.84.91 65084 10.61.110.151 22"
...
declare -x SYBASE="/database/asiql60"
declare -x SYBASE_JRE7_64="/database/asiql60/shared/SAPJRE-7_1_015_64BIT" declare -x
SYBASE_OCS="OCS-15_0"
declare -x SYBROOT="/database/asiql60"
...
declare -x USER="sybase_user"
```

2. Verify that the SAP servers have started.

```
[sybase_user@s1 NETAPP]$ stop_iq
Checking system ...
The following 2 server(s) are owned by '500'
## Owner PID Started CPU Time Additional Information
-- -----
```

```
1: 500 11328 May13 14:19:11:26 SVR:testen2 DB:testen2 PORT:2639
/database/asiql60/IQ-16_0/bin64/iqsrv16 @testen2.cfg testen2.db -gn 25 -o /database/asiql60/IQ-
16_0/logfiles/testen2.0005.srvlog -hn 5
```

3. Verify that the testen2 database files are created in the /database/testen2 directory.

```
-rw----- 1 sybase_user sybase_user 3985408 May 25 14:08 testen2.db
-rw-r--r-- 1 sybase_user sybase_user 1854061359 May 25 14:11 testen2.iqmsg
-rw-rw-r-- 1 sybase_user sybase_user 517 May 25 14:00 testen2.lmp
-rw----- 1 sybase_user sybase_user 2671050752 May 25 14:11 testen2.log
```

4. Check the LUN associated with the home directory of SAP-Sybase. This directory must not be replicated to another storage or server. The catalog.full file is used to restore the database.

```
[sybase_user@s1]$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/sda3	32G	5.7G	25G	19%	/
tmpfs	48G	0	48G	0%	/dev/shm
/dev/sda1	194M	102M	83M	56%	/boot
/dev/mapper/vg_s1_bin-lv_s1_bin	197G	8.4G	179G	5%	/database

5. Check the content of the interfaces file in /database/asiql60/IQ-16_0. This configuration is important to check the port through which to access the database and IP.

```
testen2
master tcp ether 10.61.110.151 2639
query tcp ether 10.61.110.151 2639
```

6. Log in to the database as sybase_user.

```
[sybase_user@s1 ~]$ isql -Stesten2 -U<user> -P<password>
1>
```

The output must not display errors.

7. Populate the testen2 database to a few terabytes. Check the initial size of each database; the size should be approximately 23.5TB.

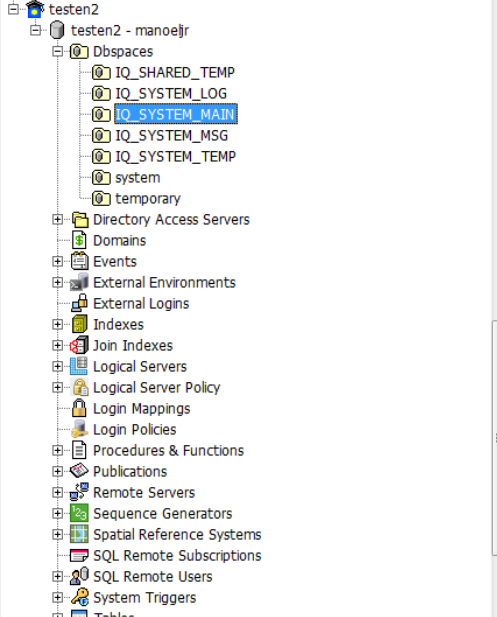
```
[sybase_user@s1 ~]$ isql -Stesten2 -U<user> -P<password>
1> select * from sp_iqstatus() where name like '%Main IQ Block Used%'
2> go
Name                                     Value
-----
Main IQ Blocks Used: 770092156 of 939197518, 81%=23501Gb, Max Block#: 1050129902 (1 row
affected)
1>
```

8. In Sybase Central, check the distribution of raw devices under IO_SYSTEM_MAIN.

Note: Verify that the raw devices and LUNs are presented to S1 server and testen2 database matches.

Note: It is possible to present the RAW devices to the database/server either directly or through symbolic links.

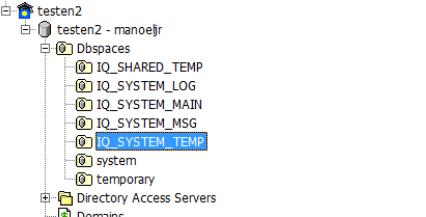
In the following example, /dev/raw/raw6 is presented as a symbolic link for database/testen/DWSIQ.NETAPPMAIN001_DB2.



Name	Path	Mode	Size	Reserve
IQ_SYSTEM_MAIN	DWSIQ.NETAPPMAIN001_DB2	rw	1048586	0
IQ_SYSTEM_MAIN01_DB2	/dev/raw/raw17	rw	1048586	0
IQ_SYSTEM_MAIN02_DB2	/dev/raw/raw18	rw	1048586	0
IQ_SYSTEM_MAIN03_DB2	/dev/raw/raw19	rw	1048586	0
IQ_SYSTEM_MAIN04_DB2	/dev/raw/raw20	rw	1048586	0
IQ_SYSTEM_MAIN05_DB2	/dev/raw/raw21	rw	1048586	0
IQ_SYSTEM_MAIN06_DB2	/dev/raw/raw22	rw	1048586	0
IQ_SYSTEM_MAIN07_DB2	/dev/raw/raw23	rw	1048586	0
IQ_SYSTEM_MAIN08_DB2	/dev/raw/raw24	rw	1048586	0
IQ_SYSTEM_MAIN09_DB2	/dev/raw/raw25	rw	1048586	0
IQ_SYSTEM_MAIN10_DB2	/dev/raw/raw26	rw	1048586	0
IQ_SYSTEM_MAIN11_DB2	/dev/raw/raw27	rw	1048586	0
IQ_SYSTEM_MAIN12_DB2	/dev/raw/raw28	rw	1048586	0
IQ_SYSTEM_MAIN13_DB2	/dev/raw/raw29	rw	1048586	0
IQ_SYSTEM_MAIN14_DB2	/dev/raw/raw30	rw	1048586	0
IQ_SYSTEM_MAIN15_DB2	/dev/raw/raw31	rw	1048586	0
IQ_SYSTEM_MAIN16_DB2	/dev/raw/raw32	rw	1048586	0
IQ_SYSTEM_MAIN17_DB2	/dev/raw/raw34	rw	1048586	0
IQ_SYSTEM_MAIN18_DB2	/dev/raw/raw35	rw	1048586	0
IQ_SYSTEM_MAIN19_DB2	/dev/raw/raw36	rw	1048586	0
IQ_SYSTEM_MAIN20_DB2	/dev/raw/raw37	rw	1048586	0
IQ_SYSTEM_MAIN21_DB2	/dev/raw/raw38	rw	1048586	0
IQ_SYSTEM_MAIN22_DB2	/dev/raw/raw39	rw	1048586	0
IQ_SYSTEM_MAIN23_DB2	/dev/raw/raw40	rw	1048586	0
IQ_SYSTEM_MAIN24_DB2	/dev/raw/raw41	rw	1048586	0
IQ_SYSTEM_MAIN25_DB2	/dev/raw/raw42	rw	1048586	0
IQ_SYSTEM_MAIN26_DB2	/dev/raw/raw43	rw	1048586	0

9. Under IO_SYSTEM_TEMP, check the distribution of RAW devices.

In the following example, /dev/raw/raw7 is presented as a symbolic link for /database/testen/DWSIQ.NETAPPTEMP001_DB2.



Name	Path	Mode	Size	Reserve
IQ_SYSTEM_TEMP	DWSIQ.NETAPPTEMP001_DB2	rw	255997	0
IQ_SYSTEM_TEMP01_DB2	/dev/raw/raw8	rw	255997	0
IQ_SYSTEM_TEMP02_DB2	/dev/raw/raw9	rw	255997	0
IQ_SYSTEM_TEMP03_DB2	/dev/raw/raw10	rw	255997	0
IQ_SYSTEM_TEMP04_DB2	/dev/raw/raw11	rw	255997	0
IQ_SYSTEM_TEMP05_DB2	/dev/raw/raw12	rw	255997	0
IQ_SYSTEM_TEMP06_DB2	/dev/raw/raw14	rw	255997	0
IQ_SYSTEM_TEMP07_DB2	/dev/raw/raw15	rw	255997	0
IQ_SYSTEM_TEMP08_DB2	/dev/raw/raw16	rw	255997	0

10. In the server S1, verify that the raw devices match the LUNs presented to the testen2 database.

```

MAIN (TOTAL = 28TB)
/dev/mapper/b1p1 RAW6
/dev/mapper/b3p1 RAW20
/dev/mapper/b5p1 RAW17
/dev/mapper/b6p1 RAW18
/dev/mapper/b8p1 RAW23
/dev/mapper/b9p1 RAW21
/dev/mapper/b10p1 RAW26
/dev/mapper/b11p1 RAW19
/dev/mapper/b12p1 RAW22
/dev/mapper/b13p1 RAW25
/dev/mapper/b14p1 RAW24
/dev/mapper/b15p1 RAW27
/dev/mapper/b16p1 RAW34
/dev/mapper/b17p1 RAW28
/dev/mapper/b18p1 RAW29
/dev/mapper/b19p1 RAW30
/dev/mapper/b20p1 RAW32
/dev/mapper/b21p1 RAW35
/dev/mapper/b22p1 RAW31
/dev/mapper/b23p1 RAW36
/dev/mapper/b24p1 RAW42
/dev/mapper/b25p1 RAW39
/dev/mapper/b26p1 RAW44

```

```
/dev/mapper/b27p1 RAW38
/dev/mapper/b28p1 RAW37
/dev/mapper/b29p1 RAW41
/dev/mapper/b30p1 RAW43
/dev/mapper/b32p1 RAW40
```

```
TEMP (TOTAL = 2.25TB)
/dev/mapper/s7p1 RAW7
/dev/mapper/s8p1 RAW8
/dev/mapper/s9p1 RAW9
/dev/mapper/s10p1 RAW10
/dev/mapper/s11p1 RAW11
/dev/mapper/s12p1 RAW12
/dev/mapper/s14p1 RAW14
/dev/mapper/s15p1 RAW15
/dev/mapper/s16p1 RAW16
```

11. Check the process security limits on both servers. Comment the following lines. This change enables avoiding resource bottlenecks in servers when connecting to the database.

```
[root@s1 ~]# vi /etc/security/limits.d/90-nproc.conf
7.
#* soft nproc 1024
root soft nproc unlimited
[root@s1 ~]# reboot

[root@s2 ~]# vi /etc/security/limits.d/90-nproc.conf
#* soft nproc 1024
root soft nproc unlimited
[root@s2 ~]# reboot
```

5 Database Backup

5.1 Snapshot Copy–Based Backup of SAP IQ Database

Test Objectives

- Demonstrate the success of the full backup of SAP IQ database testen2.
- Check the environment variables and accessibility of the SAP IQ database.

Test Procedure

1. Verify if there are Snapshot copies is created on storage1 (FAS8060HA). At this moment, there are no Snapshot copies. There are only SnapMirror® relationships.

```
superman::> volume snapshot show

Vserver   Volume ---Blocks--- Snapshot Size Total% Used% -----
-----
db         db_root

weekly.2016-05-22_0015 268KB    0% 40%
daily.2016-05-29_0010 164KB    0% 29%
weekly.2016-05-29_0015 268KB    0% 40%
daily.2016-05-30_0010 256KB    0% 39%
hourly.2016-05-30_1105 176KB    0% 30%
...

s1
snapmirror.beeccb17-11fa-11e6-9e98-00a09813fe5a_2154847965.2016-05-05_100320 112KB 0% 0%
s10
snapmirror.beeccb17-11fa-11e6-9e98-00a09813fe5a_2154847974.2016-05-05_101151 112KB 0% 0%
s11
snapmirror.beeccb17-11fa-11e6-9e98-00a09813fe5a_2154847975.2016-05-05_101536 112KB 0% 0%
s12
snapmirror.beeccb17-11fa-11e6-9e98-00a09813fe5a_2154847976.2016-05-05_101614 112KB 0% 0%
```

```

s13
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847977.2016-05-05_101642 96KB 0% 29%
s14
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847978.2016-05-05_101712 112KB 0% 0%
s15
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847979.2016-05-05_101740 112KB 0% 0%
s16
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847980.2016-05-05_101931 112KB 0% 0%
s2
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847966.2016-05-05_100406 96KB 0% 29%
s3
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847967.2016-05-05_100457 112KB 0% 0%
s4
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847968.2016-05-05_100525 112KB 0% 0%
s5
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847969.2016-05-05_100625 112KB 0% 0%
s6
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847970.2016-05-05_100708 112KB 0% 0%
s7
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847971.2016-05-05_100822 112KB 0% 0%
s8
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847972.2016-05-05_100855 112KB 0% 0%
s9
snapmirror.beecbb17-11fa-11e6-9e98-00a09813fe5a_2154847973.2016-05-05_100956 112KB 0% 0%

n1  vol0
    nightly.1      445.9MB      0%      3%
    hourly.5       68.51MB     0%      1%
    hourly.4       1.21GB      0%      8%
    hourly.3       117.9MB     0%      1%
    nightly.0       157.7MB     0%      1%
    hourly.2       116.7MB     0%      1%
    hourly.1       65.93MB     0%      0%
    hourly.0       11.66MB     0%      0%
n2  vol0
    nightly.1      383.8MB      0%      3%
    hourly.5       296.0MB     0%      3%
    hourly.4       96.75MB     0%      1%
    hourly.3       54.78MB     0%      0%
    nightly.0       278.6MB     0%      2%
    hourly.2       160.9MB     0%      1%
    hourly.1       340.6MB     0%      3%
    hourly.0       8.89MB      0%      0%
42 entries were displayed.

```

2. Check the database size.

Note that the output of the commands `sp_iqstatus` and `sp_iqdbsize` yields the same results.

```

[sybase_user@s1 testen]$ isql -S<user> -UDBA -P<password>

1> select * from sp_iqstatus() where name like '%Main IQ Block%' 2> go
Name
Value
----- Main IQ Blocks Used:
770091564 of 939197518, 81%=23501Gb, Max Block#: 1050129902

sybase_user@s1 testen]$ isql -S<user> -UDBA -P<password>
1> sp_iqdbsize
2> go
Database
PhysicalBlocks      770091564
Kbytes              18629918208
Pages               90949971
CompressedPages     90775386
NBlocks             582184944
CatalogBlocks       7536
RLVLogBlocks        0
RLVLogKBytes        0
-----

```

```
/database/testen/testen2.db
(1 row affected)
(return status = 0)
```

3. Run the following commands to create a consistency group Snapshot copy of testen2's LUNs:

```
[sybase_user@s1] isql -Stesten2 -U<user> -P<password>

1> begin
2> backup database full virtual encapsulated
3>'cd /home/sybase_user/neRO;/home/sybase_user/neRO/neRO --action cg -- conf
/home/sybase_user/neRO/conf/base2.conf --vserver db --snap snap1' 4> to
'/database/snapt2/snap1.full'
5> end
6> go
Snapshot created.
1>

[sybase_user@s1 neRO]$ cd /database/snapt2/
[sybase_user@s1 snapt2]$ ls
snap1.full.1

[sybase_user@s1 snapt2]$ ls -la
total 291976
drwxrwxr-x 2 sybase_user sybase_user 4096 May 30 17:06 .
drwxr-xr-x 6 sybase_user sybase_user 4096 May 26 15:22 ..
-rw-rw-r-- 1 sybase_user sybase_user 298973476 May 30 16:43 snap1.full.1
```

Note: We are using the SAP back up database full virtual encapsulated technique and encapsulating the Snap Creator Framework commands to create a consistency group of Snapshot copies from all involved volumes.

For the system-level backup of table data to be consistent with the virtual backup without additional steps, the system-level backup must be created by running the backup command and by the backup transaction.

The parameter `VIRTUAL ENCAPSULATED` shell command allows arbitrary shell commands to be executed as part of the backup operation to guarantee these semantics. The user must guarantee that the shell commands correctly perform the system-level backup.

The content of the `base2.conf` file is:

```
<ip_cluster>      relaxed      b1
<ip_cluster>      relaxed      b2
<ip_cluster>      relaxed      ...
<ip_cluster>      relaxed      s1
<ip_cluster>      relaxed      ...
```

4. After the process ends, copy the catalog file to the S2 server. In the S2 server, we use the catalog file to restore the testen2 database.

```
[sybase_user@s1 neRO]$ cd /database/snapt2/
[sybase_user@s1 neRO]$ scp snap1.full.1 sybase_user@10.61.110.152:/database/snapt2/

[sybase_user@s2 ~]$ cd /database/snapt2/ [sybase_user@s2 snapt2]$ ls
snap1.full.1
```

5. After the process ends, check the time and occupied size of the Snapshot copy.

```
Time =~15s
[sybase_user@s1 NETAPP]$ cat snap_testen2_log getdate() -----
      May 30 2016  4:43:28.912726PM
(1 row affected)
Snapshot created.
getdate() -----
      May 30 2016  4:43:43.771401PM
```

Note that the size of the first Snapshot copy is approximately 32MB from each LUN (1TB) and 112KB from each small LUN (250GB).

```
superman::> volume snapshot show
```

```
--Blocks---
Vserver Volume Snapshot Size Total% Used%
-----
db          b1          snap1      32.69MB    0%    0%
            b10         snap1      32.05MB    0%    0%
            b11         snap1      32.03MB    0%    0%
            b12         snap1      32.10MB    0%    0%
            b13         snap1      32.05MB    0%    0%
            ...
            b21         snap1      32.11MB    0%    0%
            b22         snap1      32.05MB    0%    0%
            b23         snap1      32.05MB    0%    0%
            ...
            b29         snap1      32.07MB    0%    0%
            b3          snap1      32.02MB    0%    0%
            b30         snap1      32.05MB    0%    0%
            b32         snap1      32.05MB    0%    0%
            ...
            s1          snap1      112KB      0%    0%
            ...
```

- After the process completes, check the log files for the time it took to create the Snapshot copy.
In the following example output, the Snapshot copy start and end times are displayed as 16:43:28 and 16:43:43.

```
[sybase_user@s1~]$ tail -150 /database/testen/testen2.iqmsg
I. 05/30 16:43:28. 0000768690 Txn 21156119 0 21156119 I. 05/30 16:43:28. 0000768690 Cmt 21156120
I. 05/30 16:43:28. 0000768690 PostCmt 0
I. 05/30 16:43:28. 0000768690 Txn 21156121 0 21156121 I. 05/30 16:43:28. 0000768690
ValidateBackup
I. 05/30 16:43:28. 0000768690 ValidateBackup
I. 05/30 16:43:28. 0000768690 Chk
I. 05/30 16:43:29. 0000768690 ChkDone [NumTxnCP: 3]
I. 05/30 16:43:29. 0000768690 PrepareForBackup
I. 05/30 16:43:29. 0000768690 PostChk
I. 05/30 16:43:29. 0000768690 DoBackup
I. 05/30 16:43:29. 0000768690 [20951]: Total number of IQ blocks to be backed up: 8284 I. 05/30
16:43:29. 0000768690 TeamBase: db IQbackup::team W 224; G 144
I. 05/30 16:43:33. 0000768690 [20949]: Number of IQ blocks backed up: 99
I. 05/30 16:43:33. 0000768690 [20950]: Total number of IQ blocks backed Up: 8284 at 2016-05-30
16:43:33
      IQ Blocksize 32768
      backup Blocksize 32808
      blocking factor 25
      buffer size 820216
I. 05/30 16:43:43. 0000768690 PostBackup
I. 05/30 16:43:43. 0000768690 Cmt 21156138
I. 05/30 16:43:43. 0000768690 PostCmt 0
I. 05/30 16:43:43. 0000768690 Chk
I. 05/30 16:43:43. 0000768690 ChkDone [NumTxnCP: 2] I. 05/30 16:43:43. 0000768690 PostChk

[sybase_user@s1 NETAPP]$ cat /database/asiq160/IQ-16_0/logfiles/backup.syb
BACKUP, -847292463.0, testen2.db, ASIQ, '2016-05-30 20:43:29.668', <user>, Full, Arch,
/database/snapt2/snap1.full, ''
```

- Start an initial instance in the S2 server.

```
[sybase_user@s2 testen]$ ls
INSTANCIA_DUMMY.cfg s2_eng_dumy_log s2_eng_dumy_log.stderr
[sybase_user@s2 testen]$ start_iq @INSTANCIA_DUMMY.cfg
Starting server s2_eng_dumy on s2 at port 2640 (05/30 17:37:55)
Run Directory
Server Executable
Server Output Log
Server Version
16.0.0/Linux 2.6.18-194.el5
Open Client Version : N/A
User Parameters : '@INSTANCIA_DUMMY.cfg' Default Parameters : -gn 25
```

```

I. 05/30 17:38:02. SAP IQ
I. 05/30 17:38:02. Version 16.0
I. 05/30 17:38:02. (64bit mode)
: /database/testen
: /database/asiql60/IQ-16_0/bin64/iqsrv16 : s2_eng_dumy_log
: 16.0.110.2014/sp11
I. 05/30 17:38:02. Copyright 1992-2015 by SAP AG or an SAP affiliate company. All rights reserved
I. 05/30 17:38:02. Copyright (c) 2015 SAP SE or an SAP affiliate company.
I. 05/30 17:38:02.
I. 05/30 17:38:02.
I. 05/30 17:38:02.
I. 05/30 17:38:02.
I. 05/30 17:38:02.
I. 05/30 17:38:02.
All rights reserved.
Use of this software is governed by the Sybase License Agreement. Refer to
http://www.sybase.com/softwarelicenses.
Processors detected: 2 (containing 24 logical processors)
Maximum number of processors the server will use: 2 physical
processor(s), 12 core(s)
I. 05/30 17:38:02. Running Linux 2.6.32-504.el6.x86_64 #1 SMP Tue Sep 16 01:56:35 EDT 2014
on X86_64
I. 05/30 17:38:02. Server built for X86_64 processor architecture
I. 05/30 17:38:02. 524288K of memory used for caching
I. 05/30 17:38:02. Minimum cache size: 524288K, maximum cache size: 524288K I. 05/30 17:38:02.
Using a maximum page size of 4096 bytes
I. 05/30 17:38:02. Multiprogramming level: 25
I. 05/30 17:38:02. Warning: -gn value of 25 is too low for -gm value of 20
I. 05/30 17:38:02. Automatic tuning of multiprogramming level
===== IQ server starting with:
is disabled
17:38
20 connections
5 cmd resources
( -gm )
( -iqgovern )
( -iqmt )
1000 threads
100 Kb thread stack size ( -iqtss )
100000 Kb thread memory size ( -iqmt * -iqtss ) 24 IQ number of cpus ( -iqnumbercpus )
0 MB maximum size of IQMSG file ( -iqmsgsz )
0 copies of IQMSG file archives ( -iqmsgnum )
=====
I. 05/30 17:38:02.
I. 05/30 17:38:02.
I. 05/30 17:38:02.
I. 05/30 17:38:02.
I. 05/30 17:38:02.
I. 05/30 17:38:07.
I. 05/30 17:38:07.
New process id is 3226
Server started successfully
Database server started at Mon May 30 2016 Trying to start SharedMemory link ...
SharedMemory link started successfully Trying to start TCPIP link ...
Starting on port 2640
TCPIP link started successfully
Now accepting requests

[sybase_user@s2 ~]$ stop_iq
Checking system ...
The following 1 server(s) are owned by '500'
## Owner PID Started CPU Time Additional Information
-----
1: 500 3226 May30
00:00:23 SVR:s2_eng_dumy DB:none PORT:2640
/database/asiql60/IQ-16_0/bin64/iqsrv16 @INSTANCIA_DUMMY.cfg -gn 25 -hn 5

```

8. Create a restore file.

```
[sybase_user@s2 testen]$ cat restore_vbe_testen2.sh
dbisqlc -c "uid=Xjr;pwd=netapp-
123;eng=s2_eng_dumy;dbn=utility_db;links=tcip{host=10.61.110.152;port=2641}" -q
restore_vbe_testen2.sql
```

9. Create clones from the snap1 Snapshot copy.

```
[sybase_user@s1 neRO]$ ./neRO --action clone-create -conf /home/sybase_user/neRO/conf/base2.conf
--vserver db --snap snap1 --clone-name cloneX

neRO - NetApp Storage Management Tool Version 1.8
Action: clone-create
Creating FlexClone on NetApp: 10.61.110.100 vservers: db - volume: cloneX__snap1__b1 - snapshot:
snap1
<results status="passed"></results>
Creating FlexClone on NetApp: 10.61.110.100 vservers: db - volume: cloneX__snap1__b5 - snapshot:
snap1
<results status="passed"></results>
...
Creating FlexClone on NetApp: 10.61.110.100 vservers: db - volume: cloneX__snap1__b13 - snapshot:
snap1
<results status="passed"></results>
Creating FlexClone on NetApp: 10.61.110.100 vservers: db - volume: cloneX__snap1__b10 - snapshot:
snap1
<results status="passed"></results>
Creating FlexClone on NetApp: 10.61.110.100 vservers: db - volume: cloneX__snap1__b15 - snapshot:
snap1
<results status="passed"></results> ...
Creating FlexClone on NetApp: 10.61.110.100 vservers: db - volume: cloneX__snap1__b26 - snapshot:
snap1
<results status="passed"></results>
...
Creating FlexClone on NetApp: 10.61.110.100 vservers: db - volume: cloneX__snap1__s10 - snapshot:
snap1
<results status="passed"></results>
Creating FlexClone on NetApp: 10.61.110.100 vservers: db - volume: cloneX__snap1__s11 - snapshot:
snap1
<results status="passed"></results>
...
Creating FlexClone on NetApp: 10.61.110.100 vservers: db - volume: cloneX__snap1__s16 - snapshot:
snap1
<results status="passed"></results>
```

10. Mount all LUNs in server S2 based on the snap1 Snapshot copy, assigning new IDs.

In the following example, the LUNs are mounted and assigned IDs as #300.

```
[sybase_user@s1 neRO]$ ./neRO --action lun-mount --conf /home/sybase_user/neRO/conf/base2.conf --
vserver db --snap snap1 --clone-name cloneX --igroup s2 --lun-id 300

neRO - NetApp Storage Management Tool Version 1.8
Action: lun-mount
LUN-ID Sequence starting with: 300
Mapping LUN on NetApp: 10.61.110.100 vservers: db - LUN: /vol/cloneX__snap1__b1/b1 - igroup: s2 -
LUN ID: 300
<results status="passed">
<lun-id-assigned>300</lun-id-assigned> </results>
Mapping LUN on NetApp: 10.61.110.100 vservers: db - LUN: /vol/cloneX__snap1__b5/b5 - igroup: s2 -
LUN ID: 301
<results status="passed"> <lun-id-assigned>301</lun-id-assigned>
</results>
Mapping LUN on NetApp: 10.61.110.100 vservers: db - LUN: /vol/cloneX__snap1__b6/b6 - igroup: s2 -
LUN ID: 302
<results status="passed"> <lun-id-assigned>302</lun-id-assigned>
</results>
...
</results>
Mapping LUN on NetApp: 10.61.110.100 vservers: db - LUN: /vol/cloneX__snap1__b8/b8 - igroup: s2 -
LUN ID: 307
<results status="passed">
```

11. Run the following commands on the server S2 to recognize all the LUNs:

12. Create a new `multipath.conf` file with the new LUN IDs on the S2 server.

18

```
[root@s2 ~]# for (( i = 10; i <= 32 ; i=i+1 )); do sanlun lun show -p | grep 'ONTAP\|3600'| grep
-A1 -w cloneX__snap1__s$i | sed 's/db:\vol\cloneX__snap1__s[0-9][0-9]\\//g' | awk -F":" '{
printf " \"$2\" \" }' | awk '{ print "multipath \" \"{\\n\" \"\\t wwid \" $2 \"\\n\" \"\\t alias \" $1 \"_cloneX\"
\"\\n\"}'; done
```

13. Restart the multipathd service.

```
[root@s2 ~]# service multipathd restart
[root@s2 ~]# chkconfig multipathd on
```

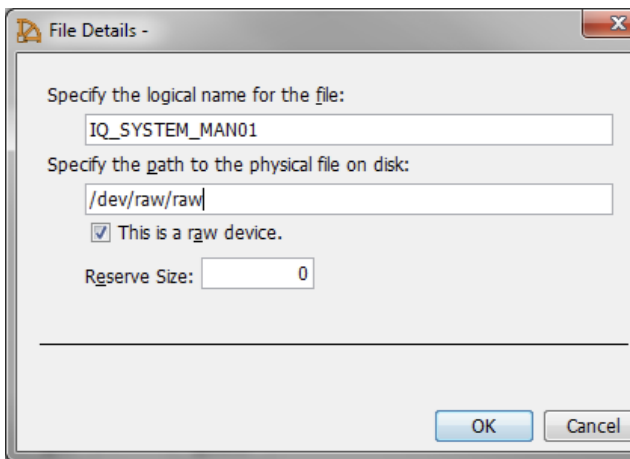
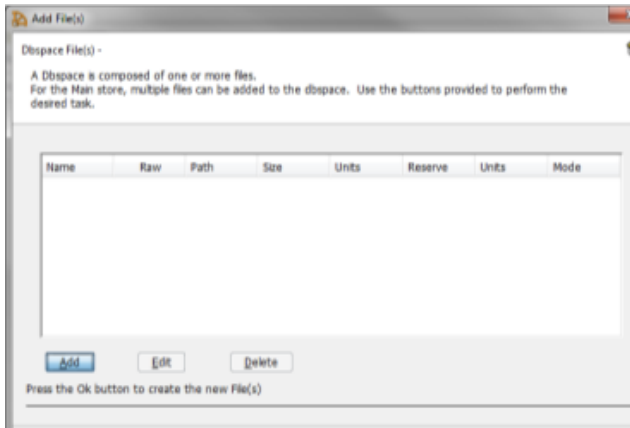
14. Create the new raw devices in the server S2, based on the new LUNs.

```
[root@s2 ~]#
/bin/raw /dev/raw/raw6 /dev/mapper/b1_cloneXp1
/bin/raw /dev/raw/raw17 /dev/mapper/b5_cloneXp1
/bin/raw /dev/raw/raw18 /dev/mapper/b6_cloneXp1
/bin/raw /dev/raw/raw19 /dev/mapper/b11_cloneXp1
/bin/raw /dev/raw/raw20 /dev/mapper/b3_cloneXp1
/bin/raw /dev/raw/raw21 /dev/mapper/b9_cloneXp1
/bin/raw /dev/raw/raw22 /dev/mapper/b12_cloneXp1
/bin/raw /dev/raw/raw23 /dev/mapper/b8_cloneXp1
/bin/raw /dev/raw/raw24 /dev/mapper/b14_cloneXp1
/bin/raw /dev/raw/raw25 /dev/mapper/b13_cloneXp1
...
/bin/raw /dev/raw/raw8 /dev/mapper/s8_cloneXp1
/bin/raw /dev/raw/raw9 /dev/mapper/s9_cloneXp1
/bin/raw /dev/raw/raw10 /dev/mapper/s10_cloneXp1
/bin/raw /dev/raw/raw11 /dev/mapper/s11_cloneXp1
/bin/raw /dev/raw/raw12 /dev/mapper/s12_cloneXp1
/bin/raw /dev/raw/raw14 /dev/mapper/s14_cloneXp1
/bin/raw /dev/raw/raw15 /dev/mapper/s15_cloneXp1
/bin/raw /dev/raw/raw16 /dev/mapper/s16_cloneXp1

[root@s2 ~]# chown sybase_user:sybase_user /dev/raw/raw*
[root@s2 ~]# chmod 660 /dev/raw/raw*

[root@s2 ~]# raw -qa
/dev/raw/raw6: bound to major 253, minor 7
/dev/raw/raw7: bound to major 253, minor 59
/dev/raw/raw8: bound to major 253, minor 5
/dev/raw/raw9: bound to major 253, minor 5
/dev/raw/raw10: bound to major 253, minor 20
/dev/raw/raw11: bound to major 253, minor 10
...
/dev/raw/raw35: bound to major 253, minor 48
/dev/raw/raw36: bound to major 253, minor 75
/dev/raw/raw37: bound to major 253, minor 68
/dev/raw/raw38: bound to major 253, minor 41
/dev/raw/raw39: bound to major 253, minor 61
/dev/raw/raw40: bound to major 253, minor 74
/dev/raw/raw41: bound to major 253, minor 43
/dev/raw/raw42: bound to major 253, minor 71
/dev/raw/raw43: bound to major 253, minor 65
/dev/raw/raw44: bound to major 253, minor 64
```

15. Add or associate the raw devices to TEMP or MAIN in the SAP IQ database, in this case, testen2.



16. Execute the restore.

```
[sybase_user@s2 testen]$ cat restore_vbe_testen2.sh
dbisqlc -c
"uid=DBA;pwd=sql;eng=s2_eng_dumy;dbn=utility_db;links=tcipip{host=10.61.110.152;port=2640}" -q
restore_vbe_testen2.sql

[sybase_user@s2 testen]$ cat restore_vbe_testen2.sql
RESTORE DATABASE '/database/testen/testen2.db' FROM '/database/testen/snap1.full'
go
```

17. Verify that the files created in server S2 are the same size as in S1 server: 8.db, *.iqmsg, and *.lmp.

```
[sybase_user@s2 testen]$ ./restore_vbe_testen2.s

[sybase_user@s2 testen]$ ls -la
total 295584
drwxrwxr-x 2 sybase_user sybase_user          4096 May 31 17:36 .
drwxr-xr-x 5 sybase_user sybase_user          4096 May 30 16:11 ..
lrwxrwxrwx 1 sybase_user sybase_user /dev/raw/raw6      13 May 31 15:30
DWSIQ.NETAPPMAN001_DB2 ->
lrwxrwxrwx 1 sybase_user sybase_user /dev/raw/raw7      13 May 31 17:36
DWSIQ.NETAPPTMP001_DB2 ->
-rw-rw-r-- 1 sybase_user sybase_user          223 May 30 17:12
INSTANCIA_DUMMY.cfg
-rwxrwxrwx 1 sybase_user sybase_user          129 May 31 15:14
restore_vbe_testen2.sh
-rwxrwxrwx 1 sybase_user sybase_user           85 May 31 15:26
restore_vbe_testen2.sql
-rw-rw-r-- 1 sybase_user sybase_user        10844 May 31 17:36
s2_eng_dumy_log
```

-rw-rw-r-- 1 sybase_user sybase_user	5814 May 31 17:36
s2_eng_dumy_log.stderr	
-rwxrwxrwx 1 sybase_user sybase_user	298973476 May 31 15:21
snapl.full.1	
-r--r--r-- 1 sybase_user sybase_user	3649536 May 31 17:36
testen2.db	
-rw-r--r-- 1 sybase_user sybase_user	5336 May 31 17:36
testen2.iqmsg	
-rw-rw-r-- 1 sybase_user sybase_user	533 May 31 17:36
testen2.lmp	

18. In the output, verify the ChkDone process.

```
[sybase_user@s2 testen]$ cat testen2.iqmsg

I. 05/31 17:36:24. 0000000000 OpenDatabase Completed
I. 05/31 17:36:24. 0000000000 IQ cmd line srv opts: -iqmc 20000 -iqtc 20000 -iqmt 1000 - iggovern
5
I. 05/31 17:36:24. 0000000000 IQ full cmd line: -n s2_eng_dumy -c 512m -gc 120 -gd DBA -gk DBA -
gl ALL -gm 20 -gp 4096 -gr 6000 -ti 60 -tl 300 -iqmc 20000 -iqtc 20000 -iqmt 1000 - iggovern 5 -
iqtss 100 -p 4096 -zo createdb.audit -x tcpip{port=2640} -o s2_eng_dumy_log - gn 25 -hn 5
I. 05/31 17:36:24. 0000000000 DB: r/w, Main 3596MB Buffs=39960, Temp 3596MB Buffs=39960,
Pgsz=524288/32768blks/16bpc, BP=4
I. 05/31 17:36:24. 0000000000 DB: Frmt#: 23F/2T/1P (FF: 03/18/1999)
I. 05/31 17:36:24. 0000000000 DB: Versn: 16.0.110.2014/10214/P/sp11/Enterprise Linux64 - x86_64 -
2.6.18-194.el5/64bit/2015-11-21 01:29:07
I. 05/31 17:36:24. 0000000000
I. 05/31 17:36:24. 0000000000
I. 05/31 17:36:24. 0000000000
I. 05/31 17:36:24. 0000000000
sybase_user
I. 05/31 17:36:24. 0000000000
I. 05/31 17:36:24. 0000000000
DB: Name: /database/testen/testen2.db
DB: Txn ID Seq: 21156124
DB: DBID Blk: 6711026
DB: IQ Server s2_eng_dumy Override Single, PID 19245, LOGIN
DB: Database encryption is OFF.
Mem: 40645mb/M40645
Main Blks: U770091564/81%, Buffers: U506/L0 Temporary Blks: U617/0%, Buffers: U38/L0
Main I: L9381/P253 O: C253/D253/P0 D:0 C:100.0 Temporary I: L682/P0 O: C38/D38/P0 D:0 C: 0.0
I. 05/31 17:36:24. 0000000000 Collation ISO_BINENG, Case Ignore, Blank Padding On, Comparisons
are Conditioned
I. 05/31 17:36:24. 0000000000 Result Cache Max Pages: 0
I. 05/31 17:36:24. 0000000000 RcvyCmpl
I. 05/31 17:36:24. 0000000000 Chk
I. 05/31 17:36:24. 0000000000 ChkDone [NumTxnCP: 0]
I. 05/31 17:36:24. 0000000000 PostChk
I. 05/31 17:36:25. 0000000000 Result Cache Max Pages: 0
I. 05/31 17:36:25. 0000000000 CloseDatabase
I. 05/31 17:36:26. 0000000000 Using licenses from: /usr/local/flexlm/licenses/license.dat
I. 05/31 17:36:26. 0000000000 Checked out graced license for 1 IQ_CORE (2015.1121) will expire
Thu 09 Jun 2016 01:00:00 AM EDT.
I. 05/31 17:36:26. 0000000000 Failed to obtain license(s) for IQ_CORE feature from license
file(s) or server(s).
I. 05/31 17:36:26. 0000000000 Cannot find license file.
I. 05/31 17:36:26. 0000000000 The license files (or license server system network addresses)
attempted are listed below.
I. 05/31 17:36:26. 0000000000 I. 05/31 17:36:26. 0000000000 2_0/licenses:/database/testen I.
05/31 17:36:26. 0000000000 2_0/licenses:/database/testen
I. 05/31 17:36:26.
file or directory"
I. 05/31 17:36:26.
Management website
License feature name: IQ_CORE
License filename: /database/asiql60/SYSAM-
License search path: /database/asiql60/SYSAM-
FLEXnet Licensing error:-1,359. System Error: 2 "No such
For further information, refer to the Sybase Software Asset at http://www.sybase.com/sysam
I. 05/31 17:36:26. 0000000000
I. 05/31 17:36:26. 0000000000
```

```

I. 05/31 17:36:26. 0000000000
I. 05/31 17:36:26. 0000000000
sybase_user
I. 05/31 17:36:26. 0000000000
I. 05/31 17:36:26. 0000000000
DB: Name: /database/testen/testen2.db
DB: Txn ID Seq: 21156125
DB: DBID Blk: 6711010
DB: IQ Server s2_eng_dumy Override Single, PID 19245, LOGIN
DB: Database encryption is OFF.
Mem: 40645mb/M40645
0000000000
0000000000
I. 05/31 17:36:26. 0000000000 WARNING: IQ functionality that requires the IQ_CORE license will be
disabled on Thu 09 Jun 2016 01:00:00 AM EDT, unless a suitable IQ_CORE license is obtained before
that date.
I. 05/31 17:36:26. 0000000000 This product is licensed to:
I. 05/31 17:36:26. 0000000000 OpenDatabase Completed
I. 05/31 17:36:26. 0000000000 IQ cmd line srv opts: -iqmc 20000 -iqtc 20000 -iqmt 1000 - iggovern
5
I. 05/31 17:36:26. 0000000000 IQ full cmd line: -n s2_eng_dumy -c 512m -gc 120 -gd DBA -gk DBA -
gl ALL -gm 20 -gp 4096 -gr 6000 -ti 60 -tl 300 -iqmc 20000 -iqtc 20000 -iqmt 1000 - iggovern 5 -
igtss 100 -p 4096 -zo createdb.audit -x tcpip{port=2640} -o s2_eng_dumy_log - gn 25 -hn 5
I. 05/31 17:36:26. 0000000000 DB: r/w, Main 3596MB Buffs=39960, Temp 3596MB Buffs=39960,
Pgpsz=524288/32768blks/16bpc, BP=4
I. 05/31 17:36:26. 0000000000 DB: Frmt#: 23F/2T/1P (FF: 03/18/1999)
I. 05/31 17:36:26. 0000000000 DB: Versn: 16.0.110.2014/10214/P/sp11/Enterprise Linux64 - x86_64 -
2.6.18-194.el5/64bit/2015-11-21 01:29:07
Main Blks: U770091564/81%, Buffers: U506/L0
Temporary Blks: U617/0%, Buffers: U38/L0
Main I: L1516/P506 O: C0/D252/P0 D:0 C:100.0 Temporary I: L682/P0 O: C38/D38/P0 D:0 C: 0.0
I. 05/31 17:36:26. 0000000003 140496505366272 Warning: Database BIO is synchronous I/O enabled
for RLV.
I. 05/31 17:36:26. 0000000003 140496505366272 Warning: Log buffer manager using synchronous I/O
for RLV.
I. 05/31 17:36:26. 0000000003 140496505366272 RLV Flusher Enabled. I. 05/31 17:36:26. 0000000003
140496505366272 RLV Logging Enabled.
I. 05/31 17:36:26. 0000000003 140496505366272 Bypassing RLV recovery because an RLV dbspace could
not be found.
I. 05/31 17:36:26. 0000000003 Chk
I. 05/31 17:36:27. 0000000003 ChkDone [NumTxnCP: 0]
I. 05/31 17:36:27. 0000000003 PostChk
I. 05/31 17:36:27. 0000000005 PostRestore
I. 05/31 17:36:27. 0000000007 Chk
I. 05/31 17:36:27. 0000000007 ChkDone [NumTxnCP: 0]
I. 05/31 17:36:27. 0000000007 PostChk
I. 05/31 17:36:27. 0000000000 CloseDatabase

```

19. Check the log files to complete the restore process.

```

[sybase_user@s2 logfiles]$ ls -la
total 48
drwxrwxrwx 2 sybase_user sybase_user
drwxrwxr-x 17 sybase_user sybase_user
-rw-rw-r-- 1 sybase_user sybase_user
-rw-rw-r-- 1 sybase_user sybase_user
-rw-rw-r-- 1 sybase_user sybase_user
-rw-rw-r-- 1 sybase_user sybase_user
-rw-rw-r-- 1 sybase_user sybase_user
-rw-rw-r-- 1 sybase_user sybase_user
-rw-rw-r-- 1 sybase_user sybase_user
-rw-rw-r-- 1 sybase_user sybase_user
-rw-rw-r-- 1 sybase_user sybase_user 123 May 31 17:36 backup.syb -rw-rw-rw- 1 sybase_user
sybase_user 39 May 31 16:20 .start_iq.tmp -rw-rw-r-- 1 sybase_user sybase user 1683 May 31 16:15
testen2.0001.srvlog -rw-rw-r-- 1 sybase_user sybase_user 1023 May 31 16:15 testen2.0001.stderr
[sybase_user@s2 logfiles]$ cat backup.syb
RESTORE, -1080687967.0, testen2.db, ASI, '2016-05-30 20:43:29.668', <user>, Full, Arch,
/database/snap2/snap1.full, ''

```

20. Check the configuration file of the database.

The following configuration starts up the new database from the clone.

Configure carefully because, for example, if you use minimum values for IQMC and IQTC, you might force the creation of swap space on the server, resulting in the INSERT and SELECT processes slowing down.

```
[sybase_user@s2 testen]$ cat testen2.cfg
-n testen2s
-c 512m
-gc 120
-gd DBA
-gk DBA
-gl ALL
-gm 20
-gp 4096
-gr 6000
-ti 60
-tl 300
-iqmc 20000 ##TO LIMIT THE MEMORY USED FOR SYBASE IQ = TAKE CARE#
-iqtc 20000 ##TO LIMIT THE MEMORY USED FOR SYBASE IQ = TAKE CARE#
-iqmt 1000
-iqgovern 5
-iqtss 100000
-p 4096
-zo testen2.audit
-n testen2
-x tcpip{port=2640}
```

21. In the S2 server, start the new database (testen2s) that has the same data as the testen2 database.

```
[sybase_user@s2 testen]$ start_iq @testen2.cfg testen2.db

Starting server testen2s on s2 at port 2640 (05/31 17:48:43)
Run Directory : /database/testen
Server Executable : /database/asiql60/IQ-16_0/bin64/iqsrv16
Server Output Log : /database/asiql60/IQ-16_0/logfiles/testen2s.0001.srvlog
Server Version : 16.0.110.2014/sp11
16.0.0/Linux 2.6.18-194.el5
Open Client Version : N/A
User Parameters : '@testen2.cfg' 'testen2.db'
Default Parameters : -gn 25
I. 05/31 17:48:49. SAP IQ
I. 05/31 17:48:49. Version 16.0
I. 05/31 17:48:49. (64bit mode)
Copyright 1992-2015 by SAP AG or an SAP affiliate company. All rights reserved
Copyright (c) 2015 SAP SE or an SAP affiliate company. All rights reserved.
Use of this software is governed by the Sybase License Agreement. Refer to
http://www.sybase.com/softwarelicenses.
Processors detected: 2 (containing 24 logical processors)
I. 05/31 17:48:49. Maximum number of processors the server will use: 2 physical
processor(s), 12 core(s)
I. 05/31 17:48:49. Running Linux 2.6.32-504.el6.x86_64 #1 SMP Tue Sep 16 01:56:35 EDT 2014 on
X86_64
I. 05/31 17:48:49. Server built for X86_64 processor architecture
I. 05/31 17:48:49. 524288K of memory used for caching
I. 05/31 17:48:49. Minimum cache size: 524288K, maximum cache size: 524288K I. 05/31 17:48:49.
Using a maximum page size of 4096 bytes
I. 05/31 17:48:49. Multiprogramming level: 25
I. 05/31 17:48:49. Warning: -gn value of 25 is too low for -gm value of 20 I. 05/31 17:48:49.
Automatic tuning of multiprogramming level is disabled
=====
IQ server starting with:
    20 connections.      (      -gm )
    5 cmd resources     ( -iqgovern )
    1000 threads.       (      -iqmt )
100000 Kb thread stack size ( -iqtss )
100000000 Kb thread memory size ( -iqmt * -iqtss )
24 IQ number of cpus ( -iqnumbercpus )
0 MB maximum size of IQMSG file ( -iqmsgsz )
0 copies of IQMSG file archives ( -iqmsgnum )
=====
I. 05/31 17:48:49. Starting database "testen2" (/database/testen/testen2.db) at Tue May 31 2016
17:48
```

```

I. 05/31 17:48:50. Transaction log: testen2.log
I. 05/31 17:48:50. Starting checkpoint of "testen2" (testen2.db) at Tue May 31 2016 17:48 I.
05/31 17:48:51. Finished checkpoint of "testen2" (testen2.db) at Tue May 31 2016 17:48
Update previous log GUID to: 0a40528c-2736-11e6-8000-e6a2f05d486a GUID to :696130e2-2736-11e6-
8000-afbcbf58c1ace
Database "testen2" (testen2.db) started at Tue May 31 2016 17:48
IQ Server testen2s.
Starting checkpoint of "testen2" (testen2.db) at Tue May 31 2016 17:48
Finished checkpoint of "testen2" (testen2.db) at Tue May 31 2016 17:48
Database server started at Tue May 31 2016 17:48
Trying to start SharedMemory link ...
SharedMemory link started successfully Trying to start TCPIP link ...
Starting on port 2640
    TCPIP link started successfully
Now accepting requests
New process id is 11154
Server started successfully

```

22. Run the following command to check the database status:

```

[sybase_user@s2 testen]$ stop_iq
Checking system ...
The following 1 server(s) are owned by '500'
## Owner      PID      Started  CPUTime  Additional Information
-----
1:    500      11154    17:48   00:00:02   SVR:testen2s DB:testen2 PORT:2640
              /database/asiql60/IQ-16_0/bin64/iqsrv16 @testen2.cfg testen2.db -gn 25 -o
              /database/asiql60/IQ-16_0/logfiles/testen2s.0001.srvlog -hn 5

```

23. Configure the entries in /database/asiql60/interfaces using the dscp command.

In our example, 10.61.110.152 was configured as the IP address of the S2 server.

```

[sybase_user@s2 testen]$ dscp
>> open
ok
Session 1 InterfacesDriver>> add testen2s Service: [ASE]
Transport Type: [tcp]
Transport Address: 10.61.110.152 2640 Transport Type: [tcp]
Transport Address:
Security Mechanism [] :
HA Failoverserver:
Retry Count:
Retry Delay:
Added testen2s
Session 1 InterfacesDriver>> exit

```

24. Check the size of the new online database on the S2 server.

Note: The resultant cloned database (testen2s) is the same size as the original (testen2).

```

[sybase_user@s2 testen]$ isql -Stesten2s -UDBA -Psql
1> select * from sp_iqstatus() where name like '%Main IQ Block%'
2> go
Name
Value
-----Main IQ Blocks Used:
770091564 of 939197518, 81%=23501Gb, Max Block#: 1050129902

1> sp_iqdbsize
2> go
Database
PhysicalBlocks      770091564
KBytes              18629918208
Pages               90949971
CompressedPages     90775386
NBlocks             582184944
CatalogBlocks       7536
RLVLogBlocks         0
RLVLogKBytes         0
-----

```

5.2 Snapshot Copy–Based Backup of SAP IQ Database During Reads and Writes

Test Objectives

- Demonstrate the resilience of database's Snapshot copies during the read-write process.
- Create at least three Snapshot copies while executing four read-write processes.

Test Procedure

1. Check the table sizes and CPU's utilization level. Execute `sp_iqconnection` in ISQL GUI.

The screenshot shows the ISQL GUI with the command `sp_iqconnection` entered in the SQL Statements pane. The Results pane displays a table with 15 rows of connection statistics.

ConnHandle	Name	User	LastReqTime	ReqType	IQCmdType	LastIQCmdTime	IQCursors	LowestIQCursorState	IQThreads	TxnID	ConnCreateTime
1	218	Sybase Central 1	2016-05-31 18:54:15.930	CLOSE	NONE	May 31, 2016 6:35:09 PM	0	NONE	0	21,252,352	May 31, 2016 4:34:45
2	221	isql	2016-05-31 18:40:26.106	EXEC_ANY_BHM	INSERT	May 31, 2016 6:40:26 PM	1	INITIALIZED	24	21,252,699	May 31, 2016 6:40:26
3	222	isql	2016-05-31 18:42:03.875	EXEC_ANY_BHM	INSERT	May 31, 2016 6:42:03 PM	1	INITIALIZED	2	21,252,804	May 31, 2016 6:42:03
4	223	isql	2016-05-31 18:43:42.059	EXEC_ANY_BHM	INSERT	May 31, 2016 6:43:42 PM	1	INITIALIZED	2	21,252,907	May 31, 2016 6:43:42
5	224	isql	2016-05-31 18:44:43.827	EXEC_ANY_BHM	INSERT	May 31, 2016 6:44:43 PM	1	INITIALIZED	3	21,252,975	May 31, 2016 6:44:43
6	225	isql	2016-05-31 18:45:38.514	EXEC_ANY_BHM	INSERT	May 31, 2016 6:45:38 PM	1	INITIALIZED	3	21,253,036	May 31, 2016 6:45:38
7	226	isql	2016-05-31 18:46:14.057	EXEC_ANY_BHM	INSERT	May 31, 2016 6:46:14 PM	1	INITIALIZED	0	21,253,077	May 31, 2016 6:46:14
8	227	isql	2016-05-31 18:47:45.907	EXEC_ANY_BHM	INSERT	May 31, 2016 6:47:45 PM	1	INITIALIZED	0	21,253,174	May 31, 2016 6:47:45
9	228	isql	2016-05-31 18:48:35.086	EXEC_ANY_BHM	INSERT	May 31, 2016 6:48:35 PM	1	INITIALIZED	0	21,253,247	May 31, 2016 6:48:35
10	229	isql	2016-05-31 18:49:23.636	EXEC_ANY_BHM	INSERT	May 31, 2016 6:49:23 PM	1	INITIALIZED	0	21,253,280	May 31, 2016 6:49:23
11	230	isql	2016-05-31 18:49:51.120	EXEC_ANY_BHM	INSERT	May 31, 2016 6:49:51 PM	1	INITIALIZED	0	21,253,311	May 31, 2016 6:49:51
12	231	isql	2016-05-31 18:51:56.063	EXEC_ANY_BHM	INSERT	May 31, 2016 6:51:56 PM	1	INITIALIZED	0	21,253,444	May 31, 2016 6:51:56
13	232	isql	2016-05-31 18:52:32.973	EXEC_ANY_BHM	INSERT	May 31, 2016 6:52:32 PM	1	INITIALIZED	0	21,253,485	May 31, 2016 6:52:32
14	233	isql	2016-05-31 18:53:07.141	EXEC_ANY_BHM	INSERT	May 31, 2016 6:53:07 PM	1	INITIALIZED	0	21,253,524	May 31, 2016 6:53:07
15	234	SQL_DBC_19429728	2016-05-31 18:55:07.021	OPEN	IQUTILITYOPENCURSOR	May 31, 2016 6:55:07 PM	0	NONE	0	21,253,591	May 31, 2016 6:53:56

2. Execute four read-write processes. Each script alternately executes the select and insert commands in the tables.

Process XX15-Db2 - tables: 1 (768KB) and xx (9.31TB) (select and insert alternated)
 Process XX14-Db2 - tables: 4a (1.57TB) and 3a (2.55TB) (select and insert alternated)
 Process XX13-DB2 - tables: dd (20.3GB) and ee (7.39GB) (select and insert alternated)
 Process XX12-DB2 - tables: 2a (1.28TB) and 1a (2.31TB) (select and insert alternated)

3. Check if Snapshot copies can be created and if there are any error messages. In this step, we execute the select and insert functions in tables from the database testen2.

```
[sybase_user@sl1]$ ps -ef | grep testen2
```

```
500 803 13255 0 11:21 pts/0 00:00:00 /database/asiql60/OCS-15_0/bin/isql -
Stesten2 -U<user> -? -i insertXX13-DB2 -o insertXX13-DB2_log
500 5381 13255 0 11:23 pts/0 00:00:00 /database/asiql60/OCS-15_0/bin/isql - Stesten2 -U<user> -?
-i insertXX12-DB2 -o insertXX12-DB2_log
500 11328 1 99 May13 ? 256-19:57:54 /database/asiql60/IQ- 16_0/bin64/iqsrsv16 @testen2.cfg
testen2.db -gn 25 -o /database/asiql60/IQ- 16_0/logfiles/testen2.0005.srvlog -hn 5
500 13144 13143 0 11:25 pts/1 00:00:00 isql -Stesten2 -U<user> -? -i snap_testen2b.sql -o
snap_testen2b_log
500 13792 13255 0 11:00 pts/0 00:00:00 /database/asiql60/OCS-15_0/bin/isql - Stesten2 -U<user> -?
-i insertXX14-Db2 -o insertXX14-Db2_log
```

```
500 16893 13849 0 09:59 pts/1 00:00:00 /database/asiql60/OCS-15_0/bin/isql - Stesten2 -U<user> -?
-i insertXX15-Db2 -o insertXX15-Db2_log
```

4. Run the `sp_iqdbspaceinfo` command to know the size of each database to use in the next steps.

```
[sybase_user@sl ~]$ isql -Stesten2 -U<user> -P<password>
1> sp_iqdbspaceinfo
dbspace_name,object_type,owner,object_name,object_id,id,columns,indexes,metadata
IQ_SYSTEM_MAIN,table,<user>,tb_data_backup_iq,6010,889,0B,0B,768K
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_1,4578,780,0B,0B,768K
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_10,4695,789,1.53M,0B,8.75M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_11,4708,790,10.7G,0B,11.1M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_12,4721,791,1.18M,0B,6.62M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_13,4734,792,11G,0B,9.15M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_14,4747,793,1.18M,0B,6.59M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_15,4760,794,2.18G,0B,7.9M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_1a,4788,795,2.31T,0B,154M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_2,4591,781,2.21M,0B,11.6M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_2a,4801,796,1.28T,0B,85.8M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_3,4604,782,141G,0B,13.6M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_3a,4814,797,2.55T,0B,171M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_4,4617,783,1.81M,0B,9.93M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_4a,4827,798,1.57T,0B,108M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_5,4630,784,21.3G,0B,14.4M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_5a,4840,799,7.53G,0B,7.15M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_6,4643,785,1.53M,0B,8.31M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_7,4656,786,21.3G,0B,14.1M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_8,4669,787,1.53M,0B,8.59M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_9,4682,788,21G,0B,14M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_aa,5243,830,768K,0B,4.12M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_bb,5256,831,20.6G,0B,13.7M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_cc,5269,832,214M,0B,4.62M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_dd,5282,833,20.2G,0B,13.1M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_ee,5295,834,7.39G,0B,6.62M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_ff,5308,835,768K,0B,4.12M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_gg,5321,836,768K,0B,4.12M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_hh,5334,837,768K,0B,4.12M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_ii,5347,838,768K,0B,4.12M
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_jj,5360,839,0B,0B,768K
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_ll,5373,840,0B,0B,768K
IQ_SYSTEM_MAIN,table,<user>,tb_data_X_teste_xx,4866,801,9.31T,0B,608M
```

5. Check the status of servers and SAP IQ services during the execution of the read-write processes.

Note: An attempt was made to execute five simultaneous reads and writes. However, this was unsupported because the amount of memory configured in our test setup was insufficient. Because of insufficient memory, we created a swap file. Our test setup does not have a swap area created for SAP IQ. The following measurements characterize the situation where five simultaneous processes were executed.

a. Run the `top` command to check the real-time CPU utilization.

```
[sybase_user@sl]$ top
top - 14:00:19 up 22 days, 5:53, 3 users, load average: 49.99, 50.18, 49.53
Tasks: 762 total, 1 running, 761 sleeping, 0 stopped, 0 zombie
Cpu(s) : 99.2%us, 0.3%sy, 0.0%ni, 0.4%id, 0.1%wa, 0.0%hi, 0.1%si, 0.0%st
Mem: 99054396k total, 64187964k used, 34866432k free, 554116k buffers
Swap: 2097148k total, 0k used, 2097148k free, 5876908k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU  %MEM    TIME+  COMMAND
 11328 sybase_u   20   0 66.0g 28g  29m S 2386.6 29.7 369860:52 iqsrv16
 21471 sybase_u   20   0 15576 1792 932 R  1.0   0.0   0:00.10 top
```

b. Run the `sar` command to check the CPU utilization of `sybase_user`.

```
[sybase_user@s1]$ sar
```

Linux 2.6.32-504.el6.x86_64 (s1)		06/01/2016		_x86_64_		(24 CPU	
12:00:01 AM	CPU	%user	%nice	%system	%iowait	%steal	%idle
12:10:01 AM	all	95.04	0.00	0.38	2.16	0.00	2.43
...							
...							
05:30:01 AM	all	94.68	0.00	0.36	2.29	0.00	2.67
05:40:01 AM	all	94.54	0.00	0.36	2.17	0.00	2.93
05:50:01 AM	all	95.34	0.00	0.36	1.97	0.00	2.32
06:00:01 AM	all	94.95	0.00	0.36	2.12	0.00	2.57
06:10:01 AM	all	95.07	0.00	0.36	2.12	0.00	2.45
06:20:01 AM	all	94.88	0.00	0.36	2.11	0.00	2.66
06:30:01 AM	all	95.21	0.00	0.36	2.02	0.00	2.41
06:40:01 AM	all	94.83	0.00	0.36	2.15	0.00	2.67
...							
06:40:01 AM	CPU	%user	%nice	%system	%iowait	%steal	%idle
06:50:01 AM	all	94.92	0.00	0.36	2.15	0.00	2.58
07:00:01 AM	all	95.10	0.00	0.36	2.06	0.00	2.48
...							
09:50:01 AM	all	0.01	0.00	0.03	12.53	0.00	87.43
10:00:01 AM	all	5.48	0.00	0.04	11.97	0.00	82.51
..							
01:10:01 PM	all	99.23	0.00	0.33	0.22	0.00	0.22
01:20:01 PM	all	99.11	0.00	0.32	0.28	0.00	0.28
...							
01:20:01 PM	CPU	%user	%nice	%system	%iowait	%steal	%idle
01:30:01 PM	all	99.04	0.00	0.34	0.32	0.00	0.31
01:40:01 PM	all	99.10	0.00	0.33	0.28	0.00	0.28
01:50:01 PM	all	99.31	0.00	0.32	0.18	0.00	0.19
02:00:01 PM	all	99.10	0.00	0.33	0.27	0.00	0.29
02:10:02 PM	all	99.08	0.00	0.33	0.27	0.00	0.32
Average:	all	93.51	0.00	0.35	2.00	0.00	4.15

c. Run the `free` command to check the memory utilization.

Note: The configuration file of the `testen2` database was configured with 40TB. However, it can be configured with more.

```
[sybase_user@s1 NETAPP]$ free
```

	total	used	free	shared	buffers	cached
Mem:	99054396	64181080	34873316	6668	554120	5878656
-/+ buffers/cache:	57748304	41306092				
Swap:	2097148	0	2097148			

- d. Run the `iostat` command to check the IOPS in various disks.

```
[sybase_user@s1]$ sar

Linux 2.6.32-504.el6.x86_64 (s1)          06/01/2016      _x86_64_          (24 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           58.70    0.00    0.37    1.20    0.00   39.72

Device:            tps    Blk_read/s    Blk_wrtn/s    Blk_read    Blk_wrtn
sda                 0.74         0.63         14.36     1213980     27605912
sdb                 0.09         0.00         65.58        5944     126102016
...
sdw                 3.05        383.21        819.88    736824277    1576426112
sdx                 0.00         0.00         0.00         2184           0
sdac                 2.99        371.83        820.05    714942432    1576757504
sdu                 3.04        380.16        820.95    730961365    1578488832
sdaa                 2.99        371.28        821.12    713869645    1578802880
sds                 3.09        427.59        848.05    822157110    1630584384
sdae                 2.99        370.92        820.40    713186645    1577417408
sdbd                 9.58        306.30       3496.84    588946364    6723537984
sdbb                 0.09         0.00         65.72         3800     126372288
sdag                 3.04        379.93        820.05    730514197    1576749056
sdai                 2.99        371.84        819.80    714946133    1576264384
sdak                 2.99        370.60        821.39    712563826    1579324800
sdbh                 0.06         0.00         53.76         3476     103373312
sdam                 3.02        375.76        820.48    722484658    1577578752
sdbf                 0.06         0.00         53.66         3476     103177984
sdbg                 0.00         0.00         0.00         2184           0
sdbe                 0.00         0.00         0.00         2184           0
sdao                 2.99        371.68        818.98    714640597    1574700352
sdat                 0.00         0.00         0.00         2408         4032
sdas                 3.12        395.70        820.78    760827762    1578159488
sdaq                 2.99        370.99        820.44    713324842    1577493312
sdax                 0.00         0.00         0.00         2184           0
sdau                 3.06        383.52        820.86    737411605    1578316032
sdap                 0.00         0.00         0.00         2184           0
sdav                 0.00         0.00         0.00         2184           0
sdbj                 0.06         0.00         54.05         8968     103929088
sdbk                 0.00         0.00         0.00         2524           0
sdbl                 0.06         0.00         53.79         3348     103418624
sdbo                 0.00         0.00         0.00         408           0
sdbn                 0.06         0.00         53.84         3348     103524480
...
```

- e. Run the `mpstat` command to check the CPU utilization.

Observe that the percentage of idle is high (approximately 40%), whereas the CPU used by the kernel is low (approximately 0.34).

```
[sybase_user@s1 NETAPP]$ mpstat

Linux 2.6.32-504.el6.x86_64 (s1)          06/01/2016      _x86_64_          (24 CPU)

02:13:43 PM CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest   %idle
02:13:43 PM all    58.71    0.00    0.34    1.20     0.00    0.03    0.00    0.00   39.72
```

Test Results

No error messages display from the server or application during the execution creation of Snapshot copies. The creation of a Snapshot copy does not disrupt or corrupt the database, instance, or tables.

6 Integrity Check of Snapshot Copy–Based Backup During Read-Write Process

6.1 Test Objectives

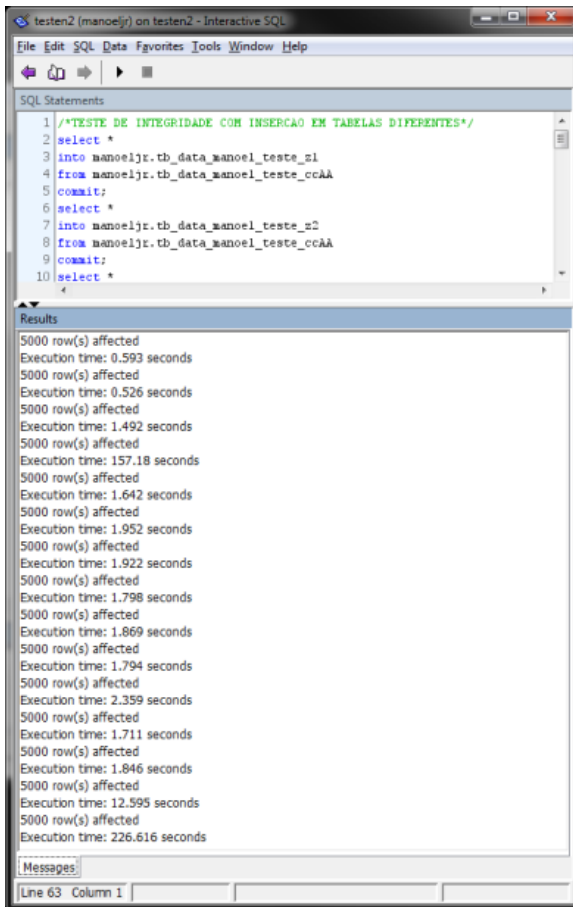
- Demonstrate the resilience of Snapshot copies of the databases during the read-write process.
- Create at least three Snapshot copies during the execution of the processes and then check the integrity of data from these Snapshot copies.

This step is more comprehensive than the previous test, because it checks the integrity of data and not just the success of execution.

6.2 Test Procedure

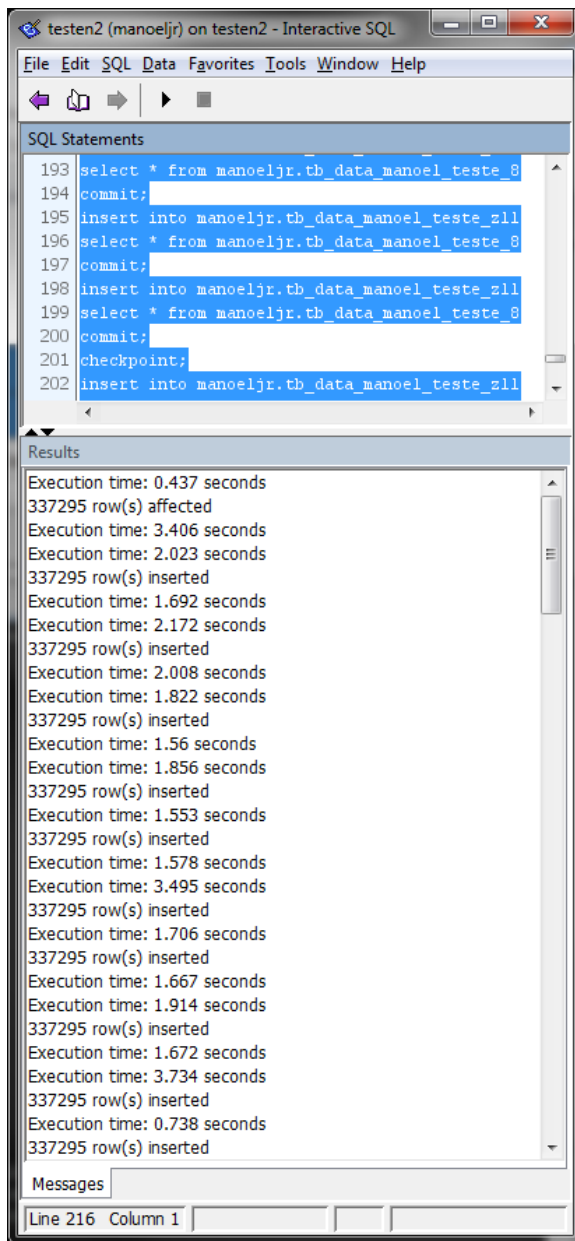
1. Two scripts were executed: one with select, insert, and commit functions and another one using SAP IQ procedures.
 - a. Process with select, insert, and commit functions in the various tables.

```
SET ROWCOUNT 5000
select *
into user.tb_data_X_teste_z1
from user.tb_data_X_teste_ccAA
commit;
select *
into user.tb_data_X_teste_z2
from user.tb_data_X_teste_ccAA
commit;
select *
into user.tb_data_X_teste_z3
from user.tb_data_X_teste_ccAA
commit;
select *
into user.tb_data_X_teste_z4
from user.tb_data_X_teste_ccAA
commit;
select *
into user.tb_data_X_teste_z5
from user.tb_data_X_teste_ccAA
commit;
select *
into user.tb_data_X_teste_z6
from user.tb_data_X_teste_ccAA
commit;
select *
into user.tb_data_X_teste_z7
from user.tb_data_X_teste_ccAA
commit;
```



b. Process with select, insert, and commit functions in two different tables.

```
select getdate();
select *
into user.tb_data_X_teste_z11
from user.tb_data_X_teste_8
commit;
checkpoint;
insert into user.tb_data_X_teste_z11
select * from user.tb_data_X_teste_8
commit;
checkpoint;
insert into user.tb_data_X_teste_z11
select * from user.tb_data_X_teste_8
commit;
checkpoint;
insert into user.tb_data_X_teste_z11
select * from user.tb_data_X_teste_8
commit;
checkpoint;
...
select getdate()
go
```



c. Process with the Procedure function in one table.

```

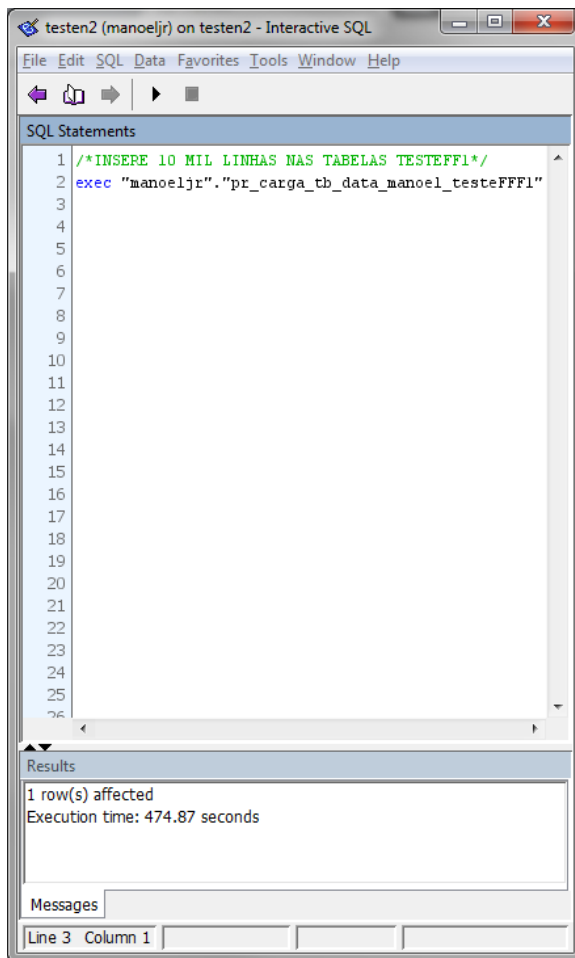
ALTER PROCEDURE "user"."pr_carga_tb_data_X_testeFFF1" as
begin
declare @contador integer
set @contador = 0
declare @i integer
set @i = 1
WHILE @i <= 10000
begin
insert into user.tb_data_X_testeFFF1 (data_bk, message_bk10, message_bk20, message_bk30,
message_bk40, message_bk50) select getdate(),
'zxc0987654321!@#f$% "1*()_+?><MNBVCXZx10#$$ "&*()_
+zxc0987654321!@#f$% "&*()_+?:><MNBVCXZ|!@#f$% "&*()_+ /;.,mnbvcxz\asdfghjklç~]
[ `poiuytrewq1234567890- =+_)(*&"%$#@!QWERTYUIOP`{}`^
ÇLKJHGFDSA|ZXCVBNM<?:?/;.,?><MNBVCXZ|ASDFGHJKLÇ^`{`POIUYTREWQ!@#f$% "&*()_+=-
0987654321qwertyuiop [ ]~çlkjhgfdsa\zxcvbnm,.;/;.,mnbvcxz\?:> <MNBVCXZ|}^ÇLKJHGFDSA
{`POIUYTREWQ=- 0987654321+_)(*&"%$#@!JKLÇ^`{`POIUYTREWQ!@#f$% "&*()_+=- 0987654321qwertyuiop [

```

```

]~çlkjhgfdsa\zxcvbnm,.;//;.,mnbcxz\?:> <MNBVCXZ|}^ÇLKJHGFDSA {`POIUYTREWQ=- 0987654321+_
( ``$#@',
'zxc0987654321x@#$$ ``&*( )_+?:><MNBVCXZ|r@#$$ ``&*( )_+ zxc0987654321u@#$$ ``&*( )_
+?:><MNBVCXZ|b@#$$ ``&*( )yx_ +/; .,mnbcxz\asdfghjklç~] [ `poiuytrewq1234567890-
=_ ) (*& ``$#@!QWERTYUIOP`{ }^ÇLKJHGFDSA{ZXCVCBNM<>: ? /; .,?:><MNBVCXZ|ASDFGHJKLÇ^}
{`POIUYTREWQ!@#$$ ``&*( )_ += -0987654321qwertyuiop [ ]~çlkjhgfdsa\zxcvbnm,.;// ;.,mnbcxz\?:>
<MNBVCXZ|}^ÇLKJHGFDSA{`POIUYTREWQ=- 0987654321+_ ) (*& ``$#@!JKLÇ^}{`POIUYTREWQ!@#$$ ``&*( )_ +=-
0987654321qwertyuiop [ ]~çlkjhgfdsa\zxcvbnm,.;//;., mnbcxz\?:><MNBVCXZ|}^ÇLKJHGFDSA
{`POIUYTREWQ=- 0987654321+_ ) (*& ``$#@',
...
set @contador = @contador + 1
if @contador = 500
begin
insert into user.tb_data_X_testeFFF1 (data_bk, data_bk_ANTES_DO_COMM_IT, message_bk10,
message_bk20, message_bk30, message_bk40, message_bk50) select getdate(), getdate(),
'BK_COMM_IT_ANTESBK_COMM_IT_ANTES_BK_COMM_IT_ANTES GRAVANDO LINHA DE REGISTRO COM HORARIO NA
COLUNA BK_COM M_IT_A NTES A INFORMACAO DO HORARIO ANTES DO COMIT DA QUANTIDADE DE LINHAS
DETERMINADO NA PROCEDUREREzxc098765432uo1!@#f$$ ``1 *() + ?><MNBVCXZx1@#$$ ``&*( )_
+zxc09uo87654321!@#$$ ``&*( )_ + ?WQ!@#$$ ``&*( )_+=- 0987654321qwertyuiop [
]~çlkjhgfdsa\zxcvbnm,.;cvbnm,.; //;.,mnbcX&*** (ç\?:> ,mnbcvMNBVCXZ|}^ÇLKJHGFDSA{`POI UYTREWQ=-
$#@BK_COMM_IT_ANTESBK_COMM_IT_ANTES_BK_COMM_IT_ANTES GRAVANDO LINHA DE REGISTRO COM HORARIO NA
COLUNA BK_COMM_IT _ANTES',
...
set @i = @i + 1
commit transaction
insert into <user>.tb_data_X_testeFFF1 (data_bk, data_bk_DEPOIS_DO_COMM_IT, message_bk10,
message_bk20, message_bk30, message_bk40, message_bk50) select getdate(), getdate(),
'BK_COMM_IT_DEPOIS BK_COMM_IT_DEPOIS_BK_COMM_IT_DEPOISGRAVANDO LINHA DE REGISTRO COMHORARIONA
COLUNABK_COMM_I T_DEPOIS_A_INFORMACAODO HORARIODEPOISDO COMIT DA QUANTIDADE DELINHAS DETERMINADO
NA PROCEDURERE zxc098765432uo1!@ # f$$ ``1*() +?> <MNBVCXZx1@#$$ ``&*( )_ +
zxc09uo87654321!@#$$ ``&*( )_+?WQ!@#$$ ``&*( )_+=- 0987654321qwertyuiop [
]~çlkjhgfdsa\zxcvbnm,.;cvbnm,.; //;.,mc\?:> ,mnbcvMNBVCXZ|} ^ÇLKJHGFDSA {`POIUYTREWQ=-
$#@BK_COMM_IT_DEPOISBK_COMM_IT_DEPOIS_BK_COMM_IT_DEPOIS GRAVANDO LINHADE REGISTRO COM HORARIO
NA COLUNABK_COMM_IT_ DEPOIS',
...
set @i = @i + 1
select @contador = 2
begin transaction
end
set @i = @i + 1
end
end

```



2. Create three Snapshot copies using the begin backup process (SAP IQ – backup virtual encapsulated) and NetApp Snap Creator Framework. This example uses a Python script named neRO, which calls ONTAP APIs using NetApp SDK.

```
[sybase_user@s1 NETAPP]$ vi snap_testen2.sh
isql -Stesten2 -U<user> -P<password> -i snap_testen2b.sql -o snap_testen2b_log_teste- integrity
[sybase_user@s1 NETAPP]$ vi snap_testen2b.sql
select getdate()
go
begin
backup database full virtual encapsulated
'cd /home/sybase_user/neRO;/home/sybase_user/neRO/neRO --action cg --conf
/home/sybase_user/neRO/conf/base2.conf --vserver db --snap snapY'
to '/database/snapt2/snapY.full'
end
go
select getdate()
go
```

3. Execute the three processes simultaneously and create three Snapshot copies in sequence.

Note: Check the Snapshot creation time in the testen2.iqmsg log file.

```
Wed Jun  1 19:36:49 EDT 2016
Wed Jun  1 19:37:51 EDT 2016
Wed Jun  1 19:40:24 EDT 2016
```

4. Check the Snapshot copies created in the first FAS8060 storage system.

```
superman::> volume snapshot show
```

Vserver	Volume	Snapshot	---Blocks---		
			Size	Total%	Used%
db	b1	snap1	100.3GB	5%	11%
		snap2	27.25MB	0%	0%
		snap3	33.73MB	0%	0%
		snap4	560.2MB	0%	0%
	b10	snap1	96.27GB	5%	10%
		snap2	8.78MB	0%	0%
		snap3	7.76MB	0%	0%
		snap4	640KB	0%	0%
	b28	snap1	96.28GB	5%	10%
		snap2	8.68MB	0%	0%
		snap3	7.77MB	0%	0%
		snap4	924KB	0%	0%
	...				

5. Check if the restore files from the SAP IQ database are created.

```
[sybase_user@s1 snap2]$ ls -la
```

```
total 904396
drwxrwxr-x 2 sybase_user sybase_user 4096 Jun  1 19:40 .
drwxr-xr-x 6 sybase_user sybase_user 4096 May 26 15:22 ..
-rw-rw-r-- 1 sybase_user sybase_user 308126924 Jun  1 19:37 snap2.full.1
-rw-rw-r-- 1 sybase_user sybase_user 308947140 Jun  1 19:38 snap3.full.1
-rw-rw-r-- 1 sybase_user sybase_user 309012676 Jun  1 19:40 snap4.full.1
```

6. Check the time it takes to create a Snapshot copy.

In this example, the execution time for each Snapshot copy was 20 seconds. Although the creation of Snapshot copies is instantaneous, the time indicated in the output is the time taken to read the file that contains the volume names needed to create a consistency group.

```
[sybase_user@s1 NETAPP]$ cat snap_testen2b_log_teste-integrity
```

```
getdate()
-----
Jun  1 2016  7:36:51.006523PM
(1 row affected)
Snapshot created.
getdate()
-----
Jun  1 2016  7:37:12.557438PM
(1 row affected)

getdate()
-----
Jun  1 2016  7:37:55.105433PM
(1 row affected)
Snapshot created.
getdate()
-----
Jun  1 2016  7:38:15.0054753PM
(1 row affected)

getdate()
-----
Jun  1 2016  7:40:18.007572PM
(1 row affected)
Snapshot created.
getdate()
-----
Jun  1 2016  7:40:40.385944PM
(1 row affected)
```

7. Check the execution of processes and Snapshot copies in the *srvlog file.

```
[sybase_user@s1 logfiles]$ tail -50 testen2.0005.srvlog
...
I. 06/01 19:41:52. Insert for 'tb_data_X_teste_zll' completed in 1 seconds. 337295 rows
inserted.
I. 06/01 19:41:54. Insert for 'tb_data_X_teste_zll' completed in 0 seconds. 337295 rows inserted.
I. 06/01 19:41:55. Starting checkpoint of "testen2" (testen2.db) at Wed Jun 01 2016 19:41
I. 06/01 19:41:58. Finished checkpoint of "testen2" (testen2.db) at Wed Jun 01 2016 19:41
I. 06/01 19:41:59. Insert for 'tb_data_X_teste_zll' completed in 1 seconds. 337295 rows
inserted.
I. 06/01 19:42:01. Insert for 'tb_data_X_teste_zll' completed in 0 seconds. 337295
rows inserted.
...
neRO - NetApp Storage Management Tool Version 1.8
Action: cg
Consistency Group operation on NetApp: 10.61.110.100 vservers: db - volume: b1 - snapshot: snap2
<results status="passed">
  <cg-id>709095114</cg-id>
</results>
...
Consistency Group operation on NetApp: 10.61.110.100 vservers: db - volume: b3 - snapshot: snap2
<results status="passed">
  <cg-id>717114680</cg-id>
</results>
...
Consistency Group operation on NetApp: 10.61.110.100 vservers: db - volume: s7 - snapshot: snap2
<results status="passed">
  <cg-id>1482638871</cg-id>
</results>
...
Consistency Group operation committed successfully - cg-id: 709095114 - volume: b1 - snapshot:
snap2
<results status="passed"></results>

Consistency Group operation committed successfully - cg-id: 1287572411 - volume: b5 - snapshot:
snap2
<results status="passed"></results>
...
Consistency Group operation on NetApp: 10.61.110.100 vservers: db - volume: b17 - snapshot: snap3
<results status="passed">
  <cg-id>1817608558</cg-id>
</results>
...
Consistency Group operation on NetApp: 10.61.110.100 vservers: db - volume: b22 - snapshot: snap3
<results status="passed">
  <cg-id>1010357004</cg-id>
</results>
...
Consistency Group operation committed successfully - cg-id: 1919192066 - volume: s10 -
snapshot: snap3
<results status="passed"></results>
...
Consistency Group operation on NetApp: 10.61.110.100 vservers: db - volume: b1 - snapshot: snap4
<results status="passed">
  <cg-id>1548683536</cg-id>
</results>
Consistency Group operation on NetApp: 10.61.110.100 vservers: db - volume: b5 - snapshot: snap4
<results status="passed">
  <cg-id>2040167801</cg-id>
</results>
...
Consistency Group operation committed successfully - cg-id: 1240902199 - volume: b14 - snapshot:
snap4
<results status="passed"></results>
Consistency Group operation committed successfully - cg-id: 1679511642 - volume: b13 - snapshot:
snap4
<results status="passed"></results>
...
```

8. Check the execution in the other log files.

a. Restore the file.

```
[sybase_user@sl logfiles]$ pwd
/database/asiql60/IQ-16_0/logfiles

[sybase_user@sl logfiles]$ cat backup.syb
...
BACKUP, -1350596655.0, testen1.db, ASIQ, '2016-05-26 20:03:27.193', <user>, Full, Arch,
/database/snap1/snap2.full, ''
BACKUP, 1518303185.0, testen1.db, ASIQ, '2016-05-26 22:11:25.005', <user>, Full, Arch,
/database/snap1/snap7.full, ''
BACKUP, -704686127.0, testen2.db, ASIQ, '2016-05-30 16:44:09.574', <user>, Full, Arch,
/database/snap2/snap2.full, ''
BACKUP, -813721647.0, testen2.db, ASIQ, '2016-05-30 17:08:21.573', <user>, Full, Arch,
/database/snap2/snap1.full, ''
BACKUP, -847292463.0, testen2.db, ASIQ, '2016-05-30 20:43:29.668', <user>, Full, Arch,
/database/snap2/snap1.full, ''
BACKUP, -687900719.0, testen2.db, ASIQ, '2016-06-01 19:36:45.001', <user>, Full, Arch,
/database/snap2/snap2.full, ''
BACKUP, -687900719.0, testen2.db, ASIQ, '2016-06-01 19:37:47.290', <user>, Full, Arch,
/database/snap2/snap3.full, ''
BACKUP, -1853925423.0, testen2.db, ASIQ, '2016-06-01 19:40:19.678', <user>, Full, Arch,
/database/snap2/snap4.full, ''
```

b. Check the insert processes in another log file from the testen2 database.

```
[sybase_user@sl testen]$ tail -f testen2.iqmsg
I. 06/01 19:13:28. 0000861927 Txn 21351076 0 21351076
I. 06/01 19:13:29. 0000861927 [20896]: Insert for 'tb_data_X_teste_z2'
completed in 1 seconds. 300000 rows inserted.
I. 06/01 19:13:29. 0000861927 Cmt 21351079
I. 06/01 19:13:29. 0000861927 PostCmt 0
I. 06/01 19:13:31. 0000861927 Txn 21351082
I. 06/01 19:13:31. 0000861927 Cmt 21351083
I. 06/01 19:13:31. 0000861927 PostCmt 0
I. 06/01 19:13:31. 0000861927 Txn 21351084
I. 06/01 19:13:31. 0000861927 Cmt 21351085
I. 06/01 19:13:31. 0000861927 PostCmt 0
I. 06/01 19:13:33. 0000861927 Txn 21351086
I. 06/01 19:13:34. 0000861927 [20896]: Insert for 'tb_data_X_teste_z3'
completed in 0 seconds. 300000 rows inserted.
I. 06/01 19:13:34. 0000861927 Cmt 21351089
I. 06/01 19:13:34. 0000861927 PostCmt 0
I. 06/01 19:13:37. 0000861927 Txn 21351092
I. 06/01 19:13:37. 0000861927 Cmt 21351093
I. 06/01 19:13:37. 0000861927 PostCmt 0
I. 06/01 19:13:37. 0000861927 Txn 21351094
I. 06/01 19:13:37. 0000861927 Cmt 21351095
I. 06/01 19:13:37. 0000861927 PostCmt 0
I. 06/01 19:13:39. 0000861927 Txn 21351096
0 21351092 0 21351094 0 21351096
...
I. 06/01 19:36:43. 0000863615 ValidateBackup
I. 06/01 19:36:43. 0000863615 ValidateBackup
I. 06/01 19:36:43. 0000863615 Chk
I. 06/01 19:36:44. 0000863615 ChkDone [NumTxnCP: 5] I. 06/01 19:36:44. 0000863615
PrepareForBackup
I. 06/01 19:36:44. 0000863615 PostChk
I. 06/01 19:36:45. 0000863474 Txn 21352699 0 21352699 I. 06/01 19:36:45. 0000863615 DoBackup
I. 06/01 19:36:45. 0000863615
backed u
p: 8608
I. 06/01 19:36:45. 0000863615
I. 06/01 19:36:45. 0000863474
'tb_data_X_teste_z14' co
mpleted in 0 seconds. 337295 rows inserted.
I. 06/01 19:36:45. 0000863474 Cmt 21352700 I. 06/01 19:36:45. 0000863474 PostCmt 0
I. 06/01 19:36:48. 0000863615
I. 06/01 19:36:49. 0000863615
Up: 860
```

```

8
at 2016-06-01 19:36:49
    IQ Blocksize 32768
    backup Blocksize 32808
    blocking factor 25
    buffer size 820216
...

```

- Execute the restore in the S2 server using Snapshot copies snap2 and snap4 and check the differences.

Note: Observe that the size of snap4 has increased due to the constant writes.

```

superman::> volume snapshot show

```

Vserver	Volume	Snapshot	Size	Total%	Used%
db	b1	snap1	100.3GB	5%	11%
		snap2	27.25MB	0%	0%
		snap3	33.73MB	0%	0%
		snap4	10.14GB	1%	1%
	b10	snap1	96.27GB	5%	10%
		snap2	8.78MB	0%	0%
		snap3	7.76MB	0%	0%
		snap4	8.04GB	0%	1%

- Mount all LUNs in the S2 server.

```

[sybase_user@s2 neRO]$ ./neRO --action lun-mount --conf /home/sybase_user/neRO/conf/base2.conf -
-vserver db --snap snap2 --clone-name cloneX --igroup s2 --lun-id 400
[root@s2 ~]# multipath -v3
...

[root@s2 ~]# multipath -ll
3600a098038303041622b493837665939 dm-116 NETAPP,LUN C-Mode
size=1.0T features='4 queue_if_no_path pg_init_retries 50 retain_attached_hw_handle'
hwhandler='1 alua' wp=rw
|-+- policy='round-robin 0' prio=50 status=active
|  |- 5:0:1:420 sdmy 70:416 active ready running
|  |- 6:0:1:420 sdsq 135:480 active ready running
|  |- 5:0:3:420 sdpu 131:320 active ready running
|  `-- 6:0:3:420 sdvm 68:640 active ready running
`+-+ policy='round-robin 0' prio=10 status=enabled
|  |- 5:0:0:420 sdlm 68:336 active ready running
|  |- 6:0:0:420 sdrf 133:400 active ready running
|  |- 5:0:2:420 sdoj 128:496 active ready running
|  `-- 6:0:2:420 sdub 66:560 active ready running
3600a098038303041493f493433703345 dm-88 NETAPP,LUN C-Mode
...

output [root@s2 ~]# for (( i = 1; i <= 9 ; i=i+1 )); do sanlun lun show -p | grep 'ONTAP\|3600'|
grep -A1 -w cloneX_snap2__b$i | sed 's/db:\vol\cloneX_snap2__b[0-9]\|g' | awk -F":" '{
printf " \"$2" " }' | awk '{ print "multipath " "{\n" "\t wwid " $2 "\n" "\t alias " $1
"_cloneX_snap2" "\n}"; done

[root@s2 ~]# for (( i = 10; i <= 32 ; i=i+1 )); do sanlun lun show -p | grep 'ONTAP\|3600'| grep
-A1 -w cloneX_snap2__b$i | sed 's/db:\vol\cloneX_snap2__b[0-9][0-9]\|g' | awk -F":" '{
printf " \"$2" " }' | awk '{ print "multipath " "{\n" "\t wwid " $2 "\n" "\t alias " $1
"_cloneX_snap2" "\n}"; done

[root@s2 ~]# for (( i = 1; i <= 9 ; i=i+1 )); do sanlun lun show -p | grep 'ONTAP\|3600'| grep -
A1 -w cloneX_snap2__s$i | sed 's/db:\vol\cloneX_snap2__s[0-9]\|g' | awk -F":" '{ printf "
"$2" " }' | awk '{ print "multipath " "{\n" "\t wwid " $2 "\n" "\t alias " $1 "_cloneX_snap2"
"\n}"; done

[root@s2 ~]# for (( i = 10; i <= 32 ; i=i+1 )); do sanlun lun show -p | grep 'ONTAP\|3600'| grep
-A1 -w cloneX_snap2__s$i | sed 's/db:\vol\cloneX_snap2__s[0-9][0-9]\|g' | awk -F":" '{
printf " \"$2" " }' | awk '{ print "multipath " "{\n" "\t wwid " $2 "\n" "\t alias " $1
"_cloneX_snap2" "\n}"; done

```

```
[root@s2 ~]# service multipathd restart
output [root@s2 ~]# ls /dev/mapper/
```

```
b10_cloneX
b25_cloneX_snap2
s15_cloneX_snap2p1
b10_cloneXp1
b25_cloneX_snap2p1
b10_cloneX_snap2
b32_cloneX_snap2
b10_cloneX_snap2p1
b26_cloneXp1
s16_cloneX_snap2
b11_cloneX
b26_cloneX_snap2
...
s15_cloneX_snap2
```

11. Stop the SAP IQ server testen2.

```
[root@s2 ~]# rescan.sh -r -s
Scanning HOST: host5
...
Checking system ...
The following 1 server(s) are owned by '500'
## Owner PID Started CPU Time Additional Information
-- -----
00:02:57 SVR:testen2s DB:testen2 PORT:2640
/database/asiql60/IQ-16_0/bin64/iqsrv16 @testen2.cfg testen2.db -gn 25 -o
/databas e/asiql60/IQ-16_0/logfiles/testen2s.0001.srvlog -hn 5
--
Please note that 'stop_iq' will shut down a server completely without regard for users,
connections, or load process status. For more control, use the 'dbstop' utility, which has
options that control stopping servers based on active connections.

Do you want to stop the server displayed above <Y/N>? y

Shutting down server (11154) ... Checkpointing server (11154) ...
Server shutdown.
```

12. Our example uses the LUNs based on snap2. If required, you can unmount all LUNs based on other Snapshot copies. For example, all LUNs based in snap1 were unmounted.

Note: This step is optional.

```
[sybase_user@s2 neRO]$ ./neRO --action lun-unmount --conf /home/sybase_user/neRO/conf/base2.conf
--vserver db --snap snap1 --clone-name cloneX --igroup s2 --lun-id 300
```

13. Associate the new LUNs with the RAW devices.

```
[root@s2 ~]#
/bin/raw /dev/raw/raw6 /dev/mapper/b1_cloX_snap2p1
/bin/raw /dev/raw/raw17 /dev/mapper/b5_cloneX_snap2p1
/...
/bin/raw /dev/raw/raw40 /dev/mapper/b32_cloneX_snap2p1
/bin/raw /dev/raw/raw43 /dev/mapper/b30_cloneX_snap2p1
/...
/bin/raw /dev/raw/raw14 /dev/mapper/s14_cloneX_snap2p1
/bin/raw /dev/raw/raw16 /dev/mapper/s16_cloneX_snap2p1
...

[root@s2 ~]# raw -qa
/dev/raw/raw6: bound to major 253, minor 7
/dev/raw/raw7: bound to major 253, minor 130
...
/dev/raw/raw41: bound to major 253, minor 146
...
```

14. From the S1 server, copy the restore file (catalog) to the S2 server.

```
[sybase_user@s1 NETAPP]$ cd /database/snapt2

[sybase_user@s1 snapt2]$ ls
snap2full.1 snap3.full.1 snap4.full.1

[sybase_user@s1 snapt2]$ scp snap* sybase_user@10.61.110.152:/database/snapt2
sybase_user@10.61.110.152's password:
snap2full.1      snap3.full.1.    snap4.full.1

[root@s2 ~]# cd /database/snapt2/ [root@s2 snapt2]# ls
snap2full.1 snap3.full.1 snap4.full.1
```

15. Check if any databases are up and running in the S2 server.

```
[sybase_user@s2 ~]$ stop_iq

Checking system ...
There are no servers running on this system
```

16. Run the following script to execute the restore from the catalog file:

```
[sybase_user@s2 testen]$ vi restore_vbe_testen2.sh
dbisqlc -c
"uid=DBA;pwd=sql;eng=s2_eng_dumy;dbn=utility_db;links=tcPIP{host=10.61.110.152;port=2640}" -q
restore_vbe_testen2.sql

[sybase_user@s2 testen]$ cat restore_vbe_testen2.sql
RESTORE DATABASE '/database/testen/testen2.db'
FROM '/database/snapt2/snap2.full'
go
```

Note: Observe that there are no files related to database testen2.

```
[sybase_user@s2 testen]$ ls -la
total 44
lrwxrwxrwx 1 sybase_user sybase_user 13 May 31 15:30 DWSIQ.NETAPPMMAIN001_DB2 ->
/dev/raw/raw6
lrwxrwxrwx 1 sybase_user sybase_user 13 May 31 17:36 DWSIQ.NETAPPTEMP001_DB2 ->
/dev/raw/raw7
-rw-rw-r-- 1 sybase_user sybase_user 223 May 30 17:12 INSTANCIA_DUMMY.cfg
-rwxrwxrwx 1 sybase_user sybase_user 129 May 31 15:14 restore_vbe_testen2.sh
-rwxrwxrwx 1 sybase_user sybase_user 85 Jun 2 12:30 restore_vbe_testen2.sql
drwxrwxr-x 2 sybase_user sybase_user 4096 Jun 2 12:32 result_snap1
-rw-rw-r-- 1 sybase_user sybase_user 11577 May 31 17:42 s2_eng_dumy_log
-rw-rw-r-- 1 sybase_user sybase_user 7271 May 31 17:42 s2_eng_dumy_log.stderr
```

17. Before beginning the restore, start an initial instance in the S2 server.

```
[sybase_user@s2 testen]$ start_iq @INSTANCIA_DUMMY.cfg
Starting server s2_eng_dumy on s2 at port 2640 (06/02 12:40:45)
Run Directory
Server Executable
Server Output Log
Server Version
16.0.0/Linux 2.6.18-194.el5
Open Client Version : N/A
User Parameters : '@INSTANCIA_DUMMY.cfg' Default Parameters : -gn 25
I. 06/02 12:40:51. SAP IQ
I. 06/02 12:40:51. Version 16.0
I. 06/02 12:40:51. (64bit mode)
I. 06/02 12:40:51. Copyright 1992-2015 by SAP AG or an SAP affiliate company. All rights reserved
I. 06/02 12:40:51. Copyright (c) 2015 SAP SE or an SAP affiliate company.
I. 06/02 12:40:51. All rights reserved.
I. 06/02 12:40:51. Use of this software is governed by the Sybase License Agreement. Refer to
http://www.sybase.com/softwarelicenses.
I. 06/02 12:40:51. Processors detected: 2 (containing 24 logical processors)
I. 06/02 12:40:51. Maximum number of processors the server will use: 2 physical processor(s), 12
core(s)
I. 06/02 12:40:51. Running Linux 2.6.32-504.el6.x86_64 #1 SMP Tue Sep 16 01:56:35 EDT 2014 on
X86_64
I. 06/02 12:40:51. Server built for X86_64 processor architecture
```

```

I. 06/02 12:40:51. 524288K of memory used for caching
I. 06/02 12:40:51. Minimum cache size: 524288K, maximum cache size: 524288K
I. 06/02 12:40:51. Using a maximum page size of 4096 bytes
I. 06/02 12:40:51. Multiprogramming level: 25
I. 06/02 12:40:51. Warning: -gn value of 25 is too low for -gm value of 20
I. 06/02 12:40:51. Automatic tuning of multiprogramming level
=====

```

IQ server starting with:

```

is disabled
20 connections          (      -gm )
 5 cmd resources        ( -iqgovern )
1000 threads            (      -iqmt )
 100 Kb thread stack size ( -igtss )
100000 Kb thread memory size ( -iqmt * -igtss )
24 IQ number of cpus ( -iqnumbercpus )
0 MB maximum size of IQMSG file ( -iqmsgsz )
0 copies of IQMSG file archives ( -iqmsgnum )
=====

```

```

I. 06/02 12:40:51. Database server started at Thu Jun 02 2016 12:40
I. 06/02 12:40:51. Trying to start Sharedmemory link ...
I. 06/02 12:40:51. SharedMemory started successfully
I. 06/02 12:40:51. Trying to start TCPIP link ...
I. 06/02 12:40:56. TCPIP link started successfully
I. 06/02 12:40:56. Now accepting requests
New process id is 2135
Server started successfully

```

18. Execute the restore and check that all files are in the /database/testen directory.

```
[sybase_user@s2 testen]$ ./restore_vbe_testen2.sh
```

```
[sybase_user@s2 testen]$ ls -la
```

```

total 304672
drwxrwxr-x 3.  sybase_user sybase_user 4096 Jun  2 13:20 .
drwxr-xr-x 5.  sybase_user sybase_user 4096 May 30 16:11 ..
lrwxrwxrwx 1.  sybase_user sybase_user 13 May 31 15:30 DWSIQ.NETAPPMAN001_DB2 ->
/dev/raw/raw6
lrwxrwxrwx 1  sybase_user sybase_user 13 May 31 17:36 DWSIQ.NETAPPTEMP001_DB2 -
>/dev/raw/raw7
-rw-rw-r-- 1  sybase_user sybase_user 223 May 30 17:12 INSTANCIA_DUMMY.cfg
-rwxrwxrwx 1  sybase_user sybase_user 129 May 31 15:14 restore_vbe_testen2.sh
-rwxrwxrwx 1  sybase_user sybase_user 84 Jun  2 12:49 restore_vbe_testen2.sql
drwxrwxr-x 2  sybase_user sybase_user 4096 Jun  2 12:32 result_snap1
-rw-rw-r-- 1  sybase_user sybase_user 17661 Jun  2 13:20 s2_eng_dumy_log
-rw-rw-r-- 1  sybase_user sybase_user 8359 Jun  2 13:20 s2_eng_dumy_log.stderr
-rwxrwxrwx 1  sybase_user sybase_user 308126924 Jun  2 12:59 snap2full.1
-r--r--r-- 1  sybase_user sybase_user 3780608 Jun  2 13:20 testen2.db
-rw-r--r-- 1  sybase_user sybase_user 5332 Jun  2 13:20 testen2.iqmsg
-rw-rw-r-- 1  sybase_user sybase_user 533 Jun  2 13:20 testen2.lmp

```

19. Start the testen2.db database using the following configuration file:

```
[sybase_user@s2 testen]$ cat testen2.cfg
```

```

-n testen2s
-c 512m
-gc 120
-gd DBA
-gk DBA
-gl ALL
-gm 20
-gp 4096
-gr 6000
-ti 60
-tl 300
-iqmc 20000
-iqtc 20000
-iqmt 1000
-iqgovern 5
-igtss 100000
-p 4096
-zo testen2.audit

```

```
-n testen2
-x tcpip{port=2640}
```

```
[sybase_user@s2 testen]$ start_iq @testen2.cfg testen2.db

Starting server testen2s on s2 at port 2640 (06/02 13:24:52)
Run Directory: /database/testen
Server Executable: /database/asiql60/IQ-16_0/bin64/iqsrv16
Server Output Log: /database/asiql60/IQ-16_0/logfiles/testen2s.0002.srvlog
Server Version: 16.0.110.2014/sp11
16.0.0/Linux 2.6.18-194.el5
Open Client Version : N/A
User Parameters : '@testen2.cfg' 'testen2.db'
Default Parameters : -gn 25
I. 06/02 13:24:58. SAP IQ
I. 06/02 13:24:58. Version 16.0
I. 06/02 13:24:58. (64bit mode)
I. 06/02 13:24:58. Copyright 1992-2015 by SAP AG or an SAP affiliate company. All rights reserved
I. 06/02 13:24:58. Copyright (c) 2015 SAP SE or an SAP affiliate company.
I. 06/02 13:24:58. All rights reserved.
I. 06/02 13:24:58. Use of this software is governed by the Sybase License Agreement.
I. 06/02 13:24:58. Refer to http://www.sybase.com/softwarelicenses.
...
=====
IQ server starting with:
    20 connections          (          -gm )
    5 cmd resources ( -iqgovern )
    1000 threads            (          -iqmt )
100000 Kb thread stack size ( -iqtss )
100000000 Kb thread memory size ( -iqmt * -iqtss )
24 IQ number of cpus ( -iqnumbercpus )
0 MB maximum size of IQMSG file ( -iqmsgsz )
0 copies of IQMSG file archives ( -iqmsgnum )
=====
I. 06/02 13:24:58.      Starting database "testen2" (/database/testen/testen2.db) at Thu Jun 02
2016 13:24
I. 06/02 13:24:59.      Database recovery in progress
I. 06/02 13:24:59.          Last checkpoint at Wed Jun 01 2016 19:36
I. 06/02 13:24:59.          Transaction log: testen2.log...
I. 06/02 13:24:59.          Forcing recovery without transaction log
I. 06/02 13:24:59.          Checkpointing...
I. 06/02 13:24:59.      Starting checkpoint of "testen2" (testen2.db) at Thu Jun 02 2016 13:24
I. 06/02 13:25:00.      Finished checkpoint of "testen2" (testen2.db) at Thu Jun 02 2016 13:25
I. 06/02 13:25:00.      Recovery complete
I. 06/02 13:25:01.      Update previous log GUID to: 83d5d74c-16cf-11e6-8000-cd095d02fa64
Update current log
GUID to :e213d904-28a3-11e6-8000-8353a2a1fad7
I. 06/02 13:25:01.      Database "testen2" (testen2.db) started at Thu Jun 02 2016 13:25
I. 06/02 13:25:01.      IQ Server testen2s.
I. 06/02 13:25:01.      Starting checkpoint of "testen2" (testen2.db) at Thu Jun 02 2016 13:25
I. 06/02 13:25:02.      Finished checkpoint of "testen2" (testen2.db) at Thu Jun 02 2016 13:25
I. 06/02 13:25:02.      Database server started at Thu Jun 02 2016 13:25
I. 06/02 13:25:02.      Trying to start SharedMemory link ...
I. 06/02 13:25:02.      SharedMemory link started successfully
I. 06/02 13:25:02.      Trying to start TCPIP link ...
I. 06/02 13:25:02.      Starting on port 2640
I. 06/02 13:25:07.      TCPIP link started successfully
I. 06/02 13:25:07.      Now accepting requests
New process id is 19721
Server started successfully
```

20. Check the contents of the /database/testen directory. Notice that they are all files related to the testen2 database.

```
[sybase_user@s2 testen]$ ls -la
total 305000
drwxrwxr-x 3 sybase_user sybase_user 4096 Jun  2 13:25 .
drwxr-xr-x 5 sybase_user sybase_user 4096 May 30 16:11 ..
lrwxrwxrwx 1 sybase_user sybase_user 13 May 31 15:30 DWSIQ.NETAPPMMAIN001_DB2 ->
/dev/raw/raw6
```

```

lrwxrwxrwx 1 sybase_user sybase_user 13 May 31 17:36 DWSIQ.NETAPPTEMP001_DB2 ->
/dev/raw/raw7
-rw-rw-r-- 1 sybase_user sybase_user 223 May 30 17:12 INSTANCIA_DUMMY.cfg
-rwxrwxrwx 1 sybase_user sybase_user 129 May 31 15:14 restore_vbe_testen2.sh
-rwxrwxrwx 1 sybase_user sybase_user 84 Jun 2 12:49 restore_vbe_testen2.sql
drwxrwxr-x 2 sybase_user sybase_user 4096 Jun 2 12:32 result_snap1
-rw-rw-r-- 1 sybase_user sybase_user 17932 Jun 2 13:22 s2_eng_dumy_log
-rw-rw-r-- 1 sybase_user sybase_user 11668 Jun 2 13:22 s2_eng_dumy_log.stderr
-rwxrwxrwx 1 sybase_user sybase_user 308126924 Jun 2 12:59 snap2full.1
-rw-r--r-- 1 root root 212 Jun 2 13:24 testen2.cfg
-rw----- 1 sybase_user sybase_user 3780608 Jun 2 13:25 testen2.db
-rw-r--r-- 1 sybase_user sybase_user 10542 Jun 2 13:25 testen2.iqmsg
-rw-rw-r-- 1 sybase_user sybase_user 632 Jun 2 13:25 testen2.lmp
-rw-rw-r-- 1 sybase_user sybase_user 327680 Jun 2 13:25 testen2.log

```

21. Check the postchecks in the *.iqmsg log file.

```

[sybase_user@s2 testen]$ cat testen2.iqmsg

I. 06/02 13:20:45. 0000000000 OpenDatabase Completed
I. 06/02 13:20:45. 0000000000 IQ cmd line srv opts: -iqmc 20000 -iqtc 20000 -iqmt 1000 - iqgo
vern 5
I. 06/02 13:20:45. 0000000000 IQ full cmd line: -n s2_eng_dumy -c 512m -gc 120 -gd DBA -gk D BA -
gl ALL -gm 20 -gp 4096 -gr 6000 -ti 60 -tl 300 -iqmc 20000 -iqtc 20000 -iqmt 1000 -iqgov ern 5 -
iqtss 100 -p 4096 -zo createdb.audit -x tcpip{port=2640} -o s2_eng_dumy_log -gn 25 -h n5
I. 06/02 13:20:45. 0000000000 DB: r/w, Main 3596MB Buffs=39960, Temp 3596MB Buffs=39960, Pgs
z=524288/32768blks/16bpc, BP=4
I. 06/02 13:20:45. 0000000000 DB: Frmt#: 23F/2T/1P (FF: 03/18/1999)
I. 06/02 13:20:45. 0000000000 DB: Versn: 16.0.110.2014/10214/P/sp11/Enterprise Linux64 - x86_64 -
2.6.18-194.el5/64bit/2015-11-21 01:29:07
...
I. 06/02 13:20:45. 0000000000 Collation ISO_BINENG, Case Ignore, Blank Padding On,
Comparisons are Conditioned
I. 06/02 13:20:45. 0000000000 Result Cache Max Pages: 0
I. 06/02 13:20:45. 0000000000 RcvyCmpl
I. 06/02 13:20:45. 0000000000 Chk
I. 06/02 13:20:46. 0000000000 ChkDone [NumTxnCP: 0]
I. 06/02 13:20:46. 0000000000 PostChk
I. 06/02 13:20:46. 0000000000 Result Cache Max Pages: 0
I. 06/02 13:20:46. 0000000000 CloseDatabase
...
I. 06/02 13:20:47. 0000000003 Chk
I. 06/02 13:20:48. 0000000003 ChkDone [NumTxnCP: 0]
I. 06/02 13:20:48. 0000000003 PostChk
I. 06/02 13:20:48. 0000000005 PostRestore
I. 06/02 13:20:48. 0000000007 Chk
I. 06/02 13:20:48. 0000000007 ChkDone [NumTxnCP: 0] I. 06/02 13:20:48. 0000000007 PostChk
I. 06/02 13:20:48. 0000000000 CloseDatabase
I. 06/02 13:24:59. 0000000000 OpenDatabase Completed
...

```

22. Compare the tables that are created in the testen2s database on the S2 server with the tables that were created in the testen2 database on the S1 server. Observe that the testen2 database on the S1 server has more tables than the testen2s database on the S2 server.

- testen2s database on the S2 server

The screenshot shows the 'testen2s' database in SQL Enterprise Manager. The left pane displays the database structure, with 'Tables' selected. The right pane shows a list of tables with columns 'Name', 'Owner', and 'Type'.

Name	Owner	Type
tb_data_manoeI_teste_yy11aa	manoeIjr	Table
tb_data_manoeI_teste_yy12aa	manoeIjr	Table
tb_data_manoeI_teste_yy13aa	manoeIjr	Table
tb_data_manoeI_teste_yy14aa	manoeIjr	Table
tb_data_manoeI_teste_yy15aa	manoeIjr	Table
tb_data_manoeI_teste_yy16aa	manoeIjr	Table
tb_data_manoeI_teste_yy17aa	manoeIjr	Table
tb_data_manoeI_teste_yy18aa	manoeIjr	Table
tb_data_manoeI_teste_yy19aa	manoeIjr	Table
tb_data_manoeI_teste_yy20aa	manoeIjr	Table
tb_data_manoeI_teste_yy21aa	manoeIjr	Table
tb_data_manoeI_teste_yy22aa	manoeIjr	Table
tb_data_manoeI_teste_yy23aa	manoeIjr	Table
tb_data_manoeI_teste_yy24aa	manoeIjr	Table
tb_data_manoeI_teste_z1	manoeIjr	Table
tb_data_manoeI_teste_z11	manoeIjr	Table
tb_data_manoeI_teste_z12	manoeIjr	Table
tb_data_manoeI_teste_z13	manoeIjr	Table
tb_data_manoeI_teste_z14	manoeIjr	Table
tb_data_manoeI_teste_z2	manoeIjr	Table
tb_data_manoeI_teste_z3	manoeIjr	Table
tb_data_manoeI_teste_z4	manoeIjr	Table
tb_data_manoeI_testeFF1	manoeIjr	Table
tb_data_manoeI_testeFFF	manoeIjr	Table
tb_data_manoeI_testeFFF1	manoeIjr	Table

– testen2 database on the S1 server

The screenshot shows the 'testen2' database in SQL Enterprise Manager. The left pane displays the database structure, with 'Tables' selected. The right pane shows a list of tables with columns 'Name', 'Owner', and 'Type'.

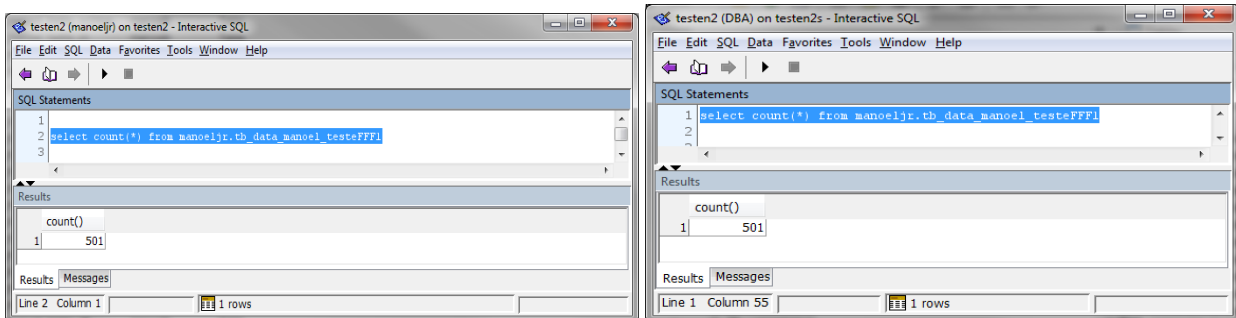
Name	Owner	Type
tb_data_manoeI_teste_yy23aa	manoeIjr	Table
tb_data_manoeI_teste_yy24aa	manoeIjr	Table
tb_data_manoeI_teste_z1	manoeIjr	Table
tb_data_manoeI_teste_z10	manoeIjr	Table
tb_data_manoeI_teste_z11	manoeIjr	Table
tb_data_manoeI_teste_z12	manoeIjr	Table
tb_data_manoeI_teste_z13	manoeIjr	Table
tb_data_manoeI_teste_z14	manoeIjr	Table
tb_data_manoeI_teste_z15	manoeIjr	Table
tb_data_manoeI_teste_z16	manoeIjr	Table
tb_data_manoeI_teste_z17	manoeIjr	Table
tb_data_manoeI_teste_z18	manoeIjr	Table
tb_data_manoeI_teste_z19	manoeIjr	Table
tb_data_manoeI_teste_z2	manoeIjr	Table
tb_data_manoeI_teste_z3	manoeIjr	Table
tb_data_manoeI_teste_z4	manoeIjr	Table
tb_data_manoeI_teste_z5	manoeIjr	Table
tb_data_manoeI_teste_z6	manoeIjr	Table
tb_data_manoeI_teste_z7	manoeIjr	Table
tb_data_manoeI_teste_z8	manoeIjr	Table
tb_data_manoeI_teste_z9	manoeIjr	Table
tb_data_manoeI_teste_z11	manoeIjr	Table
tb_data_manoeI_testeFF1	manoeIjr	Table
tb_data_manoeI_testeFFF	manoeIjr	Table
tb_data_manoeI_testeFFF1	manoeIjr	Table

23. Check the tb_data_X_teste_z11 and tb_data_X_testeFFF1 tables.

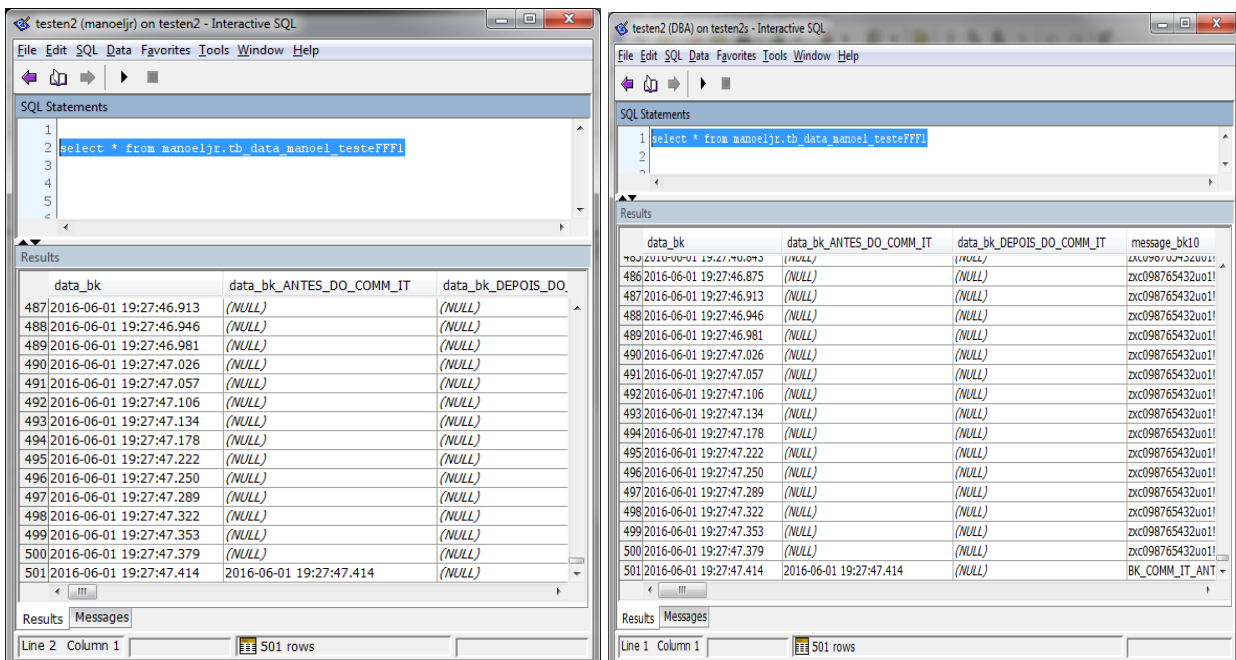
Note: In this example, when creating the snap2 Snapshot copy, the tb_data_X_teste_zll table was not created. Therefore, only the tb_data_X_testeFFF1 table is available in the testen2 database on the S1 server (source).

Note: The tb_data_X_testeFFF1 table created from snap2 is equivalent to the original tb_data_X_testeFFF1 table from testen2 database on the S1 server.

Note: The following screenshot shows the row count in the tb_data_X_testeFFF1 table.

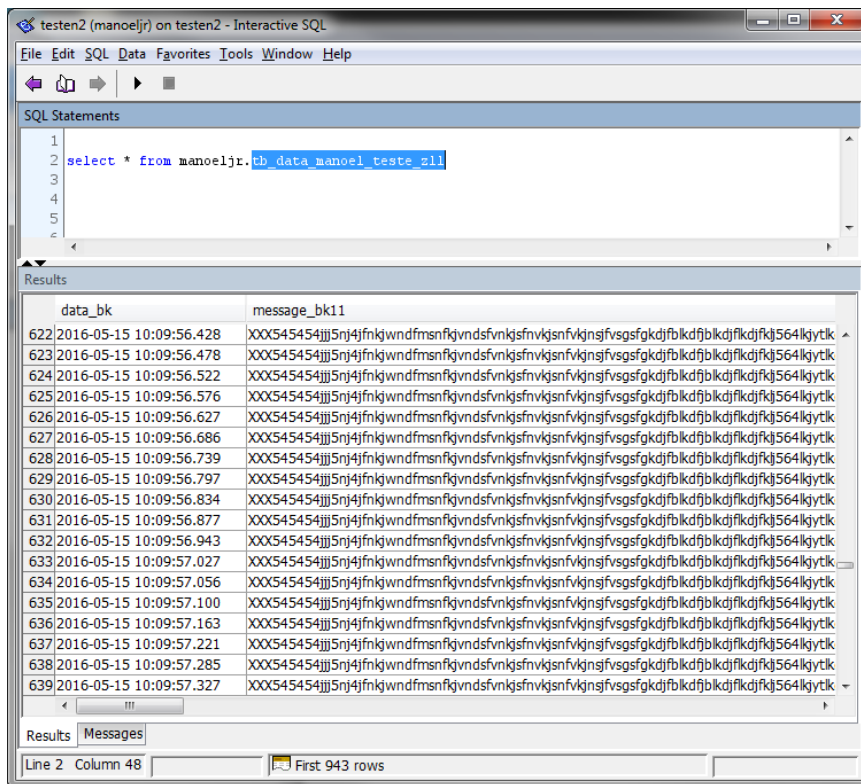


The select script output for the tb_data_X_testeFFF1 table is as follows:

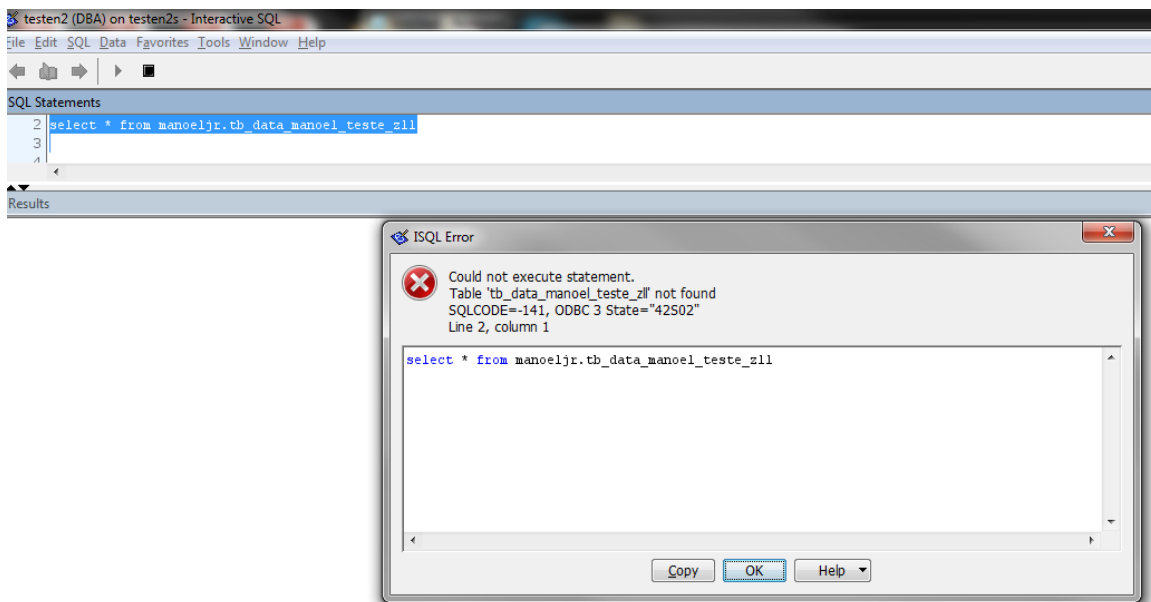


24. The tb_data_X_teste_zll table had not yet been created on the S2 server. To verify that the table does not exist, do as follows:

a. The script `select from tb_data_X_teste_zll` table at S1 server runs successfully.



- b. The script `select from tb_data_X_teste_zll` table at the S2 server results in error because the `tb_data_X_teste_zll` table is not created in the S2 server.



25. Mount the database from snap4 and then compare the number of tables from snap2 and snap4.
26. Stop the testen2s database on the S2 server.

```
[sybase_user@s2 ~]$ stop_iq
```

```

Checking system ...
The following 1 server(s) are owned by '500'
## Owner      PID      Started   CPU Time   Additional Information
-----
1: 500        19721    13:24    00:01:17    SVR:testen2s DB:testen2 PORT:2640
                /database/asiql60/IQ-16_0/bin64/iqsrv16 @testen2.cfg testen2.db -gn 25 -o
/database/asiql60/IQ-16_0/logfiles/testen2s.0002.srvlog -hn 5
--
Please note that 'stop_iq' will shut down a server completely without regard for users,
connections, or load process status. For more control, use the 'dbstop' utility, which has
options that control stopping servers based on active connections.

Do you want to stop the server displayed above <Y/N>? y
Shutting down server (19721) ...
Checkpointing server (19721) ...
Server shutdown.

```

27. Unmount the LUNs based on clones from the snap2 Snapshot copy (consistency group).

```

[sybase_user@s2 ~]$ cd neRO/

[sybase_user@s2 neRO]$ ./neRO --action lun-unmount --conf
/home/sybase_user/neRO/conf/base2.conf --vserver db --snap snap2 --clone-name cloneX --igroup s2
--lun-id 400

neRO - NetApp Storage Management Tool
Version 1.8
Action: lun-unmount

Unmapping LUN on NetApp: 10.61.110.100 vsriver: db - LUN: /vol/cloneX__snap2__b1/b1 - igroup: s2
<results status="passed"></results>
...

[sybase_user@s2 neRO]$ sudo service multipathd stop
Stopping multipathd daemon: [ OK ]

[sybase_user@s2 neRO]$ sudo multipath -F
Jun 02 22:52:15 | s2_bin: map in use

[sybase_user@s2 neRO]$ sudo rescan.sh -r -s
Scanning HOST: host5
...

[sybase_user@s2 neRO]$ sudo rescan-scsi-bus.sh -r

[sybase_user@s2 neRO]$ sudo service multipathd start
Starting multipathd daemon: [ OK ]

[sybase_user@s2 neRO]$ sudo multipath -v3
...

[sybase_user@s2 neRO]$ sudo multipath -ll
s2_bin (3600a098038303041622b49383766582d) dm-0 NETAPP,LUN C-Mode
size=200G features='4 queue_if_no_path pg_init_retries 50 retain_attached_hw_handle' hwhandler='1
alua' wp=rw
|-+- policy='round-robin 0' prio=50 status=active |
|- 5:0:3:26 sde 8:64 active ready running
| |- 6:0:1:26 sdg 8:96 active ready running
| |- 5:0:1:26 sdc 8:32 active ready running
| `-- 6:0:3:26 sdi 8:128 active ready running
`+- policy='round-robin 0' prio=10 status=enabled
|- 5:0:0:26 sdb 8:16 active ready running
|- 6:0:0:26 sdf 8:80 active ready running
|- 5:0:2:26 sdd 8:48 active ready running
`-- 6:0:2:26 sdh 8:112 active ready running

```

28. Delete all clones that are based on the snap2 Snapshot copy (consistency group).

```

[sybase_user@s2 neRO]$ ./neRO --action clone-delete --conf /home/sybase_user/neRO/conf/base2.conf
--vserver db --snap snap2 --clone-name cloneX

```

```

neRO - NetApp Storage Management Tool
Version 1.8
Action: clone-delete
Offlining volume on NetApp: 10.61.110.100 vserver: db - volume: cloneX__snap2__b1 <results
status="passed"></results>
Destroying volume on NetApp: 10.61.110.100 vserver: db - volume: cloneX__snap2__b1 <results
status="passed"></results>
....

```

29. Create the clones from the snap4 Snapshot copy (consistency group).

Note: The size of each clone is equal to that of the first Snapshot copy =~30MB.

```

superman::> volume snapshot show
...
cloneX__snap4__b1
  snap4                23.17MB    0%    0%
cloneX__snap4__b10
  snap4                33.15MB    0%    0%
cloneX__snap4__b11
  snap4                33.29MB    0%    0%
...
cloneX__snap4__b17
  snap4                32.61MB    0%    0%
cloneX__snap4__b18
  snap4                31.95MB    0%    0%
...

```

30. Mount all LUNs based on clones from the snap4 Snapshot copy (consistency group).

```

[sybase_user@s2 neRO]$ ./neRO --action lun-mount --conf /home/sybase_user/neRO/conf/base2.conf --
vserver db --snap snap4 --clone-name cloneX --igroup s2 --lun-id 500

neRO - NetApp Storage Management Tool Version 1.8
Action: lun-mount
LUN-ID Sequence starting with: 500

Mapping LUN on NetApp: 10.61.110.100 vserver: db - LUN: /vol/cloneX__snap4__b1/b1 - igroup: s2 -
LUN ID: 500
<results status="passed">
  <lun-id-assigned>500</lun-id-assigned> </results>

Mapping LUN on NetApp: 10.61.110.100 vserver: db - LUN: /vol/cloneX__snap4__b5/b5 - igroup: s2 -
LUN ID: 501
<results status="passed">
  <lun-id-assigned>501</lun-id-assigned> </results>
...

```

31. Run the following commands to rescan the server and find the LUNs.

Note that the database has two LUNs of sizes 250GB and 1TB.

```

[sybase_user@s2 neRO]$ sudo rescan.sh -r -s
[sudo] password for sybase_user: Scanning HOST: host5
...

[sybase_user@s2 neRO]$ sudo multipath -ll

size=250G features='4 queue_if_no_path pg_init_retries 50 retain_attached_hw_handle' hwhandler='1
alua' wp=rw
|-+- policy='round-robin 0' prio=50 status=active
|  |- 5:0:0:529 sdam 66:96 active ready running
|  |- 6:0:0:529 sdge 131:160 active ready running
|  |- 5:0:2:529 sddi 71:0 active ready running
|  `-- 6:0:2:529 sdja 8:320 active ready running
`+- policy='round-robin 0' prio=10 status=enabled
   |- 5:0:1:529 sdbx 68:176 active ready running
   |- 6:0:1:529 sdhp 133:240 active ready running
   |- 5:0:3:529 sdet 129:80 active ready running
   `-- 6:0:3:529 sdkl 66:400 active ready running
3600a098038303041622b49383766594e dm-11 NETAPP,LUN C-Mode
size=1.0T features='4 queue_if_no_path pg_init_retries 50 retain_attached_hw_handle' hwhandler='1

```

```

alua' wp=rw
|+- policy='round-robin 0'                                prio=50 status=active
| |- 5:0:1:513 sdbh 67:176                                active ready running
| |- 6:0:1:513 sdgz 132:240                                active ready running
| |- 5:0:3:513 sded 128:80                                active ready running
| `-- 6:0:3:513 sdjv 65:400                                active ready running
`+- policy='round-robin 0'                                prio=10 status=enabled
| |- 5:0:0:513 sdw 65:96                                   active ready running
| |- 6:0:0:513 sdfo 130:160                                active ready running
| |- 5:0:2:513 sdcs 70:0                                   active ready running
| `-- 6:0:2:513 sdik 135:64                                active ready running
...

```

32. Create the multipath.conf file in the /etc/ directory.

The following scripts facilitate the creation of the multipath.conf file.

```

output [root@sz2 ~]# for (( i = 1; i <= 9 ; i=i+1 )); do
grep -A1 -w cloneX__snap4__b$i | sed 's/db:\vol\cloneX__snap4__b[0-9]\///g' | awk -F":" '{
printf " \"$2\" \" \"}' | awk '{ print "multipath \" \"{\\n\" \"\\t wwid \" $2 \"\\n\" \"\\t alias \" $1
\"_cloneX__snap4\" \"\\n\"}";
done

[root@sz2 ~]# for (( i = 10; i <= 32 ; i=i+1 )); do sanlun lun show -p | grep 'ONTAP\\|3600'| grep
-A1 -w cloneX__snap4__b$i | sed 's/db:\vol\cloneX__snap4__b[0-9][0-9]\///g' | awk -F":" '{
printf " \"$2\" \" \"}' | awk '{ print "multipath \" \"{\\n\" \"\\t wwid \" $2 \"\\n\" \"\\t alias \" $1
\"_cloneX__snap4\" \"\\n\"}";
done

[root@sz2 ~]# for (( i = 1; i <= 9 ; i=i+1 )); do sanlun lun show -p | grep 'ONTAP\\|3600'| grep -
A1 -w cloneX__snap4__s$i | sed 's/db:\vol\cloneX__snap4__s[0-9]\///g' | awk -F":" '{ printf "
 \"$2\" \" \"}' | awk '{ print "multipath \" \"{\\n\" \"\\t wwid \" $2 \"\\n\" \"\\t alias \" $1 \"_cloneX__snap4\"
 \"\\n\"}";
done

[root@sz2 ~]# for (( i = 10; i <= 32 ; i=i+1 )); do sanlun lun show -p | grep 'ONTAP\\|3600'| grep
-A1 -w cloneX__snap4__s$i | sed 's/db:\vol\cloneX__snap4__s[0-9][0-9]\///g' | awk -F":" '{
printf " \"$2\" \" \"}' | awk '{ print "multipath \" \"{\\n\" \"\\t wwid \" $2 \"\\n\" \"\\t alias \" $1
\"_cloneX__snap4\" \"\\n\"}";
done

[root@sz2 ~]# cat /etc/multipath.conf | more
defaults {
    user_friendly_names no
}
blacklist {
    wwid 3600605b003bb2e5016b9be466f420e85
    devnode "^hd [a-z] \"
    devnode \"^(ram |raw| loop|fd |md| dm-|sr |scd| st)[0-9 ]*\"
    devnode \"^cciss. *\"
}
multipaths{
multipath {
    wwid 3600a098038303041622b493837665958
    alias s10_cloneX__snap4
}
multipath {
    wwid 3600a098038303041493f493433703361
    alias s11_cloneX__snap4
}
multipath {
    wwid 3600a098038303041622b493837665959
    alias s12_cloneX__snap4
}
multipath {
    wwid 3600a098038303041622b49383766595a
    alias s14_cloneX__snap4
}
multipath {
    wwid 3600a098038303041493f493433703362
    alias s15_cloneX__snap4
}

```

```

}
multipath {
}
    wwid 3600a098038303041622b49383766592f
    alias s16_cloneX_snap4
}
...
multipath {
    wwid 3600a098038303041493f493433703350
    alias b9_cloneX_snap4
}
...

[sybase_user@s2 neRO]$ sudo service multipathd restart
[sudo] password for sybase_user:
ok
Stopping multipathd daemon:    [ OK ]
Starting multipathd daemon:    [ OK ]

[sybase_user@s2 neRO]$ sudo chkconfig multipathd on

[sybase_user@s2 neRO]$ ls /dev/mapper/
b10_cloneX_snap4
b18_cloneX_snap4
b5_cloneX_snap4
s14_cloneX_snap4p1
b10_cloneX_snap4p1
b18_cloneX_snap4p1
b5_cloneX_snap4p1
s15_cloneX_snap4
..
b27_cloneX_snap4p1
...

[root@s2 ~]#
/bin/raw /dev/raw/raw6 /dev/mapper/b1_cloneX_snap4p1
/bin/raw /dev/raw/raw17 /dev/mapper/b5_cloneX_snap4p1
/...
/bin/raw /dev/raw/raw32 /dev/mapper/b20_cloneX_snap4p1
/bin/raw /dev/raw/raw34 /dev/mapper/b16_cloneX_snap4p1
/bin/raw /dev/raw/raw35 /dev/mapper/b21_cloneX_snap4p1
/...
/bin/raw /dev/raw/raw15 /dev/mapper/s15_cloneX_snap4p1
/bin/raw /dev/raw/raw16 /dev/mapper/s16_cloneX_snap4p1
...

[root@s2 ~]# raw -qa
/dev/raw/raw6: bound to major 253, minor 41
...
/dev/raw/raw18: bound to major 253, minor 17
/dev/raw/raw19: bound to major 253, minor 7
/dev/raw/raw20: bound to major 253, minor 43
...
/dev/raw/raw28: bound to major 253, minor 50
/dev/raw/raw29: bound to major 253, minor 15
/dev/raw/raw30: bound to major 253, minor 18
/dev/raw/raw31: bound to major 253, minor 30
/dev/raw/raw32: bound to major 253, minor 38
/dev/raw/raw34: bound to major 253, minor 35
...

```

33. After mounting all LUNs, restore the database using the snap4 Snapshot copy.

```

[sybase_user@s2 testen]$ ls
DWSIQ.NETAPPMMAIN001_DB2
INSTANCIA_DUMMY.cfg
restore_vbe_testen2.sql
result_snap2
s2_eng_dumy_log.stderr
testen2.cfg
DWSIQ.NETAPPTEMP001_DB2

```

```
restore_vbe_testen2.sh
result_snap1
s2_eng_dumy_log
snap4.full.1
```

```
[sybase_user@s2 testen]$ stop_iq
Checking system ...
There are no servers running on this system
```

34. Execute the same process from the previous steps (snap2) to restore the database from snap4:

- a. Start an initial instance.
- b. Check the initial instance.
- c. Execute the restore.
- d. Stop the initial instance.
- e. Start the database testen2 on the S2 server based on the snap4 Snapshot copy.

35. Check the current database.

```
[sybase_user@s2 testen]$ start_iq @INSTANCIA_DUMMY.cfg
Starting server s2_eng_dumy on s2 at port 2640 (06/02 23:48:20)
s2_eng_dumy_log
Run Directory: /database/testen
Server Executable: /database/asiql60/IQ-16_0/bin64/iqsrsv16
Server Output Log: s2_eng_dumy_log
Server Version: 16.0.110.2014/sp11
16.0.0/Linux 2.6.18-194.el5
Open Client Version : N/A
User Parameters : '@INSTANCIA_DUMMY.cfg'
Default Parameters : -gn 25
I. 06/02 23:48:26. SAP IQ
I. 06/02 23:48:26. Version 16.0
I. 06/02 23:48:26. (64bit mode)
...

[sybase_user@s2 testen]$ stop_iq
Checking system ...
The following 1 server(s) are owned by '500'
## Owner      PID      Started      CPU Time      Additional Information
--
1: 500        27601 23:48      00:00:00      SVR:s2_eng_dumy DB:utility_db PORT:2640
/database/asiql60/IQ-16_0/bin64/iqsrsv16 @INSTANCIA_DUMMY.cfg -gn 25 -hn 5

[sybase_user@s2 testen]$ cat restore_vbe_testen2.sql
RESTORE DATABASE '/database/testen/testen2.db'
FROM '/database/testen/snap4.full'
go
```

```

[sybase_user@s2 testen]$ ./restore_vbe_testen2.sh
[sybase_user@s2 testen]$ ls -la
total 305612
drwxrwxr-x 4 sybase_user sybase_user 4096 Jun  2 23:52 .
drwxr-xr-x 5 sybase_user sybase_user 4096 May 30 16:11 ..
lrwxrwxrwx 1 sybase_user sybase_user 13 May 31 15:30 DWSIQ.NETAPPMMAIN001_DB2 ->
/dev/raw/raw6
lrwxrwxrwx 1 sybase_user sybase_user 13 May 31 17:36 DWSIQ.NETAPPTTEMP001_DB2 ->
/dev/raw/raw7
-rw-rw-r-- 1 sybase_user sybase_user 223 May 30 17:12 INSTANCIA_DUMMY.cfg
-rwxrwxrwx 1 sybase_user sybase_user 129 May 31 15:14 restore_vbe_testen2.sh
-rwxrwxrwx 1 sybase_user sybase_user 85 Jun  2 23:50 restore_vbe_testen2.sql
drwxrwxr-x 2 sybase_user sybase_user 4096 Jun  2 12:32 result_snap1
drwxrwxr-x 2 sybase_user sybase_user 4096 Jun  2 23:46 result_snap2
-rw-rw-r-- 1 sybase_user sybase_user 20429 Jun  2 23:52 s2_eng_dumy_log
-rw-rw-r-- 1 sybase_user sybase_user 12757 Jun  2 23:52 s2_eng_dumy_log.stderr
-rwxrwxrwx 1 sybase_user sybase_user 309012676 Jun  2 23:51 snap4.full.1
-rw-r--r-- 1 root root 212 Jun  2 13:24 testen2.cfg
-r--r--r-- 1 sybase_user sybase_user 3846144 Jun  2 23:52 testen2.db
-rw-r--r-- 1 sybase_user sybase_user 5334 Jun  2 23:52 testen2.iqmsg
-rw-rw-r-- 1 sybase_user sybase_user 533 Jun  2 23:52 testen2.lmp

[sybase_user@s2 testen]$ stop_iq
Checking system ...
The following 1 server(s) are owned by '500'
## Owner PID Started CPU Time Additional Information
--
1: 500 27601 23:48 00:00:00 SVR:s2_eng_dumy DB:utility_db PORT:2640
/database/asiql60/IQ-16_0/bin64/iqsrv16 @INSTANCIA_DUMMY.cfg -gn 25 -hn 5

[sybase_user@s2 testen]$ stop_iq
Checking system ...
The following 1 server(s) are owned by '500'
## Owner PID Started CPU Time Additional Information
--
1: 500 27601 23:48 00:00:04 SVR:s2_eng_dumy DB:utility_db PORT:2640
/database/asiql60/IQ-16_0/bin64/iqsrv16 @INSTANCIA_DUMMY.cfg -gn 25 -hn 5
--
Please note that 'stop_iq' will shut down a server completely without regard for users,
connections, or load process status. For more control, use the 'dbstop' utility, which has
options that control stopping servers based on active connections.

Do you want to stop the server displayed above <Y/N>? y

Shutting down server (27601) ...
Checkpointing server (27601) ...
Server shutdown.

[sybase_user@s2 testen]$ start_iq @testen2.cfg testen2.db
Starting server testen2s on s2 at port 2640 (06/02 23:54:52)
Run Directory: /database/testen
Server Executable: /database/asiql60/IQ-16_0/bin64/iqsrv16
Server Output Log: /database/asiql60/IQ-16_0/logfiles/testen2s.0003.srvlog
Server Version: 16.0.110.2014/sp11
16.0.0/Linux 2.6.18-194.el5
Open Client Version : N/A
User Parameters : '@testen2.cfg' 'testen2.db'
Default Parameters : -gn 25
I. 06/02 23:54:59. SAP IQ
I. 06/02 23:54:59. Version 16.0
I. 06/02 23:54:59. (64bit mode)
I. 06/02 23:54:59. Copyright 1992-2015 by SAP AG or an SAP affiliate company. All rights reserved
I. 06/02 23:54:59. Copyright (c) 2015 SAP SE or an SAP affiliate company.
I. 06/02 23:54:59. All rights reserved.
I. 06/02 23:54:59. Use of this software is governed by the Sybase License Agreement.

```

```

I. 06/02 23:54:59. Refer to http://www.sybase.com/softwarelicenses.
I. 06/02 23:54:59. Processors detected: 2 (containing 24 logical processors)
...
I. 06/02 23:54:59. Starting database "testen2" (/database/testen/testen2.db) at Thu Jun 02 2016 23:54
I. 06/02 23:55:00. Database recovery in progress
I. 06/02 23:55:00.      Last checkpoint at Wed Jun 01 2016 19:40
I. 06/02 23:55:00.      Transaction log: testen2.log...
I. 06/02 23:55:00.      Forcing recovery without transaction log
I. 06/02 23:55:00.      Checkpointing...
I. 06/02 23:55:00.      Starting checkpoint of "testen2" (testen2.db) at Thu Jun 02 2016 23:55
I. 06/02 23:55:01.      Finished checkpoint of "testen2" (testen2.db) at Thu Jun 02 2016 23:55
I. 06/02 23:55:01.      Recovery complete
I. 06/02 23:55:02.      Update previous log GUID to: 83d5d74c-16cf-11e6-8000-cd095d02fa64 Update
current log GUID to :e4ffc0b2-28fb-11e6-8000-fd6f0756c909
I. 06/02 23:55:02.      Database "testen2" (testen2.db) started at Thu Jun 02 2016 23:55
I. 06/02 23:55:02.      IQ Server testen2s.
I. 06/02 23:55:02.      Starting checkpoint of "testen2" (testen2.db) at Thu Jun 02 2016 23:55
I. 06/02 23:55:03.      Finished checkpoint of "testen2" (testen2.db) at Thu Jun 02 2016 23:55
I. 06/02 23:55:03.      Database server started at Thu Jun 02 2016 23:55
I. 06/02 23:55:03.      Trying to start SharedMemory link ...
I. 06/02 23:55:03.      SharedMemory link started successfully
I. 06/02 23:55:03.      Trying to start TCPIP link ...
I. 06/02 23:55:03.      Starting on port 2640
I. 06/02 23:55:08.      TCPIP link started successfully
I. 06/02 23:55:08.      Now accepting requests
New process id is 2505
Server started successfully

[sybase_user@s2 testen]$ stop_iq
Checking system ...
The following 1 server(s) are owned by '500'
## Owner      PID      Started      CPU Time      Additional Information
--
1: 500        2505     23:54       00:00:02     SVR:testen2s DB:testen2 PORT:2640

/database/asiql60/IQ-16_0/bin64/iqsrv16 @testen2.cfg testen2.db -gn 25 -o /database/asiql60/IQ-
16_0/logfiles/testen2s.0003.srvlog -hn 5
--
Please note that 'stop_iq' will shut down a server completely without regard for users,
connections, or load process status. For more control, use the 'dbstop' utility, which has
options that control stopping servers based on active connections.

Do you want to stop the server displayed above <Y/N>? n

```

36. Check the size of the database testen2 based on the snap4 Snapshot copy (server name = testen2s).

Note: Observe that the cloned database on the S2 server is approximately 23.5TB, which is equal to the last status of database testen2 in the S1 server.

```

[sybase_user@s2 testen]$ isql -Stesten2s -UDBA -Psql
1> select * from sp_iqstatus() where name like '%Main IQ Block%'
2> go
Name
-----
Value
-----
Main IQ Blocks Used:
770101834 of 939197518, 81%=23501Gb, Max Block#: 1050129902
(1 row affected)

[sybase_user@s2 testen]$ isql -Stesten2s -UDBA -Psql
1> sp_iqdbsize
2> go

Database
PhysicalBlocks      770101834
KBytes              18630246848
Pages               90951214
CompressedPages      90776095
NBlocks              582195214
CatalogBlocks        8956

```

RLVLogBlocks	0
RLVLogKBytes	0

/database/testen/testen2.db	
(1 row affected)	
(return status = 0)	

37. Check the integrity and consistency of tables from both databases: testen2 from the S1 server and testen2s from the S2 server.

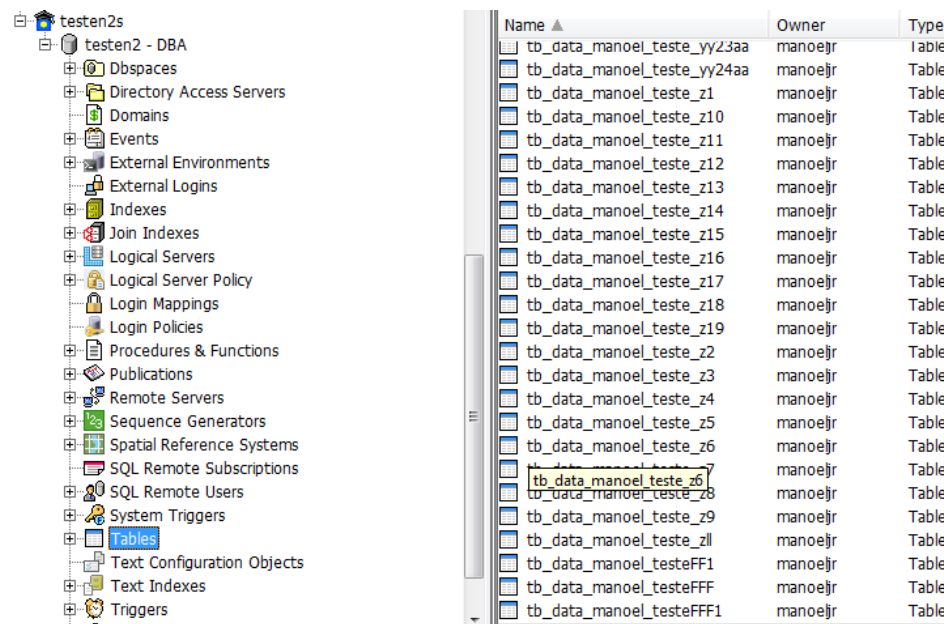
Note: Recall that the database based on the snap2 Snapshot copy did not contain the tb_data_manoeI_teste_zII table; however, the cloned database based on the snap4 Snapshot copy contains this table.

Note: Observe that the tb_data_manoeI_testeFFF1 is also present.

– testen2 database on the S1 server

Name	Owner	Type
tb_data_manoeI_teste_yy23aa	manoeIjr	Table
tb_data_manoeI_teste_yy24aa	manoeIjr	Table
tb_data_manoeI_teste_z1	manoeIjr	Table
tb_data_manoeI_teste_z10	manoeIjr	Table
tb_data_manoeI_teste_z11	manoeIjr	Table
tb_data_manoeI_teste_z12	manoeIjr	Table
tb_data_manoeI_teste_z13	manoeIjr	Table
tb_data_manoeI_teste_z14	manoeIjr	Table
tb_data_manoeI_teste_z15	manoeIjr	Table
tb_data_manoeI_teste_z16	manoeIjr	Table
tb_data_manoeI_teste_z17	manoeIjr	Table
tb_data_manoeI_teste_z18	manoeIjr	Table
tb_data_manoeI_teste_z19	manoeIjr	Table
tb_data_manoeI_teste_z2	manoeIjr	Table
tb_data_manoeI_teste_z3	manoeIjr	Table
tb_data_manoeI_teste_z4	manoeIjr	Table
tb_data_manoeI_teste_z5	manoeIjr	Table
tb_data_manoeI_teste_z6	manoeIjr	Table
tb_data_manoeI_teste_z7	manoeIjr	Table
tb_data_manoeI_teste_z8	manoeIjr	Table
tb_data_manoeI_teste_z9	manoeIjr	Table
tb_data_manoeI_teste_zII	manoeIjr	Table
tb_data_manoeI_testeFFF1	manoeIjr	Table
tb_data_manoeI_testeFFF1	manoeIjr	Table

– testen2s database on the S2 server



6.3 Test Results

Three Snapshot copies are created sequentially during the execution of three read-write processes. After this process ends, it is possible to check the integrity of the Snapshot copies (snap2 and snap4) at different times.

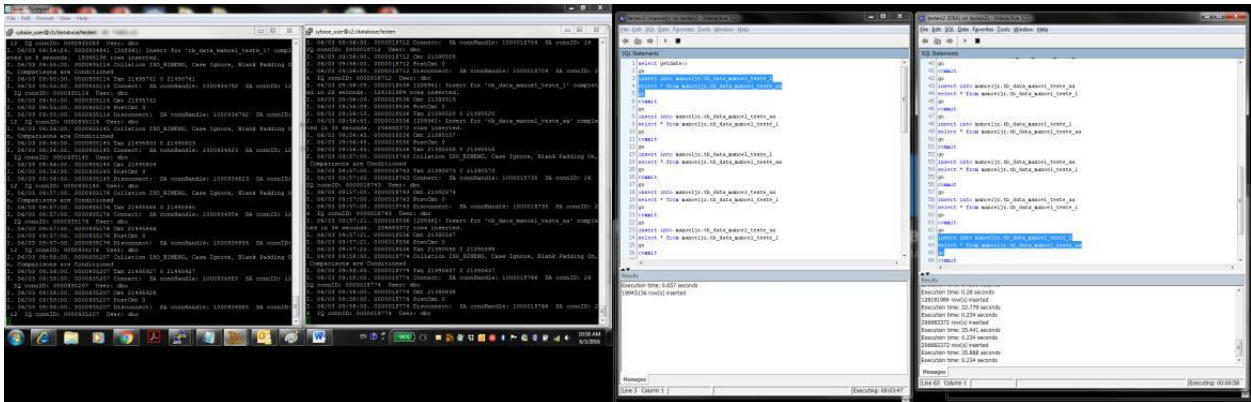
7 Integrity and Security Check During Simultaneous Reads and Writes Between Source and Cloned Databases

Test Objectives

- Demonstrate the resilience of the cloned and original database based on the Snapshot copy during simultaneous read-write processes.
- Check the integrity of database, table, and instance and whether the writing and reading process occurred without corruption of the data.

Test Procedure

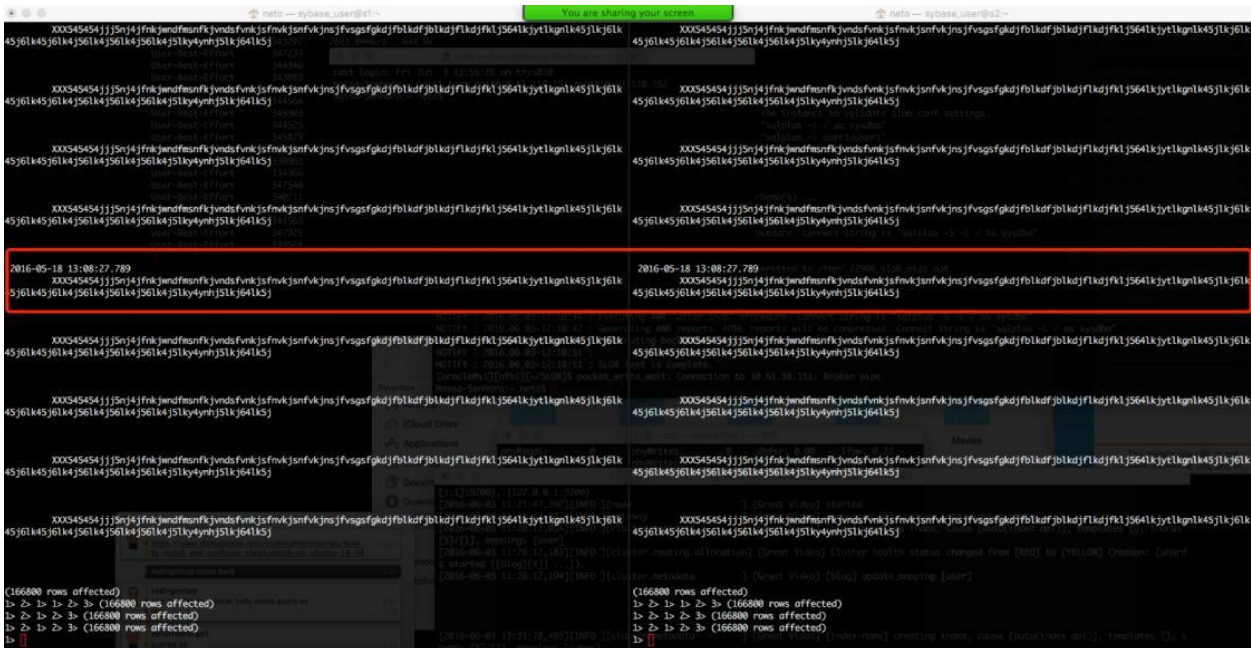
1. In the Sybase Central GUI, simultaneously execute the read and write operations on database 1 (source - testen2 at S1 server) and the cloned database (testen2s at S2 server).
Also, check the log files to follow the execution.



2. In the CLI, by running isql, simultaneously execute the read and write operations on database 1 (source - testen2 at S1 server) and the cloned database (testen2s at S2 server).

Also, check the log files to follow the execution. Note that in the different servers and different databases, we execute the two scripts simultaneously in the `tb_data_manoeel_teste_aa` and `tb_data_manoeel_teste_1` tables, in each database.

The read and write time is equal in both servers, as shown in the following screenshot. This proves that no interference is caused by simultaneous reads and writes to the source database and the clone.



7.1 Snapshot Copy of SAP IQ Cloned Database

Test Objective

Demonstrate that it is possible create a Snapshot copy of a cloned database. In this test, one Snapshot copy is created of the cloned database testen2s. The testen2 database is a clone of the testen2 database. This new Snapshot copy was presented to the S2 server.

Test Procedure

1. Create a Snapshot copy of the cloned database. In our example, the Snapshot copy is named snapteste.
2. Execute the following scripts on both servers (S1 and S2) and both databases: source and cloned.
 - Run the script on the S1 server:

```
[sybase_user@s1 NETAPP]$ cat teste-escrita.sql
select getdate()
go
insert into manoeljr.tb_data_manoel_teste_10
select * from manoeljr.tb_data_manoel_teste_12
go
commit
go
select getdate()
go
insert into manoeljr.tb_data_manoel_teste_10
select * from manoeljr.tb_data_manoel_teste_12
go
commit
select getdate()
```

```
[sybase_user@s1 NETAPP]$ ./teste-escrita_simultanea
```

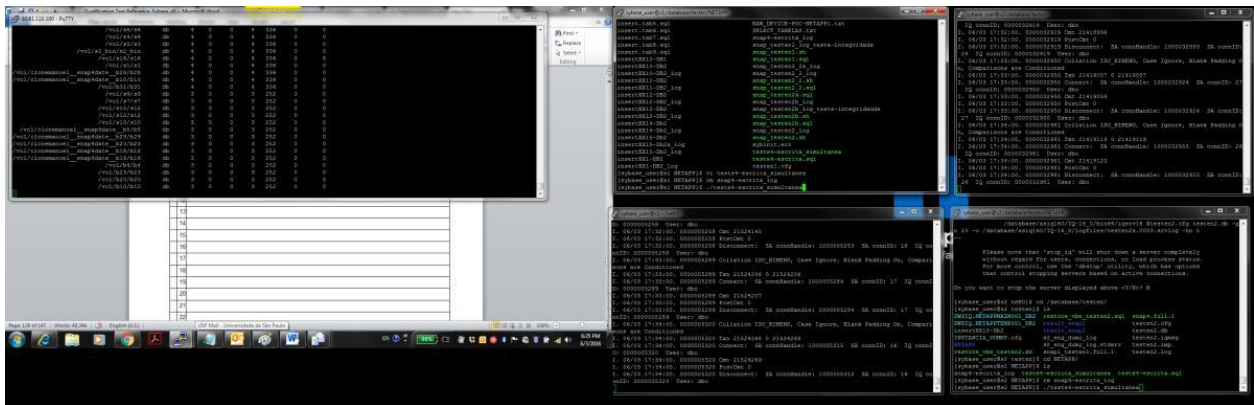
```
[sybase_user@s1 NETAPP]$ cat teste-escrita_simultanea
isql -Stesten2 -U<user> -P<password> -i teste-escrita.sql -o snap4-escrita_log
```

- Run the script on the S2 server:

```
[sybase_user@s2 NETAPP]$ ./teste-escrita_simultanea
```

```
[sybase_user@s2 NETAPP]$ cat teste-escrita_simultanea
isql -Stesten2s -U<user> -P<password> -i teste-escrita.sql -o snap4-escrita_log
```

3. At the same time, check the testen2.iqmsg file in both servers. On the storage, you can also run the statistics lun show command.



4. Check the size of Snapshot copies and clones from source LUNs.

```
superman::> volume snapshot show
```

Vserver	Volume	Snapshot	---Blocks---		
			Size	Total%	Used%
db	b1	snap4	10.17GB	1%	1%
		snapteste	7.11MB	0%	0%
		snapteste1	10.70MB	0%	0%
	b10				

	snap4	8.68GB	0%	1%
	snapteste	196KB	0%	0%
	snaptestel	144KB	0%	0%
...				
b16				
	snap4	8.56GB	0%	1%
	snapteste	188KB	0%	0%
	snaptestel	152KB	0%	0%
...				
clone_teste__snapteste__b1				
	snapteste	396.9MB	0%	0%
clone_teste__snapteste__b10				
	snapteste	37.85MB	0%	0%
...				
clone_teste__snapteste__b20				
	snapteste	37.61MB	0%	0%
...				

5. Check the writes to the testen2.iqmsg file on the S1 server.

```
[sybase_user@s1 neRO]$ hostname
s1
[sybase_user@s1 testen]$ tail -f testen2.iqmsg
...
I. 06/03 17:33:00. 0000005289 Cmt 21524207
I. 06/03 17:33:00. 0000005289 PostCmt 0
I. 06/03 17:33:00. 0000005289 Disconnect: SA connHandle: 1000005284 SA connID: 17 IQ connID:
0000005289 User: dbo
I. 06/03 17:34:00. 0000005320 Collation ISO_BINENG, Case Ignore, Blank Padding On, Comparisons
are Conditioned
I. 06/03 17:34:00. 0000005320 Txn 21524268 0 21524268
I. 06/03 17:34:00. 0000005320 Connect: SA connHandle: 1000005315 SA connID: 16 IQ connID:
0000005320 User: dbo
I. 06/03 17:34:00. 0000005320 Cmt 21524269
...
I. 06/03 17:34:48. 0000005345 [20896]: Insert for 'tb_data_manoel_teste_10' completed in 1
seconds. 166800 rows inserted.
I. 06/03 17:34:48. 0000005345 Cmt 21524321
I. 06/03 17:34:48. 0000005345 PostCmt 0
I. 06/03 17:34:48. 0000005345 Txn 21524322 0 21524322
I. 06/03 17:34:48. 0000005345 Cmt 21524323
I. 06/03 17:34:48. 0000005345 PostCmt 0
I. 06/03 17:34:48. 0000005345 Txn 21524324 0 21524324
I. 06/03 17:34:48. 0000005345 Cmt 21524325
I. 06/03 17:34:48. 0000005345 PostCmt 0
I. 06/03 17:34:48. 0000005345 Txn 21524326 0 21524326
I. 06/03 17:34:48. 0000005345 [20896]: Insert for 'tb_data_manoel_teste_10' completed in 0
seconds. 166800 rows inserted.
...
```

6. Check the writes to the testen2.iqmsg file on the S2 server.

```
[sybase_user@s2]$ hostname
S2
[sybase_user@s2 testen]$ tail -f testen2.iqmsg
...
I. 06/03 17:35:07. 0000033016 Connect: SA connHandle: 28 SA connID: 29 IQ connID:
0000033016 User: <user>
I. 06/03 17:35:07. 0000033016 Cmt 21419190
...
I. 06/03 17:35:07. 0000033016 [20896]: Insert for 'tb_data_manoel_teste_10' completed in 0
seconds. 166800 rows inserted.
I. 06/03 17:35:07. 0000033016 Cmt 21419194
I. 06/03 17:35:07. 0000033016 PostCmt 0
I. 06/03 17:35:07. 0000033016 Txn 21419195 0 21419195
I. 06/03 17:35:07. 0000033016 Cmt 21419196
I. 06/03 17:35:07. 0000033016 PostCmt 0
I. 06/03 17:35:07. 0000033016 Txn 21419197 0 21419197
I. 06/03 17:35:07. 0000033016 Cmt 21419198
I. 06/03 17:35:07. 0000033016 PostCmt 0
I. 06/03 17:35:07. 0000033016 Txn 21419199 0 21419199
```

```
I. 06/03 17:35:08. 0000033016 [20896]: Insert for 'tb_data_manoel_teste_10' completed in 1
seconds. 166800 rows inserted.
...
```

Test Results

It was possible create a Snapshot copy (snapteste) of a cloned database (testen2s based on the snap4 Snapshot copy) and to present it to the S2 server. After this, it was possible to execute scripts of read-write operations in both databases and servers without any interference between them.

8 Verification of Snapshot Copy Creation Time and Size

8.1 Test Objectives

- Check the size of each Snapshot copy.
- Check the time it takes to create a Snapshot copy.

8.2 Test Procedure

1. Check the initial size of Snapshot copies. The current size of the database is approximately 23.5TB.

```
superman::> volume snapshot show
```

```
Blocks---
Vserver  Volume      Snapshot      Size      Total% Used%
-----
db        b1
          snap1      100.3GB      5%      11%
          snap2      27.25MB      0%      0%
          snap3      33.73MB      0%      0%
          snap4      10.14GB      1%      1%
          b10
          snap1      96.27GB      5%      10%
          snap2      8.78MB      0%      0%
          snap3      7.76MB      0%      0%
          snap4      8.65GB      0%      1%
```

2. Delete the snap2 Snapshot copy and check the size of other Snapshot copies created. At this point, there are no writes to the database testen2.

```
[root@s1 neRO]# ./neRO --action snap-delete --vserver db -conf
/home/sybase_user/neRO/conf/base2.conf --snap snap2
output superman::> volume snapshot show
```

```
Blocks---
Vserver  Volume      Snapshot      Size      Total% Used%
-----
db        b1
          snap1      100.3GB      5%      11%
          snap3      33.73MB      0%      0%
          snap4      10.14GB      1%      1%
          b10
          snap1      96.27GB      5%      10%
          snap3      7.76MB      0%      0%
          snap4      8.65GB      0%      1%
```

3. Delete the Snapshot copies snap1 and snap3 and check the size of snap4. The size of the Snapshot copy increases because it references metadata from other Snapshot copies.

Initially it was 10.14GB. After deletion, it is 10.16GB from volume b1, as shown in the following example.

```
[root@s1 neRO]# ./neRO --action snap-delete --vserver db -conf
/home/sybase_user/neRO/conf/base2.conf --snap snap1
```

```
[root@s1 neRO]# ./neRO --action snap-delete --vserver db -conf
/home/sybase_user/neRO/conf/base2.conf --snap snap3
superman::> volume snapshot show
```

```
Blocks---
Vserver  Volume      Snapshot
-----  -
db        b1
          snap4      10.16GB      1%      1%
          b10
          snap4      8.67GB       0%      1%
```

4. Execute the following two scripts to populate the testen2 database (25TB and 28.5TB). During the execution of the scripts, create Snapshot copies to check the size and execution time.

Note: At this moment, the database size is 23.5TB:

```
[sybase_user@s1 NETAPP]$ isql -Stesten2 -U<user> -P<password>
1> select * from sp_iqstatus() where name like '%Main IQ Block%'
2> go
Name
-----
Value
-----
Main IQ Blocks Used:
771571450 of 939197518, 82%=23546Gb, Max Block#: 1050129582
(1 row affected)
exit
```

```
[sybase_user@s1 NETAPP]$ cat insertXX15-Db2 | more
select getdate()
go
insert into manoeljr.tb_data_manoel_teste_1
select * from manoeljr.tb_data_manoel_teste_xx
go
commit
go
insert into manoeljr.tb_data_manoel_teste_xx
select * from manoeljr.tb_data_manoel_teste_1
go
commit
...

[sybase_user@s1 NETAPP]$ cat insertXX14-Db2 | more
select getdate()
go
insert into manoeljr.tb_data_manoel_teste_4a
select * from manoeljr.tb_data_manoel_teste_3a
go
commit
...
```

Table is a record of the values of reads and writes to the testen2 database with the following sizes: 23.5TB, 25TB, and 28.5TB.

Table 4) Snapshot: time and size, verified with writes.

Database Size (TB)	Time (Seconds)	Size of First Backup (GB)	Size of Second Backup (GB)	Size of Third Backup (GB)
23.5	15	12.7	2.86	0.99

Database Size (TB)	Time (Seconds)	Size of First Backup (GB)	Size of Second Backup (GB)	Size of Third Backup (GB)
25	17	15	16.9	93
28.5	17	7.42	4.12	0.53

Table 5) Snapshot: time and size, verified without writes.

Database Size (TB)	Time (Seconds)	Size of First Backup (MB)	Size of Second Backup (MB)	Size of Third Backup (MB)
23.5	20	916.32	16.27	8.67
25	20	112.22	17.76	11.45
28.5	20	1060	24.6	9.09

9 Compression and Deduplication

In addition to Snapshot technology, NetApp controllers have two core features: NetApp deduplication and NetApp data compression.

These technologies can work together or independently to achieve optimal savings. NetApp deduplication is a process that can be scheduled to run when it is most convenient, whereas NetApp data compression can be run either as an inline process as data is written to disk or as a scheduled process. When the two are enabled on the same volume, the data is first compressed and then deduplicated. Deduplication removes duplicate compressed or uncompressed blocks in a data volume. Although compression and deduplication work well together, it should be noted that the savings are not necessarily the sum of the savings when each is run individually on a dataset.

9.1 Test Objective

Test the efficiency of inline compression and deduplication.

NetApp data compression can be configured as an inline operation. In this way, as data is sent to the storage system, it is compressed in memory before being written to the disk. The advantage of this implementation is that it can reduce the amount of write I/O. This proof of concept uses inline compression and postprocess deduplication.

9.2 Test Procedure

1. Configure the efficiency policies related to inline compression and deduplication. Verify the space savings.

Goal	Recommendation
Minimize Snapshot space	Inline compression: Inline compression will minimize the amount of space used by Snapshot copies.
Minimize qtree SnapMirror or SnapVault destinations' disk space usage	Inline compression: Inline compression provides immediate savings with minimal impact on backup windows. Further, it takes up less space in the snapshot reserve.
Minimize disk I/O	Inline compression: Inline compression can reduce the number of new blocks written to disk. Postprocess compression requires an initial write of all the uncompressed data blocks followed by a read of the uncompressed data and a new write of compressed blocks.

Goal	Recommendation
Do not affect performance when writing new data to disk	Postprocess compression: Postprocess compression writes the new data to disk as uncompressed without any impact on the initial write performance. You can then schedule when compression occurs to gain the savings.
Minimize effect on CPU during peak hours	Postprocess compression: Postprocess compression allows you to schedule when compression occurs, thus minimizing the impact of compressing new data during peak hours.

Note: It is important to determine that you have sufficient resources available on your system before considering enabling inline compression including during peak hours. NetApp highly recommends that the performance impact be fully tested and understood before you implement in production.

```
superman::> volume efficiency policy create -vserver db -policy efficiency_policy_POC_CUSTOMER -
schedule 5min -enabled true -qos-policy background
superman::> volume efficiency policy show
```

Vserver	Policy	Job	Duration	QoS	Policy	Enabled	Comment
		Name	Schedule	(Hours)			
db	default	daily	-	best_effort	true		Default policy
db	efficiency_policy_POC_CUSTOMER	5min	-	background	true	-	
db	inline-only	-	-	-	-		Inline-Only policy

3 entries were displayed.

2. Apply the efficiency policy to all volumes.

```
superman::> volume efficiency on -vserver db -volume b2, b3, b4, b5, b6, b8,b9 ,b10, b11,b12,
b13,b14, b15, b16, b17,b18,b19,b20,b21,b22,b23,b24,b25,b26,b2
7,b28,b29,b30,b31,s7,s8,s9,s10,s11,s12,s14,s15,s16

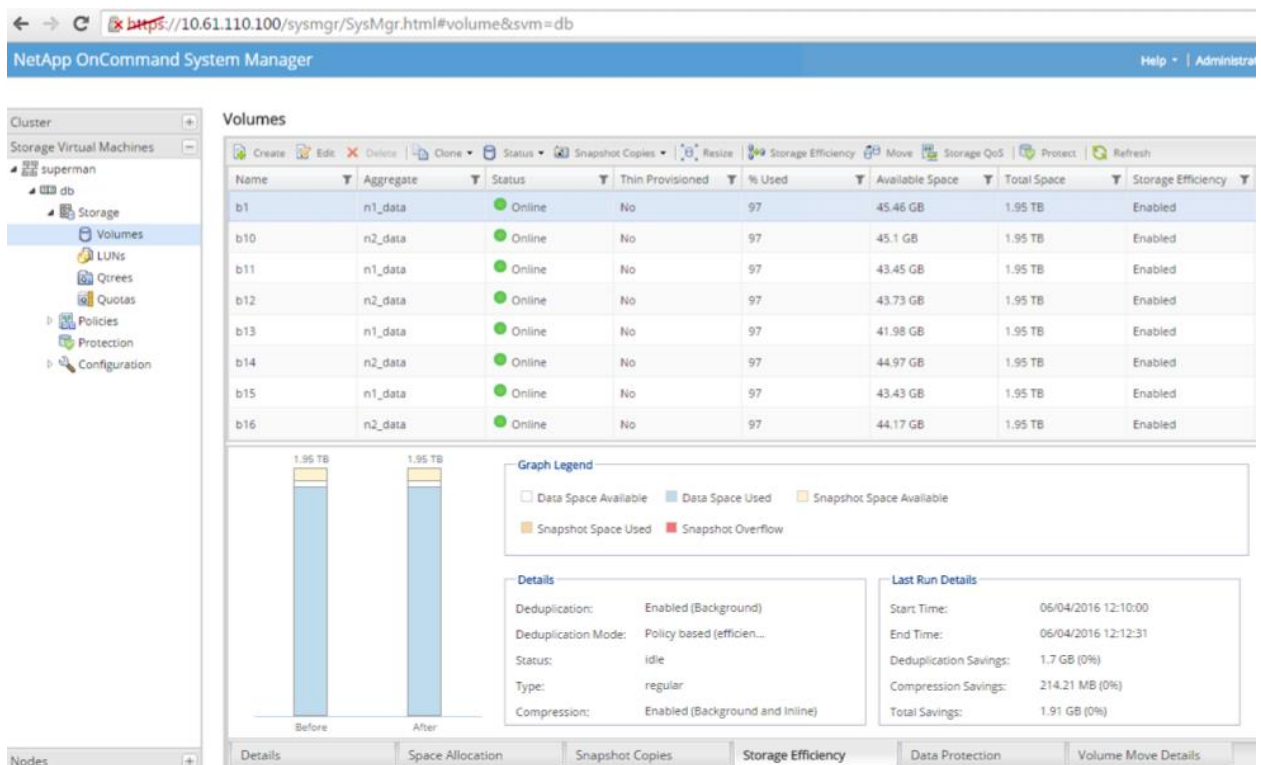
superman::> volume efficiency modify -vserver db -volume b3, b4, b5, b6, b8,b9, b10, b11,b12,
b13, b14, b15,b16, b17,b18, b19,b20,b21, b22,b23,b24,b25,b26,b27,b 28,b29,b30,b31,
s7,s8,s9,s10,s11, s12,s14, s15,s16 -policy efficiency_policy_POC_CUSTOMER

superman::> volume efficiency modify -vserver db -volume b3, b4, b5, b6, b8,b9, b10, b11,b12,
b13, b14, b15,b16, b17,b18, b19,b20,b21, b22,b23,b24,b25,b26,b27,b 28,b29,b30,b31,
s7,s8,s9,s10,s11, s12,s14, s15,s16 -compression true -inline-compression true
```

3. Verify the space saved.

```
superman::> volume show -vserver db -volume b1, b2, b3, b4, b5, b6, b8, b9, b10, b11,b12, b13,
b14, b15, b16, b17, b18, b19, b20, b21, b22, b23, b24, b25, b26, b27, b28, b29, b30, b31, s7, s8,
s9, s10, s11, s12, s14, s15, s16 -fields sis-space-saved
```

vserver	volume	sis-space-saved
db	b1	658.3MB
db	b10	335.9MB
db	b11	384.9MB
db	b14	298.8MB
db	b15	294.9MB
db	b16	299.6MB
db	b17	287MB
...		
db	b9	342.9MB
db	s10	1.02MB
db	s11	956KB
db	s12	1.02MB
db	s14	1.02MB
db	s15	1012KB
db	s16	1012KB
...		



- The actual efficiency ratio can be evaluated when the database reaches its maximum capacity (28.5TB).

Space saving of 62.32GB is observed in each 1TB LUN, and 3.4MB of space was saved in each 250GB LUN.

```
[sybase_user@sl ~]$ isql -Stesten2 -U<user> -P<password>
1> select * from sp_iqstatus() where name like '%Main IQ Block%'
2> go
Name
Value
-----
Main IQ Blocks Used:
935049885 of 939197518, 99%=28535Gb, Max Block#: 1050133377

superman::> date
Node Date Time zone
-----
n1 Sun Jun 05 12:32:28 2016 America/New_York
n2 Sun Jun 05 12:32:28 2016 America/New_York
2 entries were displayed.
output superman::> volume show -vserver db -volume b1, b2, b3, b4, b5, b6, b8, b9, b10, b11,b12,
b13, b14, b15, b16, b17, b18, b19, b20, b21, b22, b23, b24, b25, b26, b27, b28, b29, b30, b31,
s7, s8, s9, s10, s11, s12, s14, s15, s16 -fields sis-space-saved

vserver volume sis-space-saved
-----
db b1 62.32GB
db b10 61.53GB
db b11 61.54GB
...
db b9 61.54GB
db s10 3.44MB
db s11 3.49MB
...
db s16 3.51MB
...
```

Table 6) Space saved: compression and deduplication.

Savings in Testen2 Database (28.5TB)			
Volume	Saved Space (GB)	Deduplication (GB)	Compression (MB)
b1	62.51	61.96	563.54
b10	61.53	61.16	379.14
b11	61.54	61.16	389.11
...	...	...	...
b9	61.54	61.17	381.2
s10	0.0024	0.00238	0.028
...	...	...	...
s9	0.00239	0.00234	0.316
Total	1.59TB	1.58TB	10GB

References

The following references were used in this TR:

- SnapDrive for UNIX
<http://www.netapp.com/us/products/management-software/snapdrive-unix.aspx>
- NetApp OnCommand Unified Manager (example)
<http://www.netapp.com/us/products/management-software/oncommand/unified-manager.aspx>
- NetApp ONTAP: Storage Efficiency
<http://www.netapp.com/us/products/platform-os/ontap/storage-efficiency.aspx>
- SAP IQ Publishing
<http://www.sap.com/documents/2016/10/52c77ebf-8f7c-0010-82c7-eda71af511fa.html>
- Clustered Data ONTAP 8.3: Data Protection Guide
https://library.netapp.com/ecm/ecm_download_file/ECMP1610205
- Leveraging NetApp infrastructure for large scale SAP IQ backups via virtual backup
<https://www.sap.com/documents/2016/10/52c77ebf-8f7c-0010-82c7-eda71af511fa.html>

Version History

Version	Date	Document Version History
Version 1.0	March 2017	Initial version

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

Copyright Information

Copyright © 1994–2017 NetApp, Inc. All rights reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark Information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.

TR-4583-0417