SPC BENCHMARK 1™
FULL DISCLOSURE REPORT

HUAWEI TECHNOLOGIES CO., LTD.
HUAWEI OCEANSTOR™ 6800 V3

SPC-1 V1.14

Submitted for Review: November 21, 2014
Submission Identifier: A00149

First Edition – November 2014

THE INFORMATION CONTAINED IN THIS DOCUMENT IS DISTRIBUTED ON AN AS IS BASIS WITHOUT ANY WARRANTY EITHER EXPRESS OR IMPLIED. The use of this information or the implementation of any of these techniques is the customer's responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item has been reviewed by Huawei Technologies Co., Ltd. for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environment do so at their own risk.

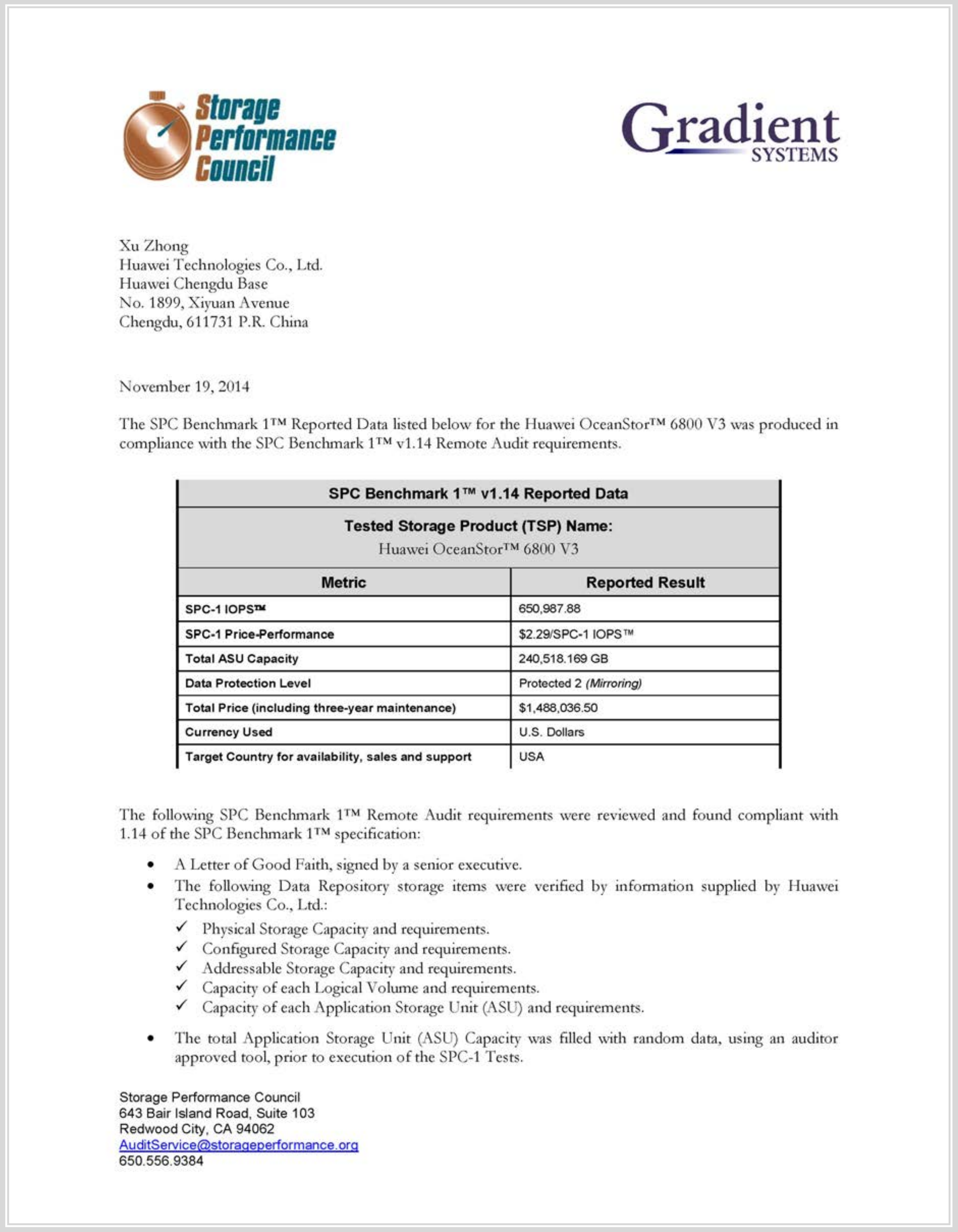
This publication was produced in the United States. Huawei Technologies Co., Ltd. may not offer the products, services, or features discussed in this document in other countries, and the information is subject to change with notice. Consult your local Huawei Technologies Co., Ltd. representative for information on products and services available in your area.

Trademarks

SPC Benchmark-1, SPC-1, SPC-1 IOPS, SPC-1 LRT and SPC-1 Price-Performance are trademarks of the Storage Performance Council. Huawei, the Huawei logo and OceanStor are trademarks or registered trademarks of Huawei Technologies Co., Ltd. in the United States and other countries. All other brands, trademarks, and product names are the property of their respective owners.

Table of Contents

AUDIT CERTIFICATION



Xu Zhong
Huawei Technologies Co., Ltd.
Huawei Chengdu Base
No. 1899, Xiyuan Avenue
Chengdu, 611731 P.R. China

November 19, 2014

The SPC Benchmark 1™ Reported Data listed below for the Huawei OceanStor™ 6800 V3 was produced in compliance with the SPC Benchmark 1™ v1.14 Remote Audit requirements.

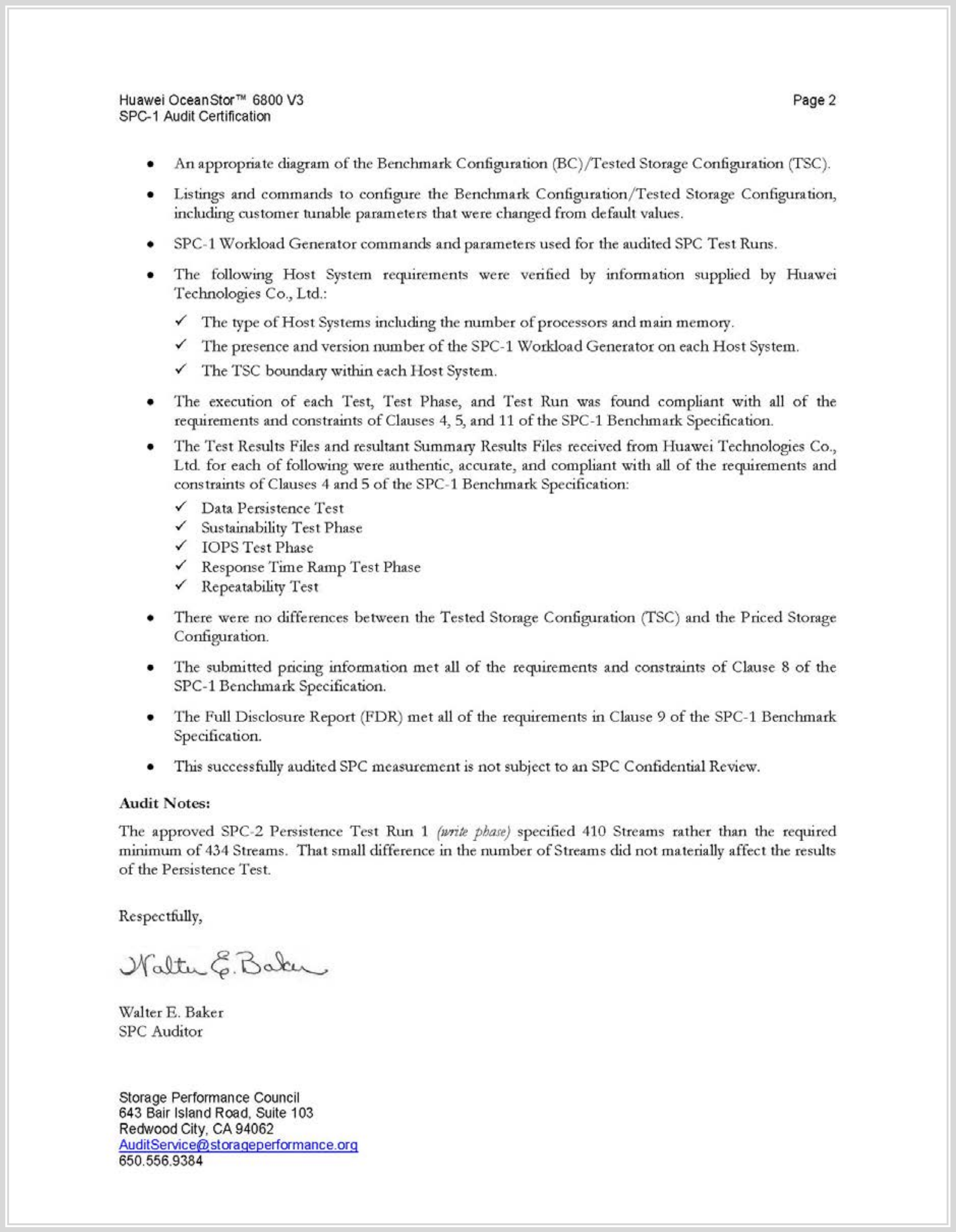
SPC Benchmark 1™ v1.14 Reported Data	
Tested Storage Product (TSP) Name: Huawei OceanStor™ 6800 V3	
Metric	**Reported Result**
SPC-1 IOPS™	650,987.88
SPC-1 Price-Performance	$2.29/SPC-1 IOPS™
Total ASU Capacity	240,518.169 GB
Data Protection Level	Protected 2 *(Mirroring)*
Total Price (including three-year maintenance)	$1,488,036.50
Currency Used	U.S. Dollars
Target Country for availability, sales and support	USA

The following SPC Benchmark 1™ Remote Audit requirements were reviewed and found compliant with 1.14 of the SPC Benchmark 1™ specification:

- A Letter of Good Faith, signed by a senior executive.
- The following Data Repository storage items were verified by information supplied by Huawei Technologies Co., Ltd.:
  - ✓ Physical Storage Capacity and requirements.
  - ✓ Configured Storage Capacity and requirements.
  - ✓ Addressable Storage Capacity and requirements.
  - ✓ Capacity of each Logical Volume and requirements.
  - ✓ Capacity of each Application Storage Unit (ASU) and requirements.
- The total Application Storage Unit (ASU) Capacity was filled with random data, using an auditor approved tool, prior to execution of the SPC-1 Tests.

Storage Performance Council
643 Bair Island Road, Suite 103
Redwood City, CA 94062
AuditService@storageperformance.org
650.556.9384

AUDIT CERTIFICATION *(CONT.)*



- An appropriate diagram of the Benchmark Configuration (BC)/Tested Storage Configuration (TSC).
- Listings and commands to configure the Benchmark Configuration/Tested Storage Configuration, including customer tunable parameters that were changed from default values.
- SPC-1 Workload Generator commands and parameters used for the audited SPC Test Runs.
- The following Host System requirements were verified by information supplied by Huawei Technologies Co., Ltd.:
  - ✓ The type of Host Systems including the number of processors and main memory.
  - ✓ The presence and version number of the SPC-1 Workload Generator on each Host System.
  - ✓ The TSC boundary within each Host System.
- The execution of each Test, Test Phase, and Test Run was found compliant with all of the requirements and constraints of Clauses 4, 5, and 11 of the SPC-1 Benchmark Specification.
- The Test Results Files and resultant Summary Results Files received from Huawei Technologies Co., Ltd. for each of following were authentic, accurate, and compliant with all of the requirements and constraints of Clauses 4 and 5 of the SPC-1 Benchmark Specification:
  - ✓ Data Persistence Test
  - ✓ Sustainability Test Phase
  - ✓ IOPS Test Phase
  - ✓ Response Time Ramp Test Phase
  - ✓ Repeatability Test
- There were no differences between the Tested Storage Configuration (TSC) and the Priced Storage Configuration.
- The submitted pricing information met all of the requirements and constraints of Clause 8 of the SPC-1 Benchmark Specification.
- The Full Disclosure Report (FDR) met all of the requirements in Clause 9 of the SPC-1 Benchmark Specification.
- This successfully audited SPC measurement is not subject to an SPC Confidential Review.

Audit Notes:

The approved SPC-2 Persistence Test Run 1 *(write phase)* specified 410 Streams rather than the required minimum of 434 Streams. That small difference in the number of Streams did not materially affect the results of the Persistence Test.

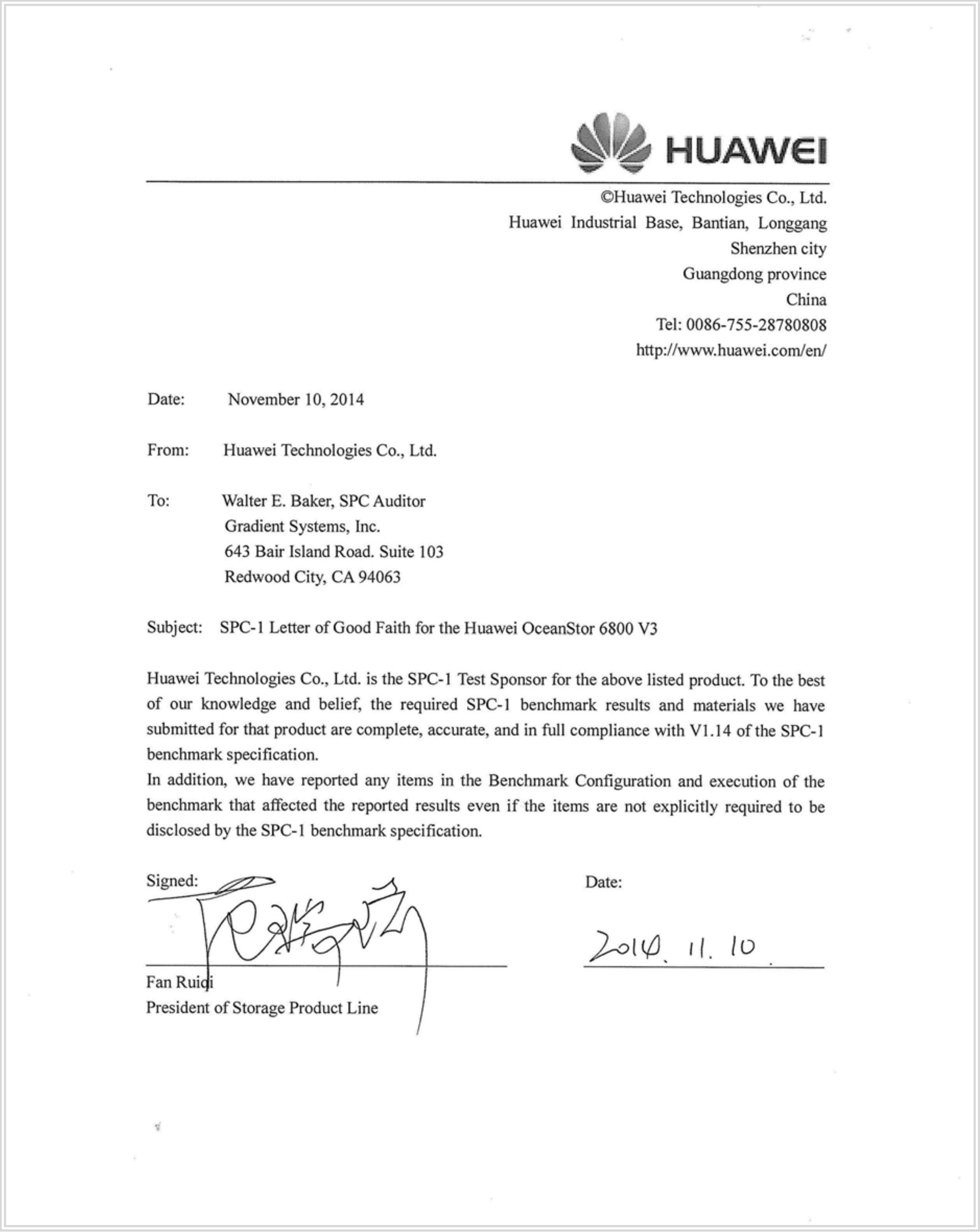
Respectfully,

Walter E. Baker

Walter E. Baker
SPC Auditor

Storage Performance Council
643 Bair Island Road, Suite 103
Redwood City, CA 94062
AuditService@storageperformance.org
650.556.9384

LETTER OF GOOD FAITH



HUAWEI

©Huawei Technologies Co., Ltd.
Huawei Industrial Base, Bantian, Longgang
Shenzhen city
Guangdong province
China
Tel: 0086-755-28780808
http://www.huawei.com/en/

Date: November 10, 2014

From: Huawei Technologies Co., Ltd.

To: Walter E. Baker, SPC Auditor
Gradient Systems, Inc.
643 Bair Island Road. Suite 103
Redwood City, CA 94063

Subject: SPC-1 Letter of Good Faith for the Huawei OceanStor 6800 V3

Huawei Technologies Co., Ltd. is the SPC-1 Test Sponsor for the above listed product. To the best of our knowledge and belief, the required SPC-1 benchmark results and materials we have submitted for that product are complete, accurate, and in full compliance with V1.14 of the SPC-1 benchmark specification.

In addition, we have reported any items in the Benchmark Configuration and execution of the benchmark that affected the reported results even if the items are not explicitly required to be disclosed by the SPC-1 benchmark specification.

Signed: Date: 2014. 11. 10.

Fan Ruiqi
President of Storage Product Line

EXECUTIVE SUMMARY

Test Sponsor and Contact Information

Test Sponsor and Contact Information	
Test Sponsor Primary Contact	Huawei Technologies Co., Ltd. – http://www.huawei.com/en/ Xu Zhong – xuzhong@huawei.com Huawei Chengdu Base No. 1899, Xiyuan Avenue Chengdu, 611731 P.R. China Phone: 86 28 65281927 FAX: 86 28 62282516
Test Sponsor Alternate Contact	Huawei Technologies Co., Ltd. – http://www.huawei.com/en/ Li Huan – tomas.l@huawei.com Huawei Chengdu Base No. 1899, Xiyuan Avenue Chengdu, 611731 P.R. China Phone: 86 28 65281927 FAX: 86 28 62282516
Test Sponsor Alternate Contact	Huawei Technologies Co., Ltd. – http://www.huawei.com/en/ Li Xingxin – lixingxin@huawei.com Huawei Chengdu Base No. 1899, Xiyuan Avenue Chengdu, 611731 P.R. China Phone: 86 28 65281926 FAX: 86 28 62282516
Auditor	Storage Performance Council – http://www.storageperformance.org Walter E. Baker – AuditService@StoragePerformance.org 643 Bair Island Road, Suite 103 Redwood City, CA 94063 Phone: (650) 556-9384 FAX: (650) 556-9385

Revision Information and Key Dates

Revision Information and Key Dates	
SPC-1 Specification revision number	V1.14
SPC-1 Workload Generator revision number	V2.3.0
Date Results were first used publicly	November 21, 2014
Date the FDR was submitted to the SPC	November 21, 2014
Date the Priced Storage Configuration is available for shipment to customers	currently available
Date the TSC completed audit certification	November 19, 2014

Tested Storage Product (TSP) Description

Huawei OceanStor™ 6800 V3 high-end storage system is the next-generation unified storage product specifically designed for enterprise-class applications. Leveraging a storage operating system, OceanStor OS, built on a cloud-oriented architecture, a powerful new hardware platform, and a suite of intelligent management software, the V3 high-end storage system delivers industry-leading functionality, performance, efficiency, reliability, and ease-of-use. It provides data storage for applications such as large-database Online Transaction Processing (OLTP)/Online Analytical Processing (OLAP), file sharing, and cloud computing, which can be widely applied to industries ranging from government, finance, telecommunications, energy, media and entertainment (M&E). Meanwhile, the V3 high-end storage system can provide a wide range of efficient and flexible backup and disaster recovery solutions to ensure business continuity and data security, delivering excellent storage services.

OceanStor OS, the Huawei OceanStor storage operating system, enables Huawei storage products evolve to the future cloud architecture and deliver the core business platform. It supports all OceanStor Storage arrays, specifically, for managing the underlying infrastructure, the physical space and logical space. OceanStor OS delivers intelligent and convergent services and multiple SLAs to the application scenarios, including SAN and NAS convergence, all-level storage convergence, performance and capacity convergence, primary and backup storage convergence, and heterogeneous storage convergence. OceanStor OS helps customers evolve their traditional storage to cloud services in the future.

Summary of Results

SPC-1 Reported Data	
Tested Storage Product (TSP) Name: Huawei OceanStor™ 6800 V3	
Metric	**Reported Result**
SPC-1 IOPS™	650,987.88
SPC-1 Price-Performance™	$2.29/SPC-1 IOPS™
Total ASU Capacity	240,518.169 GB
Data Protection Level	Protected 2 *(mirroring)*
Total Price	$1,488,036.50
Currency Used	U.S. Dollars
Target Country for availability, sales and support	USA

SPC-1 IOPS™ represents the maximum I/O Request Throughput at the 100% load point.

SPC-1 Price-Performance™ is the ratio of **Total Price** to **SPC-1 IOPS™**.

Total ASU (Application Storage Unit) **Capacity** represents the total storage capacity available to be read and written in the course of executing the SPC-1 benchmark.

A **Data Protection Level** of **Protected 2** using ***Mirroring*** configures two or more identical copies of user data.

> ***Protected 2:*** *The single point of failure of any* ***component*** *in the configuration will not result in permanent loss of access to or integrity of the SPC-1 Data Repository.*

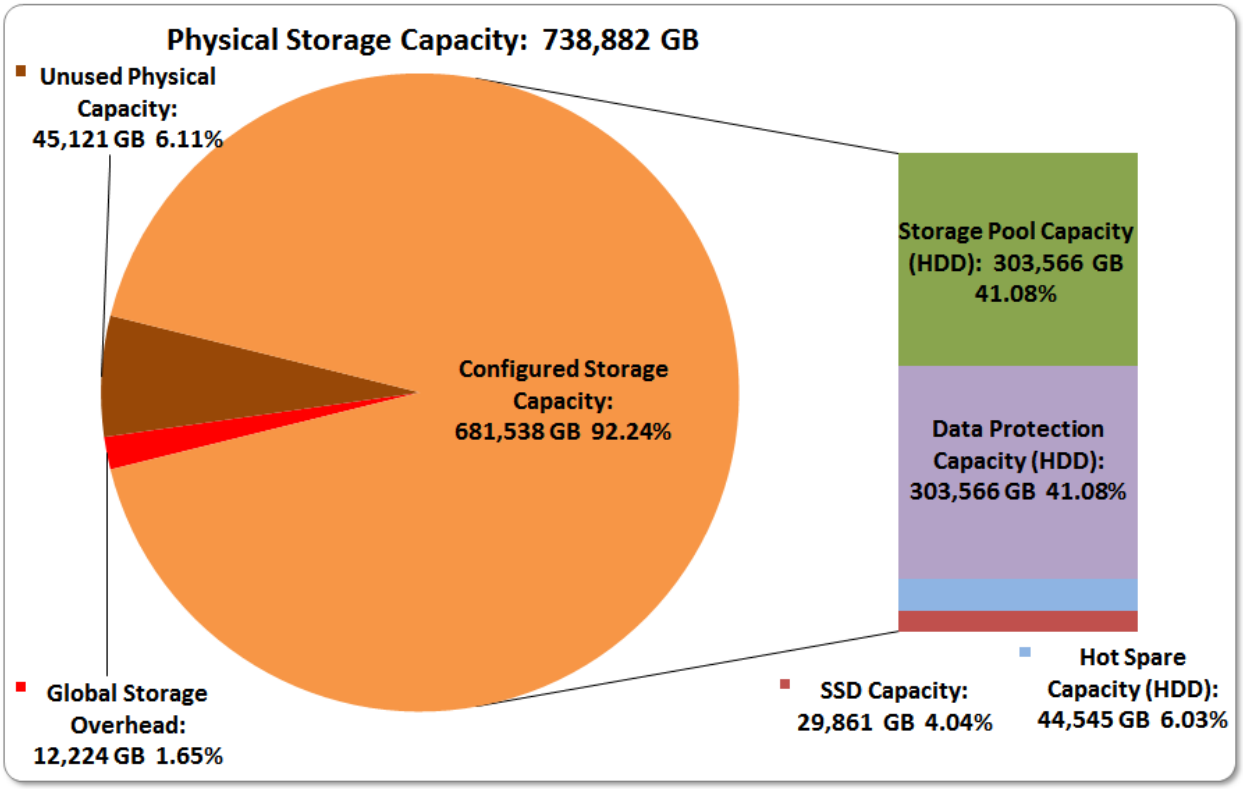
Total Price includes the cost of the Priced Storage Configuration plus three years of hardware maintenance and software support as detailed on page 17.

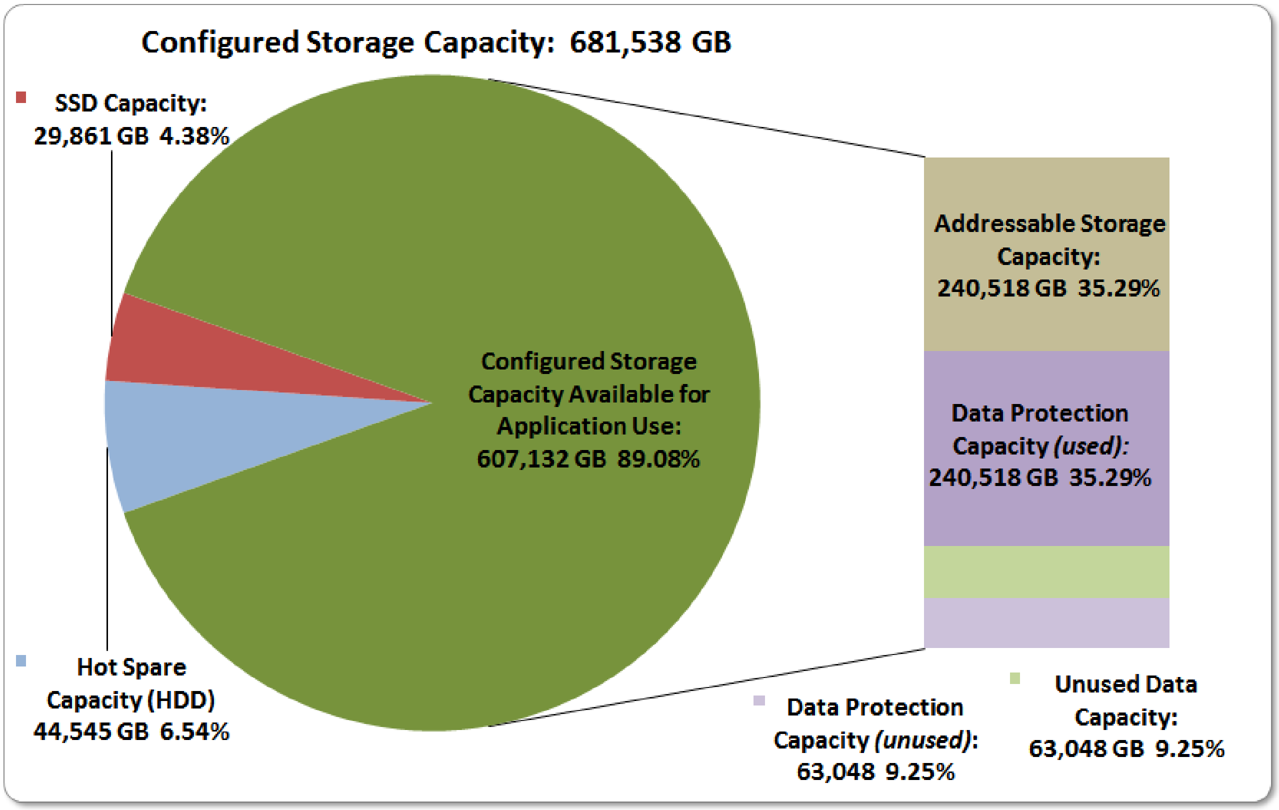
Currency Used is formal name for the currency used in calculating the **Total Price** and **SPC-1 Price-Performance™**. That currency may be the local currency of the **Target Country** or the currency of a difference country *(non-local currency).*

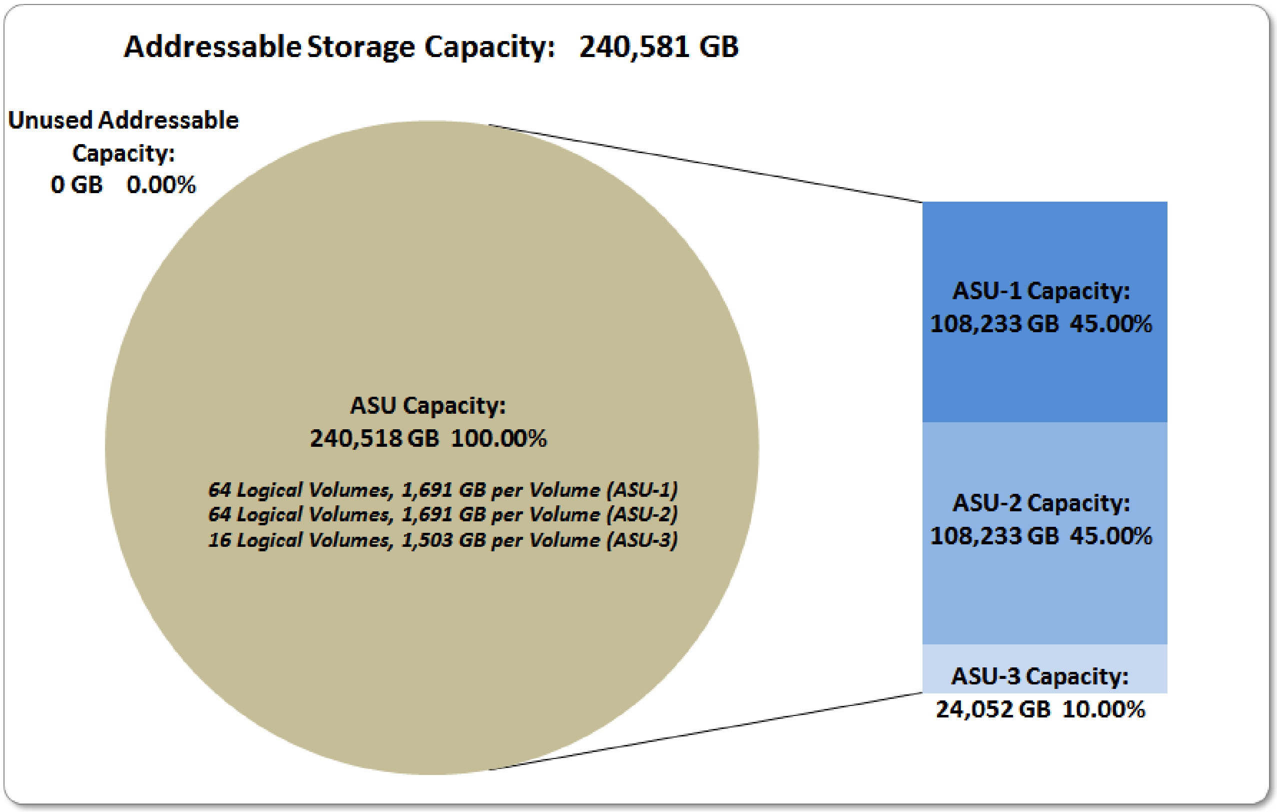
The **Target Country** is the country in which the Priced Storage Configuration is available for sale and in which the required hardware maintenance and software support is provided either directly from the Test Sponsor or indirectly via a third-party supplier.

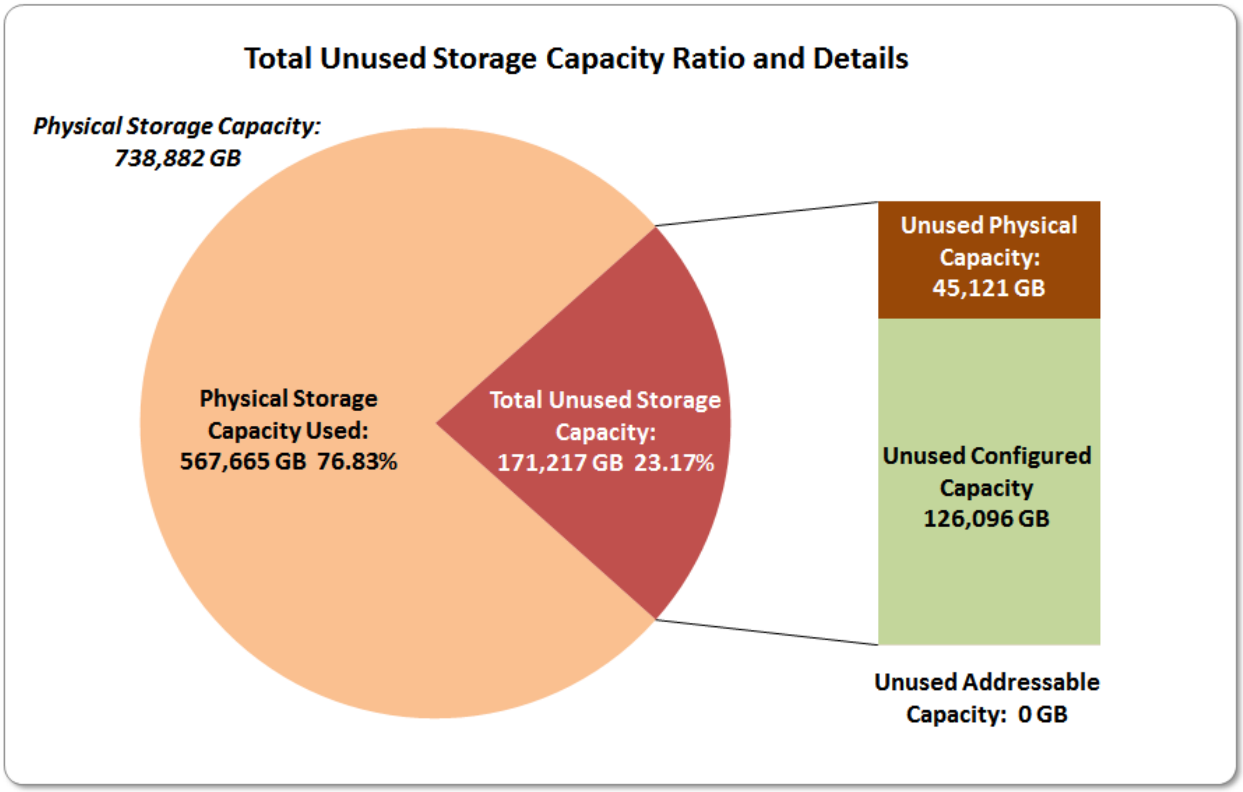
Storage Capacities, Relationships, and Utilization

The following four charts and table document the various storage capacities, used in this benchmark, and their relationships, as well as the storage utilization values required to be reported.




Configured Storage Capacity: 681,538 GB
SSD Capacity: 29,861 GB 4.38%
Configured Storage Capacity Available for Application Use: 607,132 GB 89.08%
Hot Spare Capacity (HDD) 44,545 GB 6.54%
Addressable Storage Capacity: 240,518 GB 35.29%
Data Protection Capacity *(used)*: 240,518 GB 35.29%
Data Protection Capacity *(unused)*: 63,048 9.25%
Unused Data Capacity: 63,048 GB 9.25%


Addressable Storage Capacity: 240,581 GB
Unused Addressable Capacity: 0 GB 0.00%
ASU Capacity: 240,518 GB 100.00%
64 Logical Volumes, 1,691 GB per Volume (ASU-1)
64 Logical Volumes, 1,691 GB per Volume (ASU-2)
16 Logical Volumes, 1,503 GB per Volume (ASU-3)
ASU-1 Capacity: 108,233 GB 45.00%
ASU-2 Capacity: 108,233 GB 45.00%
ASU-3 Capacity: 24,052 GB 10.00%



SPC-1 Storage Capacity Utilization	
Application Utilization	32.55%
Protected Application Utilization	66.92%
Unused Storage Ratio	23.17%

Application Utilization: Total ASU Capacity *(240,518.169 GB)* divided by Physical Storage Capacity *(738,882.470 GB)*.

Protected Application Utilization: Total ASU Capacity *(240,518.169 GB)* plus total Data Protection Capacity *(316,983.619 GB)* minus unused Data Protection Capacity *(63,047.972 GB)* divided by Physical Storage Capacity *(738,882.470 GB)*.

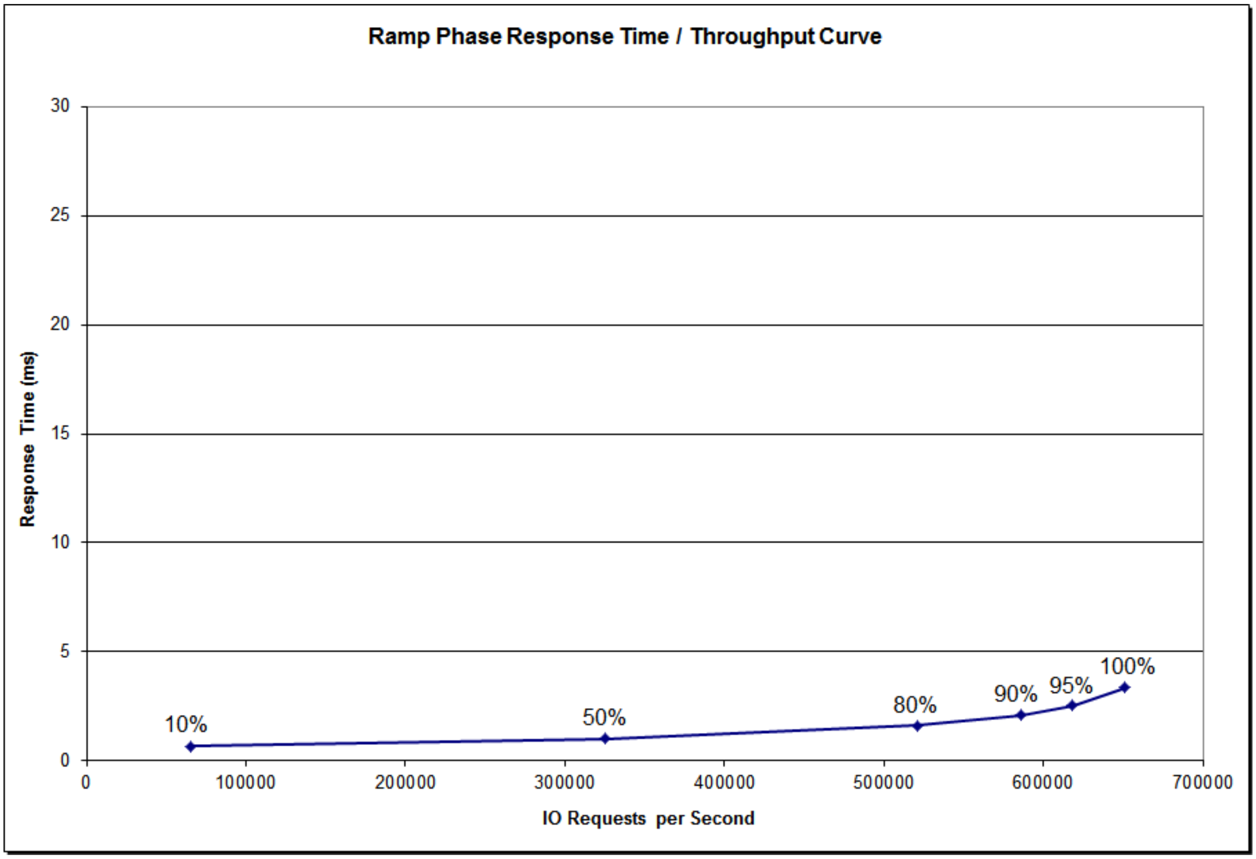
Unused Storage Ratio: Total Unused Capacity *(171,217.146 GB)* divided by Physical Storage Capacity *(738,882.470 GB)* and may not exceed 45%.

Detailed information for the various storage capacities and utilizations is available on pages 26-27.

Response Time – Throughput Curve

The Response Time-Throughput Curve illustrates the Average Response Time (milliseconds) and I/O Request Throughput at 100%, 95%, 90%, 80%, 50%, and 10% of the workload level used to generate the SPC-1 IOPS™ metric.

The Average Response Time measured at any of the above load points cannot exceed 30 milliseconds or the benchmark measurement is invalid.



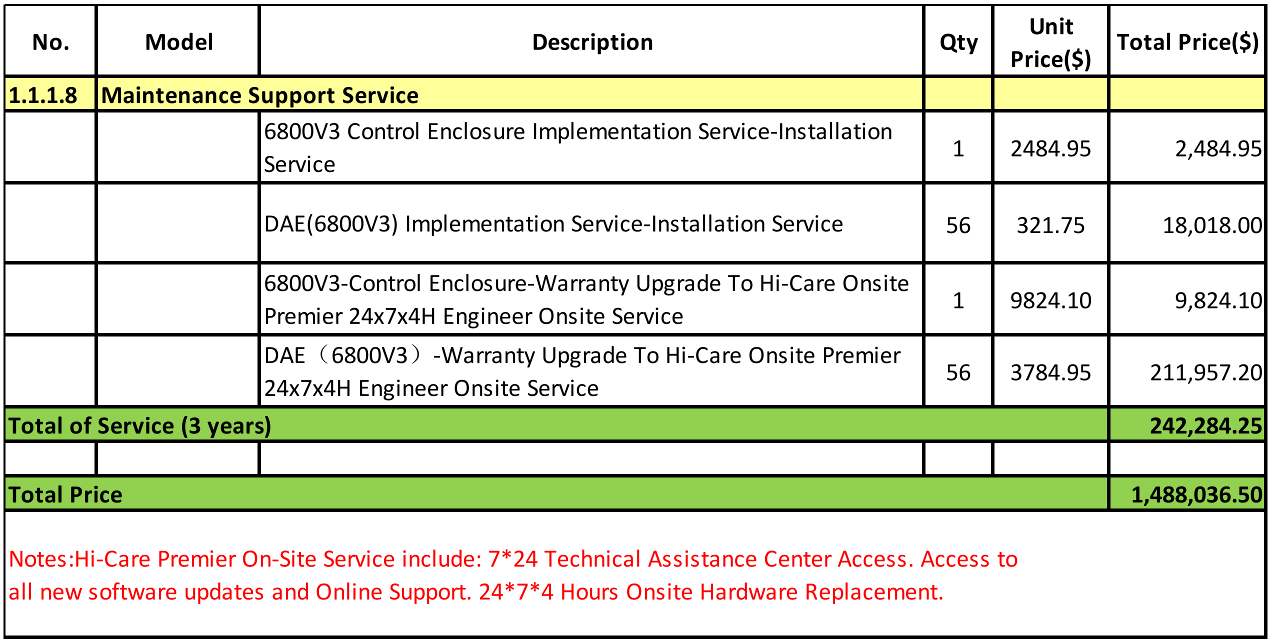
Response Time – Throughput Data

	10% Load	50% Load	80% Load	90% Load	95% Load	100% Load
I/O Request Throughput	65,098.38	325,492.99	520,705.64	585,883.12	618,437.41	650,987.88
Average Response Time (ms):						
All ASUs	0.68	1.01	1.63	2.08	2.54	3.36
ASU-1	0.61	0.86	1.35	1.70	2.10	2.78
ASU-2	1.55	2.49	3.82	4.43	4.87	5.65
ASU-3	0.45	0.67	1.26	1.85	2.46	3.61
Reads	1.04	1.53	2.21	2.46	2.71	3.04
Writes	0.45	0.67	1.25	1.83	2.43	3.57

Priced Storage Configuration Pricing

No.	Model	Description	Qty	Unit Price($)	Total Price($)
1	Phase				
1.1	Location				
1.1.1	6800 V3 Storage System				
1.1.1.1	Control Module				
	6800V3-512G-AC	6800 V3(6U,Dual Ctrl,AC,512GB,HW Storage System Software,SPE72C0600)	1	41091.00	41,091.00
	6800V3-256G-CTL	Controller Module(2*Intel 6 Cores,256GB Cache)	2	8903.00	17,806.00
1.1.1.2	Disk Enclosure				
	DAE22525U2-1-AC	Disk Enclosure(2U,2.5",AC,SAS Expanding Module)	56	2054.50	115,052.00
1.1.1.3	Hard Disk Drives				
	SAS600-10K-2-V3	600GB 10K RPM SAS Disk Unit(2.5")	1184	457.75	541,976.00
	SLC200-2-02	200GB HSSD SLC SAS Disk Unit(2.5")	160	2888.00	462,080.00
1.1.1.4	IO Interface				
	LPU4FC8V3	4*8Gbps Fibre Channel I/O modules(Total 4 ports)	12	1983.75	23,805.00
	LPU4S12V3	4*12Gbps SAS I/O module(Total 4 ports,MiniSAS HD)	12	741.25	8,895.00
1.1.1.5	Accessory				
	SS-OP-D-LC-M-3	Patchcord,DLC/PC-DLC/PC,Multimode,2mm Parallel,3m	24	11.00	264.00
	HS-SAS-3-01	High Speed Cable,Mini SAS HD Cable,3m,(SFF 8644	1	67.00	67.00
	HS-SAS-5-01	High Speed Cable,Mini SAS HD Cable,5m,(SFF 8644 Plug),(26AWG*4P*2B(S)),(SFF 8644 Plug),Indoor use	18	91.00	1,638.00
	RACK-46U-AC	N610E-22 46U Common AC Storage Rack(include 2 AC power distribution panels)	3	1712.00	5,136.00
1.1.1.6	HBA				
	N8GHBA000	QLOGIC QLE2562 HBA Card,PCIE,8Gbps DualPort ,Fiber Channel Multimode LC Optic Interface,English Manual, No Drive CD	12	1000.00	12,000.00
1.1.1.7	Storage Software				
	LIC-6800V3-BS	Basic Software License,Include Device Management	1	3281.50	3,281.50
	LIC-6800V3-TIER	SmartTier License	1	11944.50	11,944.50
	LIC-6800V3-PATH	OceanStor HW UltraPath Software License	1	716.25	716.25
Total of Product					1,245,752.25

Priced Storage Configuration Pricing *(continued)*



No.	Model	Description	Qty	Unit Price($)	Total Price($)
1.1.1.8	**Maintenance Support Service**				
		6800V3 Control Enclosure Implementation Service-Installation Service	1	2484.95	2,484.95
		DAE(6800V3) Implementation Service-Installation Service	56	321.75	18,018.00
		6800V3-Control Enclosure-Warranty Upgrade To Hi-Care Onsite Premier 24x7x4H Engineer Onsite Service	1	9824.10	9,824.10
		DAE（6800V3）-Warranty Upgrade To Hi-Care Onsite Premier 24x7x4H Engineer Onsite Service	56	3784.95	211,957.20
Total of Service (3 years)					**242,284.25**
Total Price					**1,488,036.50**

Notes:Hi-Care Premier On-Site Service include: 7*24 Technical Assistance Center Access. Access to all new software updates and Online Support. 24*7*4 Hours Onsite Hardware Replacement.

The above pricing includes hardware maintenance and software support for three years, 7 days per week, 24 hours per day. The hardware maintenance and software support provides the following:

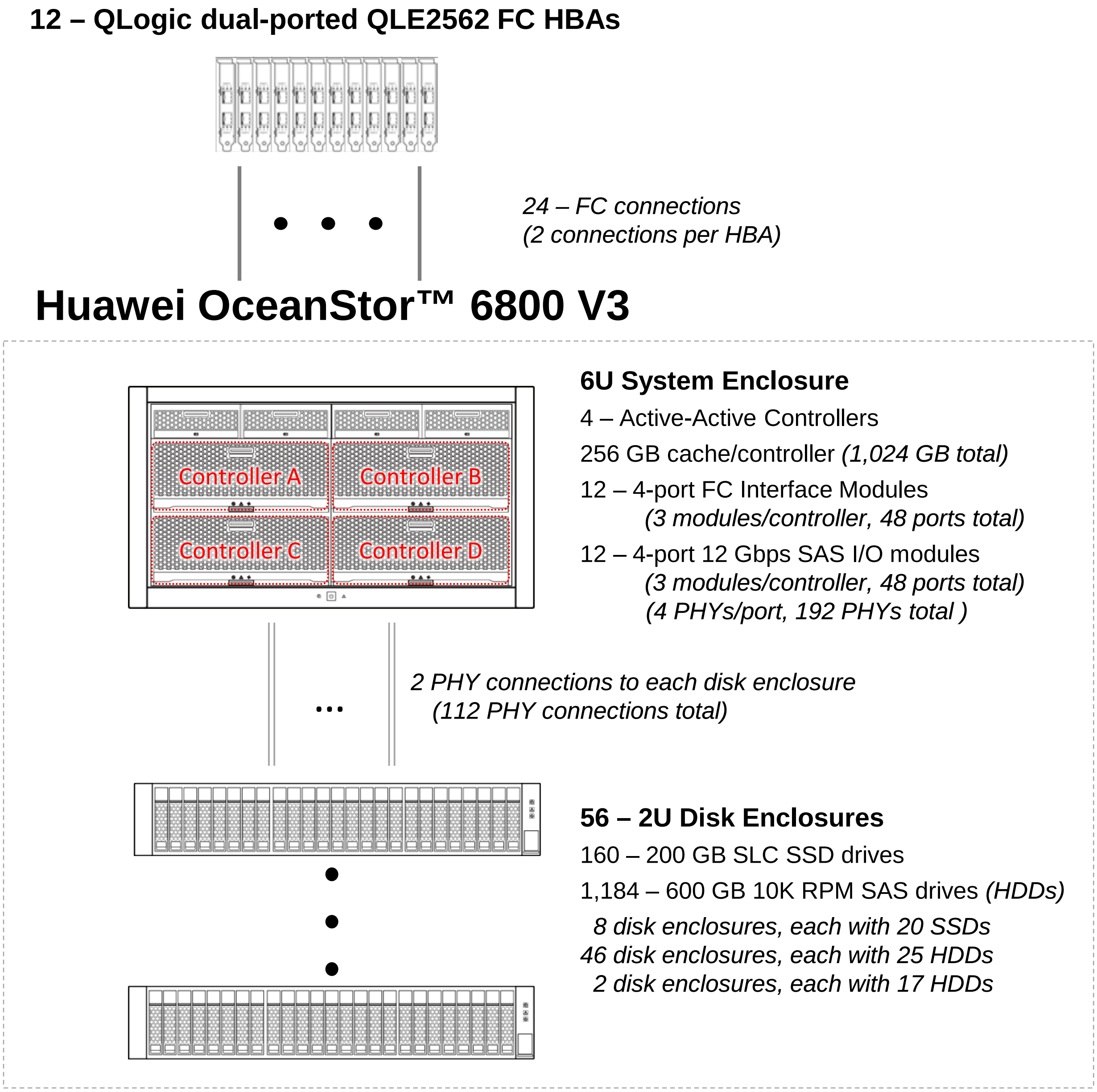
- Acknowledgement of new and existing problems within four (4) hours.
- Onsite presence of a qualified maintenance engineer or provision of a customer replaceable part within four (4) hours of the above acknowledgement for any hardware failure that results in an inoperative Price Storage Configuration that can be remedied by the repair or replacement of a Priced Storage Configuration component.

Huawei Technologies Co., Ltd. only sells its products to third-party resellers, who in turn, sell those products to U.S. customers. The above pricing, which also includes the required three-year maintenance and support, was obtained from one of those third-party resellers. See page 111 (*Appendix F: Third-Party Quotation*) for a copy of the third-party reseller quotation.

Differences between the Tested Storage Configuration (TSC) and Priced Storage Configuration

There were no differences between the TSC and Priced Storage Configuration.

Priced Storage Configuration Diagram



12 – QLogic dual-ported QLE2562 FC HBAs

24 – FC connections
(2 connections per HBA)

Huawei OceanStor™ 6800 V3

6U System Enclosure

4 – Active-Active Controllers

256 GB cache/controller *(1,024 GB total)*

12 – 4-port FC Interface Modules
(3 modules/controller, 48 ports total)

12 – 4-port 12 Gbps SAS I/O modules
(3 modules/controller, 48 ports total)
(4 PHYs/port, 192 PHYs total)

2 PHY connections to each disk enclosure
(112 PHY connections total)

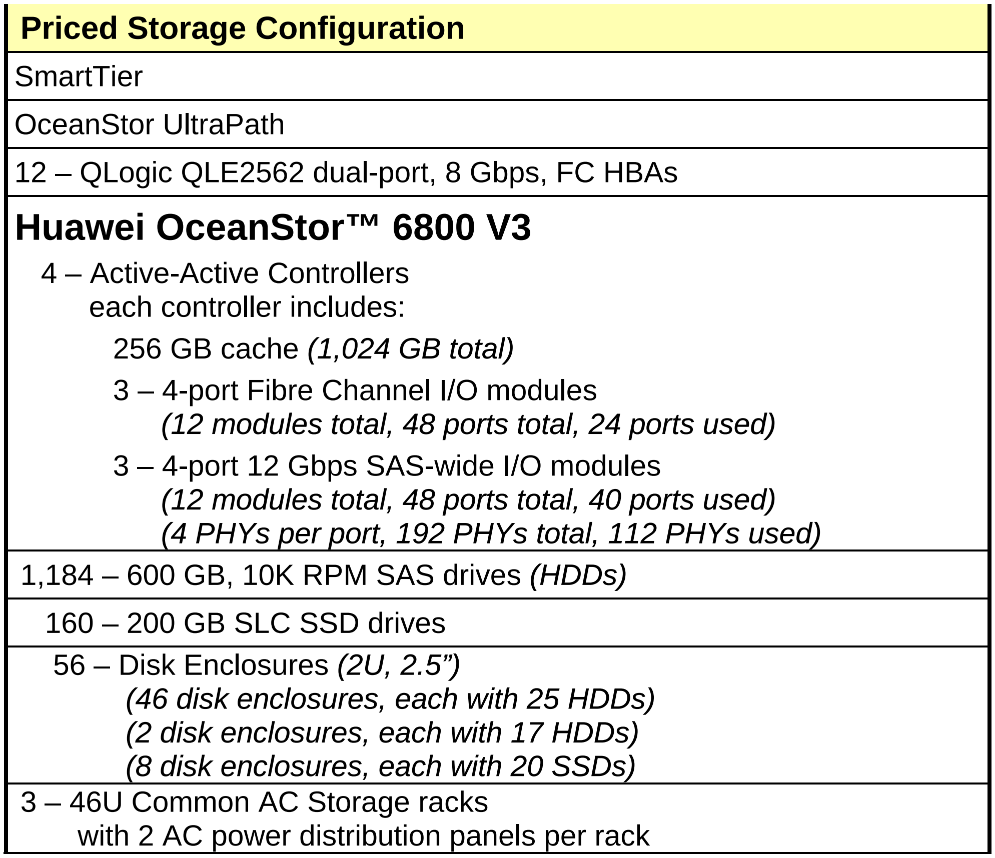
56 – 2U Disk Enclosures

160 – 200 GB SLC SSD drives

1,184 – 600 GB 10K RPM SAS drives *(HDDs)*

8 disk enclosures, each with 20 SSDs
46 disk enclosures, each with 25 HDDs
2 disk enclosures, each with 17 HDDs

Priced Storage Configuration Components



Priced Storage Configuration
SmartTier
OceanStor UltraPath
12 – QLogic QLE2562 dual-port, 8 Gbps, FC HBAs
Huawei OceanStor™ 6800 V3 4 – Active-Active Controllers each controller includes: 256 GB cache *(1,024 GB total)* 3 – 4-port Fibre Channel I/O modules *(12 modules total, 48 ports total, 24 ports used)* 3 – 4-port 12 Gbps SAS-wide I/O modules *(12 modules total, 48 ports total, 40 ports used)* *(4 PHYs per port, 192 PHYs total, 112 PHYs used)*
1,184 – 600 GB, 10K RPM SAS drives *(HDDs)*
160 – 200 GB SLC SSD drives
56 – Disk Enclosures *(2U, 2.5”)* *(46 disk enclosures, each with 25 HDDs)* *(2 disk enclosures, each with 17 HDDs)* *(8 disk enclosures, each with 20 SSDs)*
3 – 46U Common AC Storage racks with 2 AC power distribution panels per rack

In each of the following sections of this document, the appropriate Full Disclosure Report requirement, from the SPC-1 benchmark specification, is stated in italics followed by the information to fulfill the stated requirement.

CONFIGURATION INFORMATION

Benchmark Configuration (BC)/Tested Storage Configuration (TSC) Diagram

Clause 9.4.3.4.1

A one page Benchmark Configuration (BC)/Tested Storage Configuration (TSC) diagram shall be included in the FDR…

The Benchmark Configuration (BC)/Tested Storage Configuration (TSC) is illustrated on page 22 (*Benchmark Configuration/Tested Storage Configuration Diagram*).

Storage Network Configuration

Clause 9.4.3.4.1

…

5. *If the TSC contains network storage, the diagram will include the network configuration. If a single diagram is not sufficient to illustrate both the Benchmark Configuration and network configuration in sufficient detail, the Benchmark Configuration diagram will include a high-level network illustration as shown in Figure 9-8. In that case, a separate, detailed network configuration diagram will also be included as described in Clause 9.4.3.4.2.*

Clause 9.4.3.4.2

If a storage network was configured as a part of the Tested Storage Configuration and the Benchmark Configuration diagram described in Clause 9.4.3.4.1 contains a high-level illustration of the network configuration, the Executive Summary will contain a one page topology diagram of the storage network as illustrated in Figure 9-9.

The Benchmark Configuration (BC)/Tested Storage Configuration (TSC) utilized direct attached storage.

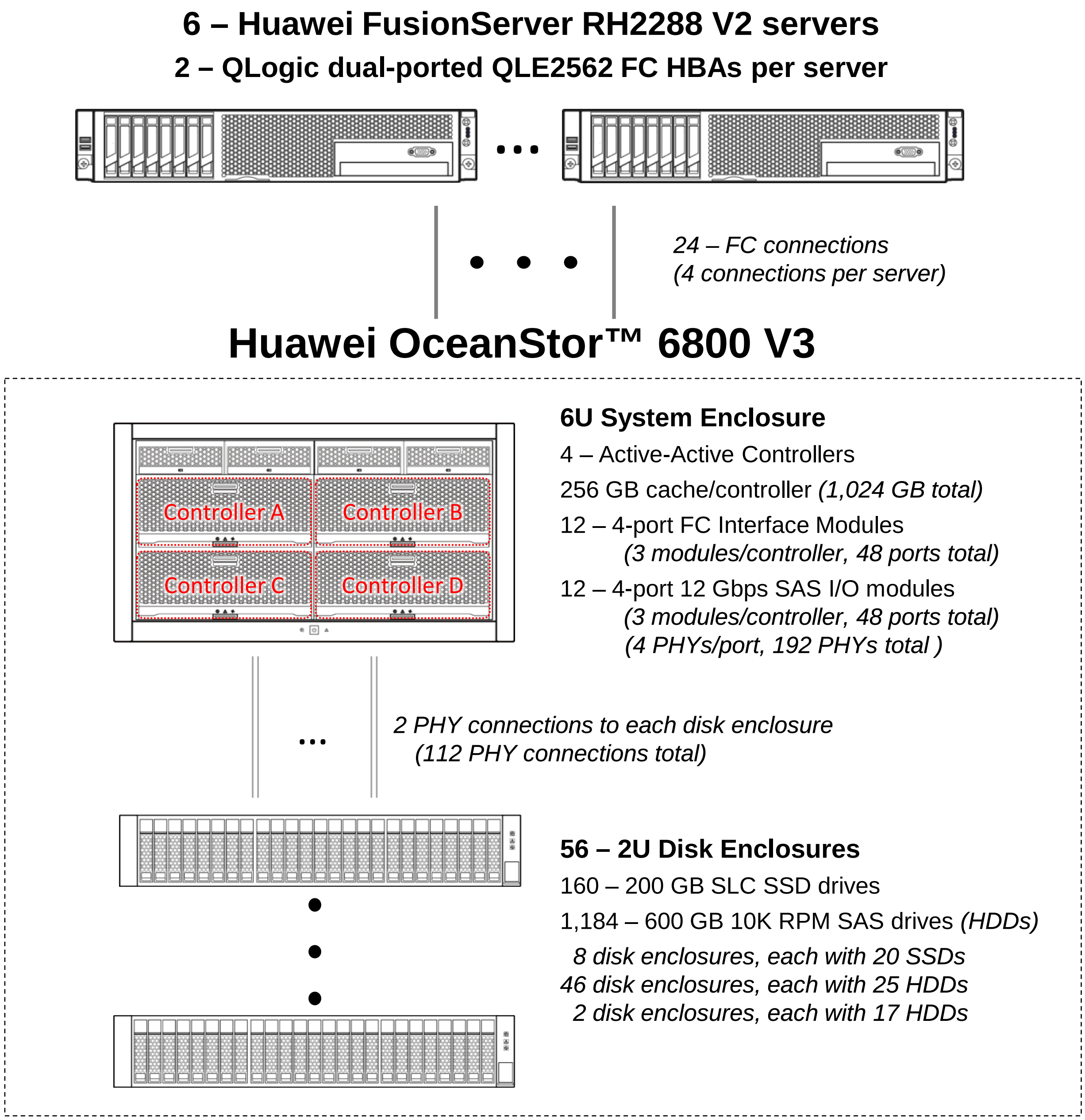
Host System(s) and Tested Storage Configuration (TSC) Table of Components

Clause 9.4.3.4.3

The FDR will contain a table that lists the major components of each Host System and the Tested Storage Configuration (TSC).

The Host System(s) and TSC table of components may be found on page 23 (*Host System and Tested Storage Configuration Components*).

Benchmark Configuration/Tested Storage Configuration Diagram



6 – Huawei FusionServer RH2288 V2 servers

2 – QLogic dual-ported QLE2562 FC HBAs per server

24 – FC connections
(4 connections per server)

Huawei OceanStor™ 6800 V3

6U System Enclosure

4 – Active-Active Controllers

256 GB cache/controller *(1,024 GB total)*

12 – 4-port FC Interface Modules
(3 modules/controller, 48 ports total)

12 – 4-port 12 Gbps SAS I/O modules
(3 modules/controller, 48 ports total)
(4 PHYs/port, 192 PHYs total)

2 PHY connections to each disk enclosure
(112 PHY connections total)

56 – 2U Disk Enclosures

160 – 200 GB SLC SSD drives

1,184 – 600 GB 10K RPM SAS drives *(HDDs)*

8 disk enclosures, each with 20 SSDs
46 disk enclosures, each with 25 HDDs
2 disk enclosures, each with 17 HDDs

Host System and Tested Storage Configuration Components

Host Systems
6 – Huawei FusionServer RH2288 V2 servers, each with: 2 – Intel® Xeon® 2.00 GHz processor E5-2620 each with 6 cores, 15 MB Intel® Smart Cache 192 GB main memory Red Hat Enterprise Linux Server release 5.5 x86_64 PCIe
Priced Storage Configuration
SmartTier
OceanStor UltraPath
12 – QLogic QLE2562 dual-port, 8 Gbps, FC HBAs
Huawei OceanStor™ 6800 V3 4 – Active-Active Controllers each controller includes: 256 GB cache *(1,024 GB total)* 3 – 4-port Fibre Channel I/O modules *(12 modules total, 48 ports total, 24 ports used)* 3 – 4-port 12 Gbps SAS-wide I/O modules *(12 modules total, 48 ports total, 40 ports used)* *(4 PHYs per port, 192 PHYs total, 112 PHYs used)*
1,184 – 600 GB, 10K RPM SAS drives *(HDDs)*
160 – 200 GB SLC SSD drives
56 – Disk Enclosures *(2U, 2.5”)* *(46 disk enclosures, each with 25 HDDs)* *(2 disk enclosures, each with 17 HDDs)* *(8 disk enclosures, each with 20 SSDs)*
3 – 46U Common AC Storage racks with 2 AC power distribution panels per rack

Customer Tunable Parameters and Options

Clause 9.4.3.5.1

All Benchmark Configuration (BC) components with customer tunable parameter and options that have been altered from their default values must be listed in the FDR. The FDR entry for each of those components must include both the name of the component and the altered value of the parameter or option. If the parameter name is not self-explanatory to a knowledgeable practitioner, a brief description of the parameter's use must also be included in the FDR entry.

Appendix B: Customer Tunable Parameters and Options on page 68 contains the customer tunable parameters and options that have been altered from their default values for this benchmark.

Tested Storage Configuration (TSC) Description

Clause 9.4.3.5.2

The FDR must include sufficient information to recreate the logical representation of the TSC. In addition to customer tunable parameters and options (Clause 4.2.4.5.3), that information must include, at a minimum:

- *A diagram and/or description of the following:*
  - ➢ *All physical components that comprise the TSC. Those components are also illustrated in the BC Configuration Diagram in Clause 9.2.4.4.1 and/or the Storage Network Configuration Diagram in Clause 9.2.4.4.2.*
  - ➢ *The logical representation of the TSC, configured from the above components that will be presented to the Workload Generator.*
- *Listings of scripts used to create the logical representation of the TSC.*
- *If scripts were not used, a description of the process used with sufficient detail to recreate the logical representation of the TSC.*

Appendix C: Tested Storage Configuration (TSC) Creation on page 69 contains the detailed information that describes how to create and configure the logical TSC.

SPC-1 Workload Generator Storage Configuration

Clause 9.4.3.5.3

The FDR must include all SPC-1 Workload Generator storage configuration commands and parameters.

The SPC-1 Workload Generator storage configuration commands and parameters for this measurement appear in Appendix D: SPC-1 Workload Generator Storage Commands and Parameters on page 88.

ASU Pre-Fill

Clause 5.3.3

Each of the three SPC-1 ASUs (ASU-1, ASU-2 and ASU-3) is required to be completely filled with specified content prior to the execution of audited SPC-1 Tests. The content is required to consist of random data pattern such as that produced by an SPC recommended tool.

The configuration file used to complete the required ASU pre-fill appears in Appendix D: SPC-1 Workload Generator Storage Commands and Parameters on page 88.

SPC-1 DATA REPOSITORY

This portion of the Full Disclosure Report presents the detailed information that fully documents the various SPC-1 storage capacities and mappings used in the Tested Storage Configuration. SPC-1 Data Repository Definitions on page 64 contains definitions of terms specific to the SPC-1 Data Repository.

Storage Capacities and Relationships

Clause 9.4.3.6.1

Two tables and four charts documenting the storage capacities and relationships of the SPC-1 Storage Hierarchy (Clause 2.1) shall be included in the FDR. … The capacity value in each chart may be listed as an integer value, for readability, rather than the decimal value listed in the table below.

SPC-1 Storage Capacities

The Physical Storage Capacity consisted of 706.374.402 GB distributed over1,184 disk drives (HDDs), each with a formatted capacity of 596.600 GB, and 32.508.068 GB distributed over 160 solid state storage devices (SSDs), each with a formatted capacity of 203.175 GB, for a total of 738,882 GB. There was 45,121.201 GB (6.11%) of Unused Storage within the Physical Storage Capacity. Global Storage Overhead consisted of 12,223.532 GB (1.65%) of the Physical Storage Capacity. There was 126,095.945 GB (18.50%) of Unused Storage within the Configured Storage Capacity. The Total ASU Capacity utilized 100% of the Addressable Storage Capacity resulting in 0.000 GB (0.00%) of Unused Storage within the Addressable Storage Capacity. The Data Protection *(mirroring)* capacity was 316,983.619 GB of which 240,518.169 GB was utilized. The total Unused Storage capacity was 171,217.146 GB.

Note: The configured Storage Devices may include additional storage capacity reserved for system overhead, which is not accessible for application use. That storage capacity may not be included in the value presented for Physical Storage Capacity.

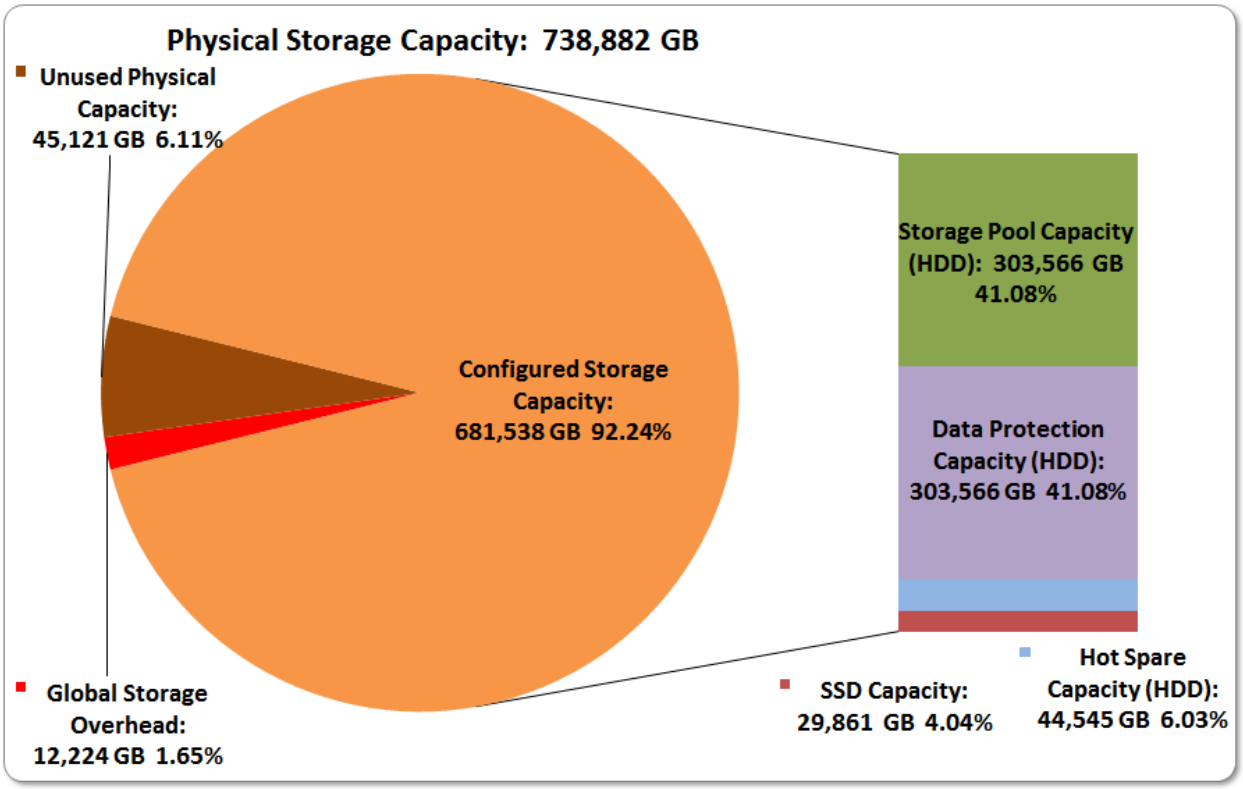
SPC-1 Storage Capacities

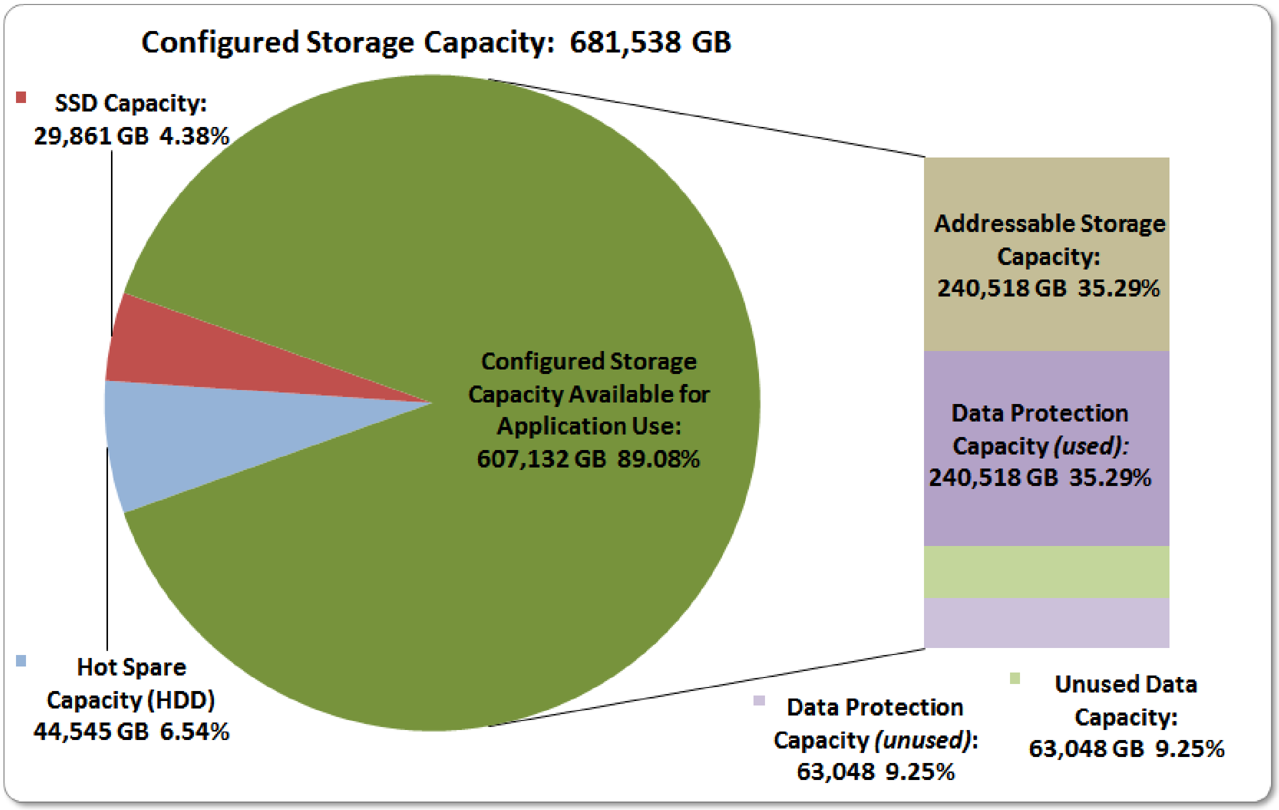
Storage Hierarchy Component	Units	Capacity
Total ASU Capacity	Gigabytes (GB)	240,518.169
Addressable Storage Capacity	Gigabytes (GB)	240,518.169
Configured Storage Capacity	Gigabytes (GB)	681,537.737
Physical Storage Capacity	Gigabytes (GB)	738,882.470
Data Protection *(Mirroring)*	Gigabytes (GB)	316,983.619
Required Storage *(sparing capacity)*	Gigabytes (GB)	74,405.455
Global Storage Overhead	Gigabytes (GB)	12,223.532
Total Unused Storage	Gigabytes (GB)	171,217.146

SPC-1 Storage Hierarchy Ratios

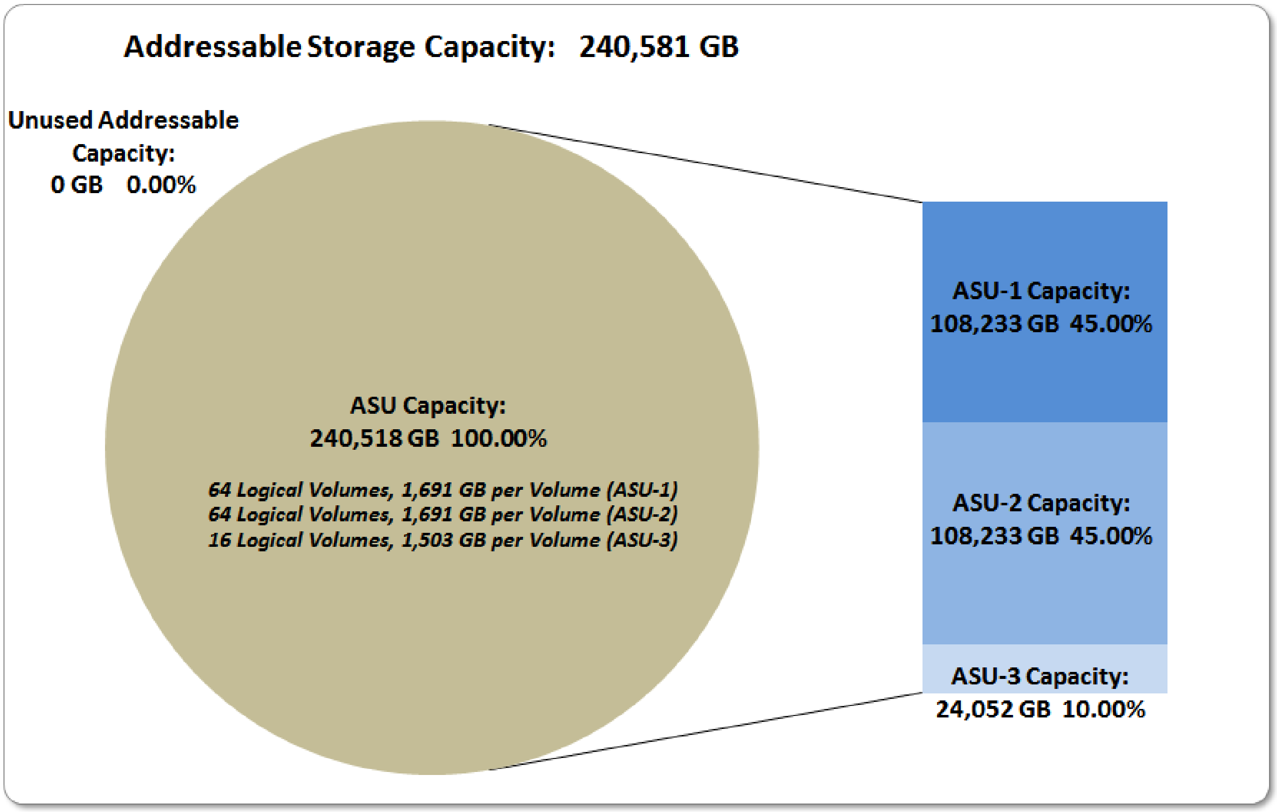
	Addressable Storage Capacity	Configured Storage Capacity	Physical Storage Capacity
Total ASU Capacity	100.00%	35.29%	32.55%
Required for Data Protection *(Mirroring)*		46.51%	42.90%
Addressable Storage Capacity		35.29%	32.55%
Required Storage *(sparing capacity)*		10.92%	10.07%
Configured Storage Capacity			92.24%
Global Storage Overhead			1.65%
Unused Storage:			
Addressable	0.00%		
Configured		18.50%	
Physical			6.11%

SPC-1 Storage Capacity Charts

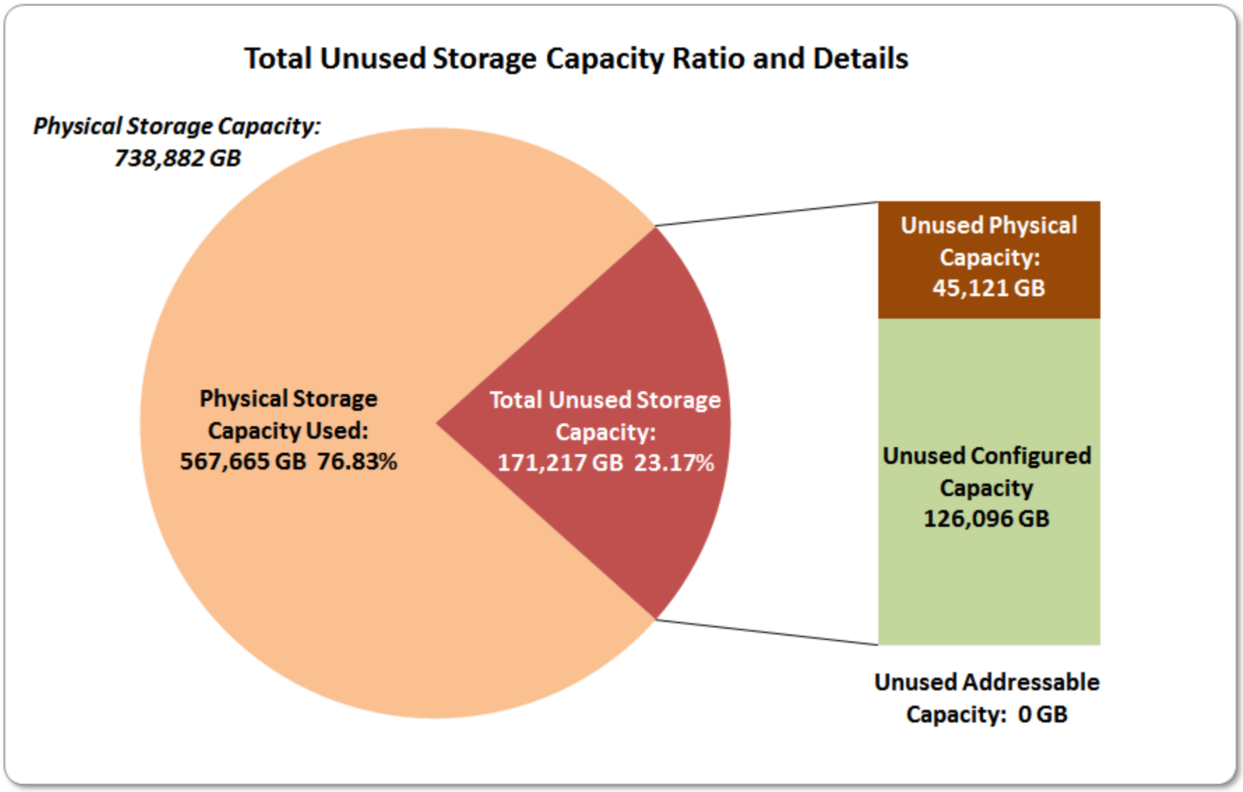




Configured Storage Capacity: 681,538 GB



Addressable Storage Capacity: 240,581 GB



Storage Capacity Utilization

Clause 9.4.3.6.2

The FDR will include a table illustrating the storage capacity utilization values defined for Application Utilization (Clause 2.8.1), Protected Application Utilization (Clause 2.8.2), and Unused Storage Ratio (Clause 2.8.3).

Clause 2.8.1

Application Utilization is defined as Total ASU Capacity divided by Physical Storage Capacity.

Clause 2.8.2

Protected Application Utilization is defined as (Total ASU Capacity plus total Data Protection Capacity minus unused Data Protection Capacity) divided by Physical Storage Capacity.

Clause 2.8.3

Unused Storage Ratio is defined as Total Unused Capacity divided by Physical Storage Capacity and may not exceed 45%.

SPC-1 Storage Capacity Utilization	
Application Utilization	32.55%
Protected Application Utilization	66.92%
Unused Storage Ratio	23.17%

Logical Volume Capacity and ASU Mapping

Clause 9.4.3.6.3

A table illustrating the capacity of each ASU and the mapping of Logical Volumes to ASUs shall be provided in the FDR. … Logical Volumes shall be sequenced in the table from top to bottom per its position in the contiguous address space of each ASU. The capacity of each Logical Volume shall be stated. … In conjunction with this table, the Test Sponsor shall provide a complete description of the type of data protection (see Clause 2.4.5) used on each Logical Volume.

Logical Volume Capacity and Mapping		
ASU-1 (108,233.176 GB)	**ASU-2 (108,233.176 GB)**	**ASU-3 (24,051.817 GB)**
64 Logical Volumes 1,691.143 GB per Logical Volume (1,691.143 GB used per Logical Volume)	64 Logical Volumes 1,691.143 GB per Logical Volume (1,691.143 GB used per Logical Volume)	64 Logical Volumes 1,503.239 GB per Logical Volume (1,503.239 GB used per Logical Volume)

The Data Protection Level used for all Logical Volumes was **Protected 2** using ***Mirroring*** as described on page 12. See "ASU Configuration" in the **IOPS Test Results File** for more detailed configuration information.

SPC-1 BENCHMARK EXECUTION RESULTS

This portion of the Full Disclosure Report documents the results of the various SPC-1 Tests, Test Phases, and Test Runs. An SPC-1 glossary on page 64 contains definitions of terms specific to the SPC-1 Tests, Test Phases, and Test Runs.

Clause 5.4.3

The Tests must be executed in the following sequence: Primary Metrics, Repeatability, and Data Persistence. That required sequence must be uninterrupted from the start of Primary Metrics to the completion of Persistence Test Run 1. Uninterrupted means the Benchmark Configuration shall not be power cycled, restarted, disturbed, altered, or adjusted during the above measurement sequence. If the required sequence is interrupted other than for the Host System/TSC power cycle between the two Persistence Test Runs, the measurement is invalid.

SPC-1 Tests, Test Phases, and Test Runs

The SPC-1 benchmark consists of the following Tests, Test Phases, and Test Runs:

- **Primary Metrics Test**
  - Sustainability Test Phase and Test Run
  - IOPS Test Phase and Test Run
  - Response Time Ramp Test Phase
    - 95% of IOPS Test Run
    - 90% of IOPS Test Run
    - 80% of IOPS Test Run
    - 50% of IOPS Test Run
    - 10% of IOPS Test Run (LRT)
- **Repeatability Test**
  - Repeatability Test Phase 1
    - 10% of IOPS Test Run (LRT)
    - IOPS Test Run
  - Repeatability Test Phase 2
    - 10% of IOPS Test Run (LRT)
    - IOPS Test Run
- **Data Persistence Test**
  - Data Persistence Test Run 1
  - Data Persistence Test Run 2

Each Test is an atomic unit that must be executed from start to finish before any other Test, Test Phase, or Test Run may be executed.

The results from each Test, Test Phase, and Test Run are listed below along with a more detailed explanation of each component.

“Ramp-Up” Test Runs

<u>*Clause 5.3.13*</u>

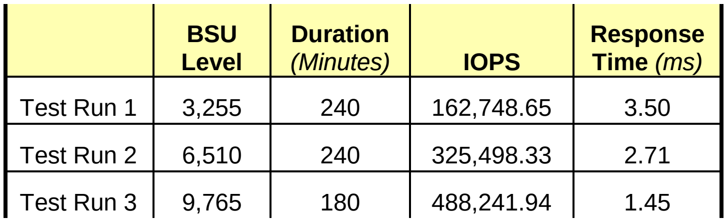
In order to warm-up caches or perform the initial ASU data migration in a multi-tier configuration, a Test Sponsor may perform a series of “Ramp-Up” Test Runs as a substitute for an initial, gradual Ramp-Up.

<u>*Clause 5.3.13.3*</u>

The “Ramp-Up” Test Runs will immediately precede the Primary Metrics Test as part of the uninterrupted SPC-1 measurement sequence.

<u>*Clause 9.4.3.7.1*</u>

If a series of “Ramp-Up” Test Runs were included in the SPC-1 measurement sequence, the FDR shall report the duration (ramp-up and measurement interval), BSU level, SPC-1 IOPS and average response time for each “Ramp-Up” Test Run in an appropriate table.



	BSU Level	**Duration** *(Minutes)*	**IOPS**	**Response Time** *(ms)*
Test Run 1	3,255	240	162,748.65	3.50
Test Run 2	6,510	240	325,498.33	2.71
Test Run 3	9,765	180	488,241.94	1.45

Primary Metrics Test – Sustainability Test Phase

<u>*Clause 5.4.4.1.1*</u>

The Sustainability Test Phase has exactly one Test Run and shall demonstrate the maximum sustainable I/O Request Throughput within at least a continuous eight (8) hour Measurement Interval. This Test Phase also serves to insure that the TSC has reached Steady State prior to reporting the final maximum I/O Request Throughput result (SPC-1 IOPS™).

<u>*Clause 5.4.4.1.2*</u>

The computed I/O Request Throughput of the Sustainability Test must be within 5% of the reported SPC-1 IOPS™ result.

<u>*Clause 5.4.4.1.4*</u>

The Average Response Time, as defined in Clause 5.1.1, will be computed and reported for the Sustainability Test Run and cannot exceed 30 milliseconds. If the Average Response time exceeds that 30-milliseconds constraint, the measurement is invalid.

<u>*Clause 9.4.3.7.2*</u>

For the Sustainability Test Phase the FDR shall contain:

1. *A Data Rate Distribution graph and data table.*
2. *I/O Request Throughput Distribution graph and data table.*
3. *A Response Time Frequency Distribution graph and table.*
4. *An Average Response Time Distribution graph and table.*

5. *The human readable Test Run Results File produced by the Workload Generator (may be included in an appendix).*
6. *A listing or screen image of all input parameters supplied to the Workload Generator (may be included in an appendix).*
7. *The Measured Intensity Multiplier for each I/O stream.*
8. *The variability of the Measured Intensity Multiplier, as defined in Clause 5.3.13.3.*

SPC-1 Workload Generator Input Parameters

The SPC-1 Workload Generator input parameters for the Sustainability, IOPS, Response Time Ramp, Repeatability, and Persistence Test Runs are documented in Appendix E: SPC-1 Workload Generator Input Parameters on Page 103.

Sustainability Test Results File

A link to the test results file generated from the Sustainability Test Run is listed below.

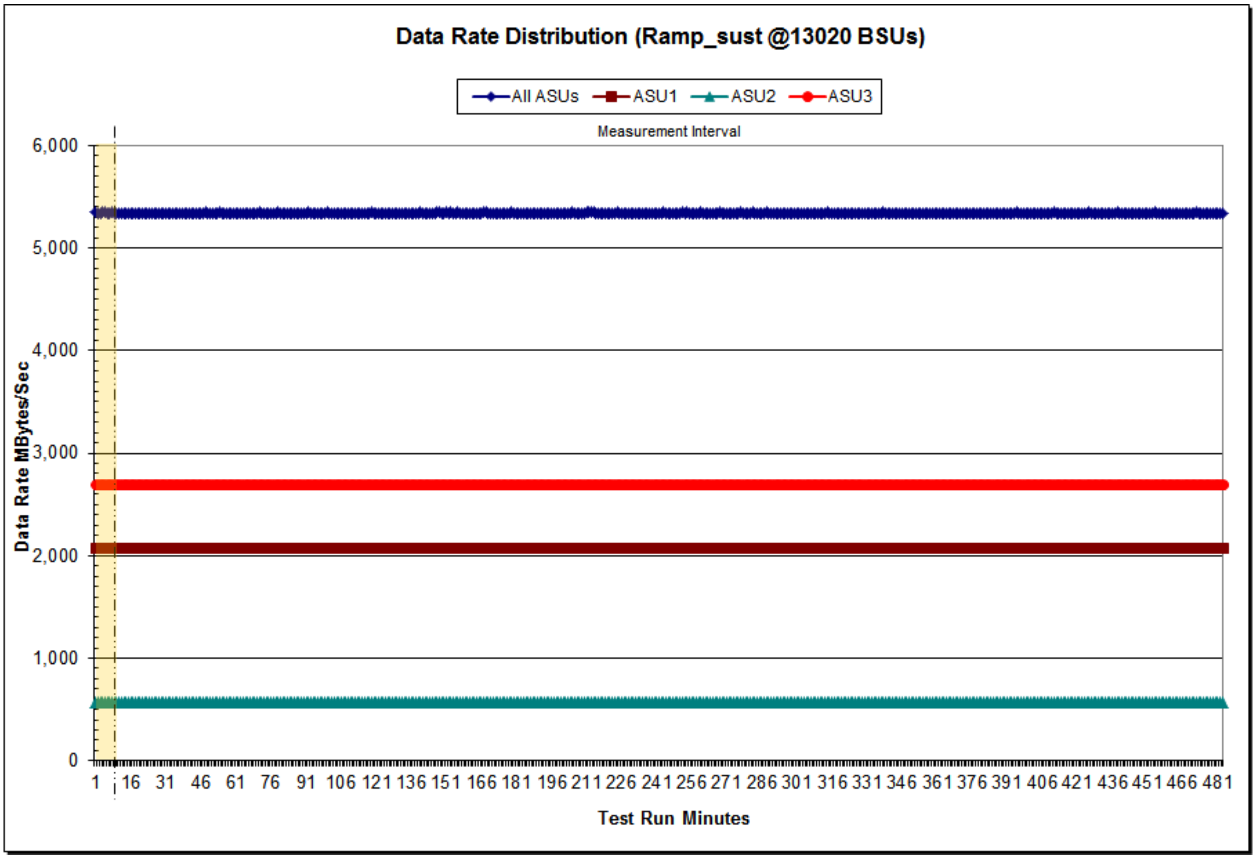
Sustainability Test Results File

Sustainability – Data Rate Distribution Data (MB/second)

The Sustainability Data Rate table of data is not embedded in this document due to its size. The table is available via the following URL:

Sustainability Data Rate Table

Sustainability – Data Rate Distribution Graph

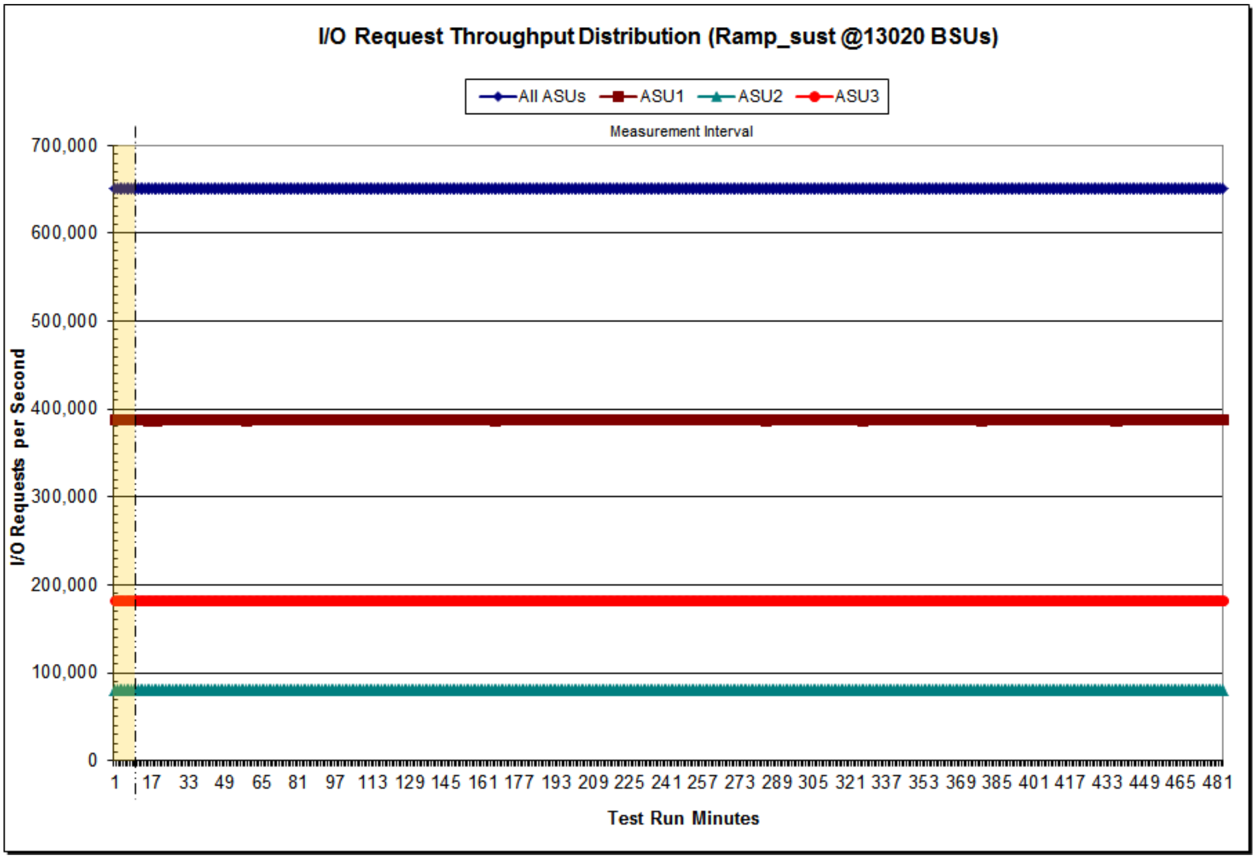


Sustainability – I/O Request Throughput Distribution Data

The Sustainability I/O Request Throughput table of data is not embedded in this document due to its size. The table is available via the following URL:

Sustainability I/O Request Throughput Table

Sustainability – I/O Request Throughput Distribution Graph

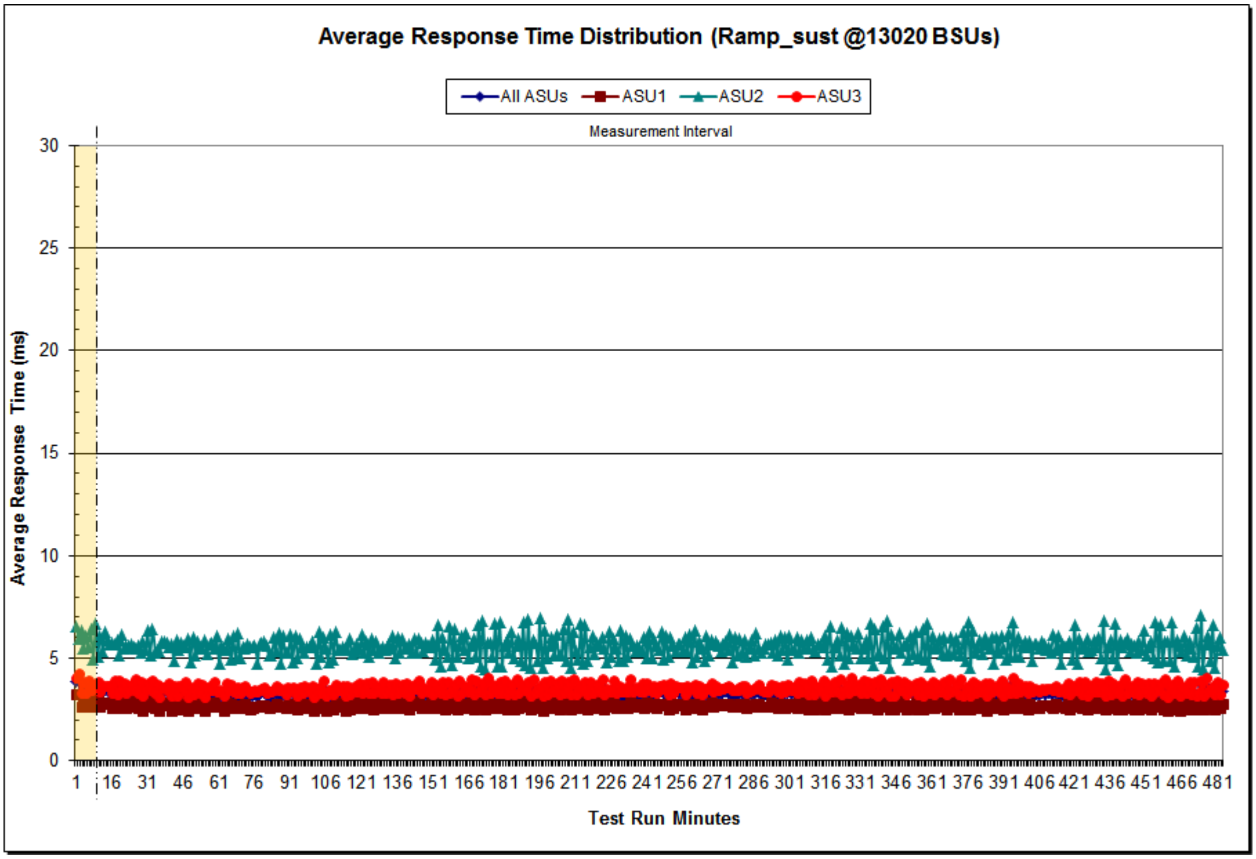


Sustainability – Average Response Time (ms) Distribution Data

The Sustainability Average Response Time table of data is not embedded in this document due to its size. The table is available via the following URL:

Sustainability Average Response Time Table

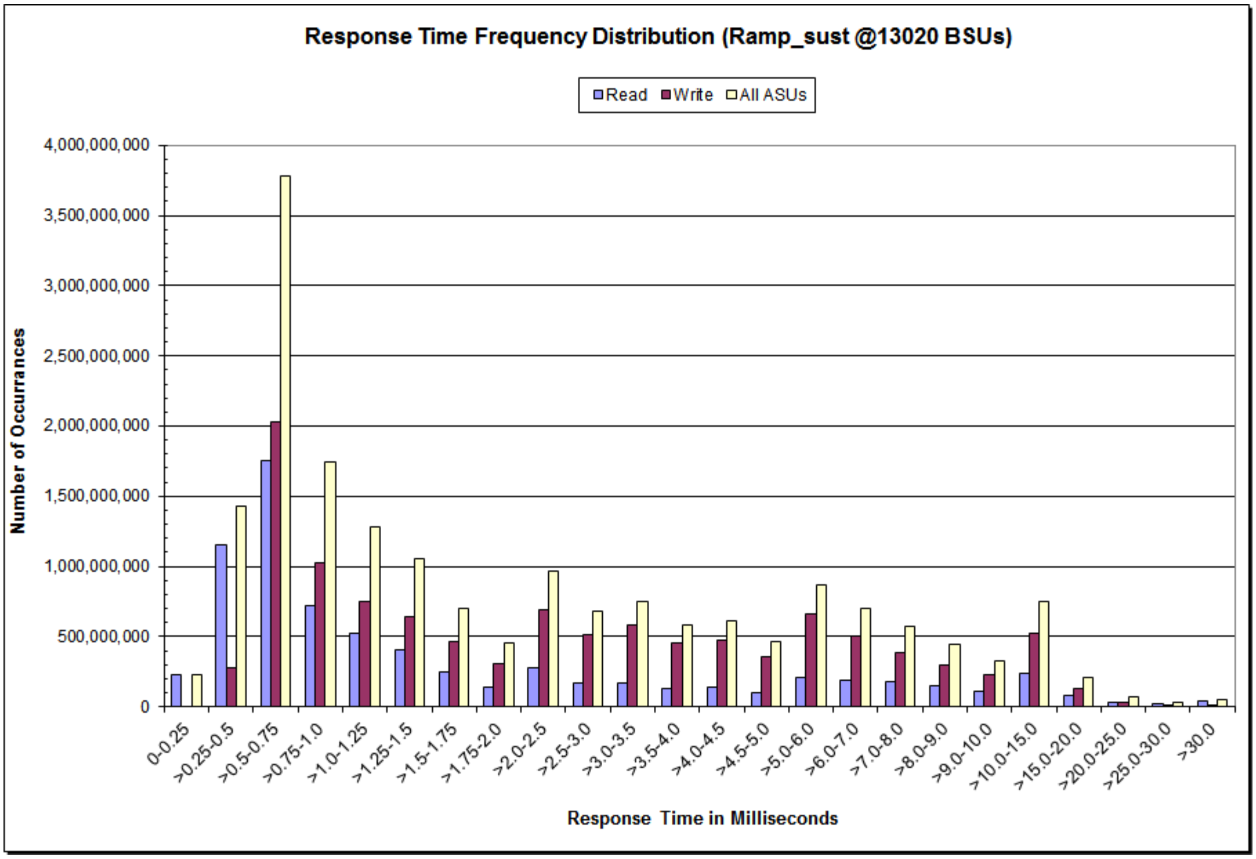
Sustainability – Average Response Time (ms) Distribution Graph



Sustainability – Response Time Frequency Distribution Data

Response Time (ms)	0-0.25	>0.25-0.5	>0.5-0.75	>0.75-1.0	>1.0-1.25	>1.25-1.5	>1.5-1.75	>1.75-2.0
Read	224,088,406	1,151,599,340	1,756,392,210	718,730,012	527,674,742	407,981,405	245,733,353	143,413,982
Write	5	281,076,014	2,029,027,762	1,029,259,128	753,926,228	643,537,627	459,821,172	306,385,578
All ASUs	224,088,411	1,432,675,354	3,785,419,972	1,747,989,140	1,281,600,970	1,051,519,032	705,554,525	449,799,560
ASU1	163,888,123	1,081,629,842	2,556,589,826	1,124,879,244	829,058,387	668,307,981	433,785,077	268,933,737
ASU2	60,200,285	222,760,391	305,240,306	144,132,592	101,706,502	83,771,698	57,840,418	38,483,043
ASU3	3	128,285,121	923,589,840	478,977,304	350,836,081	299,439,353	213,929,030	142,382,780
Response Time (ms)	**>2.0-2.5**	**>2.5-3.0**	**>3.0-3.5**	**>3.5-4.0**	**>4.0-4.5**	**>4.5-5.0**	**>5.0-6.0**	**>6.0-7.0**
Read	274,378,731	167,292,777	169,771,769	127,459,740	137,549,957	104,585,157	207,046,711	190,982,066
Write	694,688,435	510,272,285	582,559,411	450,692,199	478,929,757	356,871,975	658,674,688	506,961,564
All ASUs	969,067,166	677,565,062	752,331,180	578,151,939	616,479,714	461,457,132	865,721,399	697,943,630
ASU1	557,451,801	371,373,442	396,264,957	296,856,507	310,251,246	228,003,999	417,906,032	330,262,697
ASU2	89,887,991	69,739,127	86,094,785	72,128,125	83,813,849	67,643,790	141,378,602	131,278,545
ASU3	321,727,374	236,452,493	269,971,438	209,167,307	222,414,619	165,809,343	306,436,765	236,402,388
Response Time (ms)	**>7.0-8.0**	**>8.0-9.0**	**>9.0-10.0**	**>10.0-15.0**	**>15.0-20.0**	**>20.0-25.0**	**>25.0-30.0**	**>30.0**
Read	180,069,942	149,962,207	105,304,270	234,945,126	76,779,967	34,500,976	18,430,237	39,835,015
Write	389,660,665	297,391,849	223,600,757	518,632,534	128,942,351	35,721,945	11,101,865	6,774,242
All ASUs	569,730,607	447,354,056	328,905,027	753,577,660	205,722,318	70,222,921	29,532,102	46,609,257
ASU1	269,032,212	213,559,484	156,489,320	350,093,324	92,389,112	30,369,322	12,267,879	14,713,714
ASU2	118,537,103	94,370,891	67,244,623	157,168,938	50,563,730	22,023,851	11,631,978	28,459,524
ASU3	182,161,292	139,423,681	105,171,084	246,315,398	62,769,476	17,829,748	5,632,245	3,436,019

Sustainability – Response Time Frequency Distribution Graph



Sustainability – Measured Intensity Multiplier and Coefficient of Variation

Clause 3.4.3

IM – Intensity Multiplier: *The ratio of I/Os for each I/O stream relative to the total I/Os for all I/O streams (ASU1-1 – ASU3-1) as required by the benchmark specification.*

Clauses 5.1.10 and 5.3.15.2

MIM – Measured Intensity Multiplier: *The Measured Intensity Multiplier represents the ratio of measured I/Os for each I/O stream relative to the total I/Os measured for all I/O streams (ASU1-1 – ASU3-1). This value may differ from the corresponding Expected Intensity Multiplier by no more than 5%.*

Clause 5.3.15.3

COV – Coefficient of Variation: *This measure of variation for the Measured Intensity Multiplier cannot exceed 0.2.*

	ASU1-1	ASU1-2	ASU1-3	ASU1-4	ASU2-1	ASU2-2	ASU2-3	ASU3-1
IM	***0.0350***	***0.2810***	***0.0700***	***0.2100***	***0.0180***	***0.0700***	***0.0350***	***0.2810***
MIM	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
COV	0.001	0.000	0.001	0.000	0.001	0.001	0.001	0.000

Primary Metrics Test – IOPS Test Phase

Clause 5.4.4.2

The IOPS Test Phase consists of one Test Run at the 100% load point with a Measurement Interval of ten (10) minutes. The IOPS Test Phase immediately follows the Sustainability Test Phase without any interruption or manual intervention.

The IOPS Test Run generates the SPC-1 IOPS™ primary metric, which is computed as the I/O Request Throughput for the Measurement Interval of the IOPS Test Run.

The Average Response Time is computed for the IOPS Test Run and cannot exceed 30 milliseconds. If the Average Response Time exceeds the 30 millisecond constraint, the measurement is invalid.

Clause 9.4.3.7.3

For the IOPS Test Phase the FDR shall contain:

1. *I/O Request Throughput Distribution (data and graph).*
2. *A Response Time Frequency Distribution.*
3. *An Average Response Time Distribution.*
4. *The human readable Test Run Results File produced by the Workload Generator.*
5. *A listing or screen image of all input parameters supplied to the Workload Generator.*
6. *The total number of I/O Requests completed in the Measurement Interval as well as the number of I/O Requests with a Response Time less than or equal to 30 milliseconds and the number of I/O Requests with a Response Time greater than 30 milliseconds.*

SPC-1 Workload Generator Input Parameters

The SPC-1 Workload Generator input parameters for the Sustainability, IOPS, Response Time Ramp, Repeatability, and Persistence Test Runs are documented in Appendix E: SPC-1 Workload Generator Input Parameters on Page 103.

IOPS Test Results File

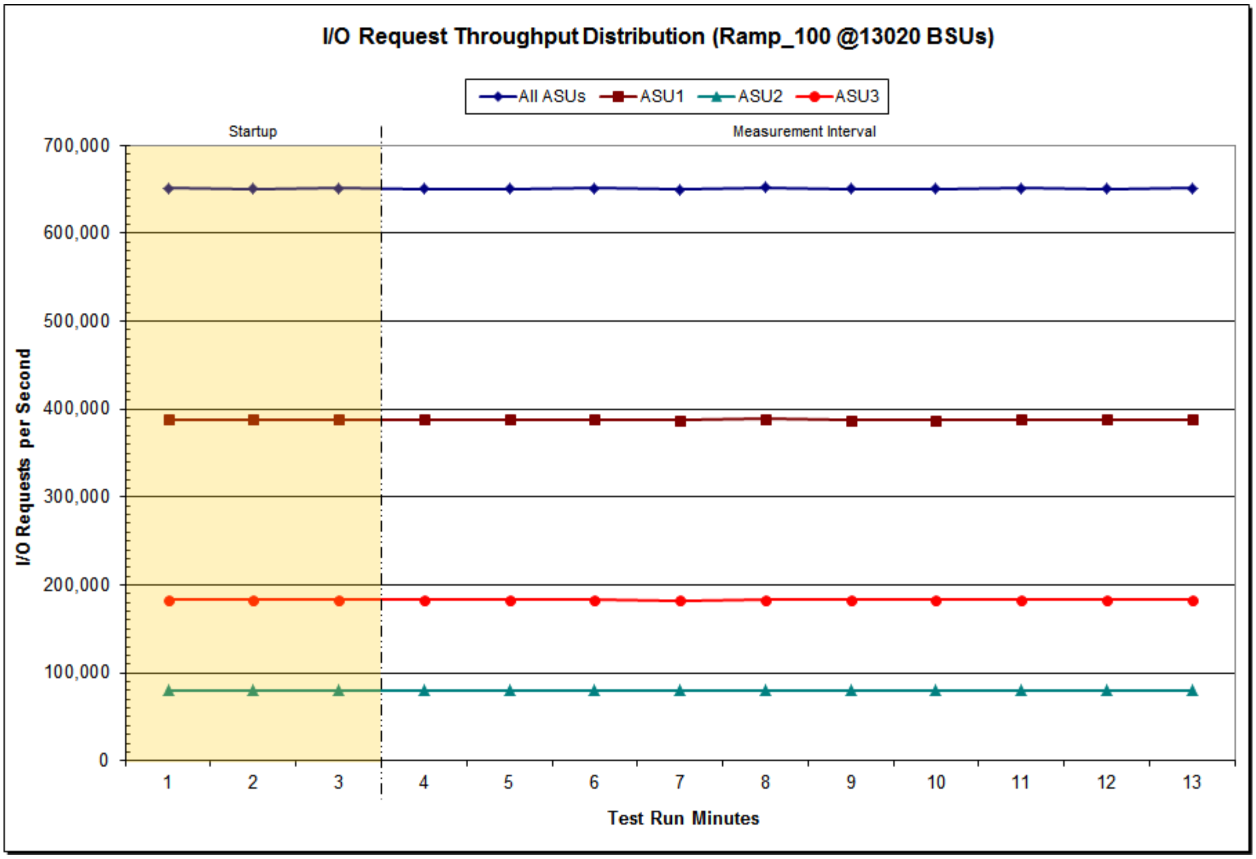
A link to the test results file generated from the IOPS Test Run is listed below.

IOPS Test Results File

IOPS Test Run – I/O Request Throughput Distribution Data

13,020 BSUs	**Start**	**Stop**	**Interval**	**Duration**
Start-Up/Ramp-Up	0:07:48	0:10:49	0-2	0:03:01
Measurement Interval	0:10:49	0:20:50	3-12	0:10:01
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	651,402.87	388,196.70	80,179.65	183,026.52
1	651,021.90	388,006.28	80,085.25	182,930.37
2	651,074.23	388,127.15	80,071.20	182,875.88
3	650,978.60	388,071.12	79,988.57	182,918.92
4	650,981.33	387,905.95	80,069.42	183,005.97
5	651,067.82	388,084.70	80,050.95	182,932.17
6	649,519.55	387,142.95	79,875.43	182,501.17
7	652,570.98	388,974.12	80,267.97	183,328.90
8	650,907.72	387,803.07	80,049.52	183,055.13
9	650,371.60	387,648.78	79,948.97	182,773.85
10	651,381.12	388,263.33	80,080.33	183,037.45
11	650,903.90	387,943.78	80,090.37	182,869.75
12	651,196.15	388,058.30	80,068.58	183,069.27
Average	***650,987.88***	***387,989.61***	***80,049.01***	***182,949.26***

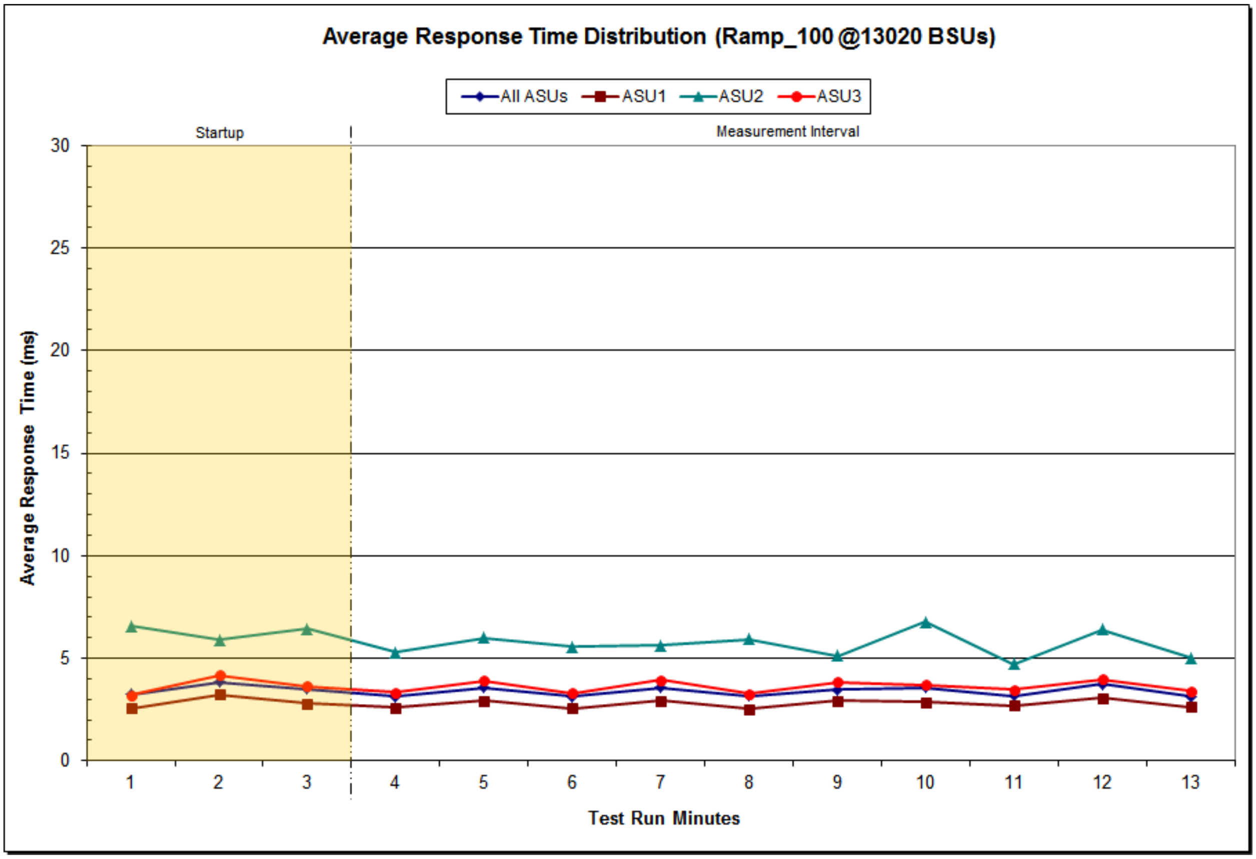
IOPS Test Run – I/O Request Throughput Distribution Graph



IOPS Test Run – Average Response Time (ms) Distribution Data

13,020 BSUs	**Start**	**Stop**	**Interval**	**Duration**
Start-Up/Ramp-Up	0:07:48	0:10:49	0-2	0:03:01
Measurement Interval	0:10:49	0:20:50	3-12	0:10:01
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	3.23	2.55	6.56	3.22
1	3.84	3.25	5.89	4.18
2	3.49	2.80	6.44	3.65
3	3.14	2.61	5.28	3.34
4	3.58	2.92	6.00	3.90
5	3.14	2.56	5.58	3.29
6	3.56	2.95	5.64	3.94
7	3.16	2.53	5.94	3.25
8	3.46	2.94	5.12	3.85
9	3.59	2.88	6.78	3.70
10	3.17	2.69	4.72	3.49
11	3.73	3.06	6.39	3.98
12	3.13	2.62	5.01	3.40
Average	***3.36***	***2.78***	***5.65***	***3.61***

IOPS Test Run – Average Response Time (ms) Distribution Graph



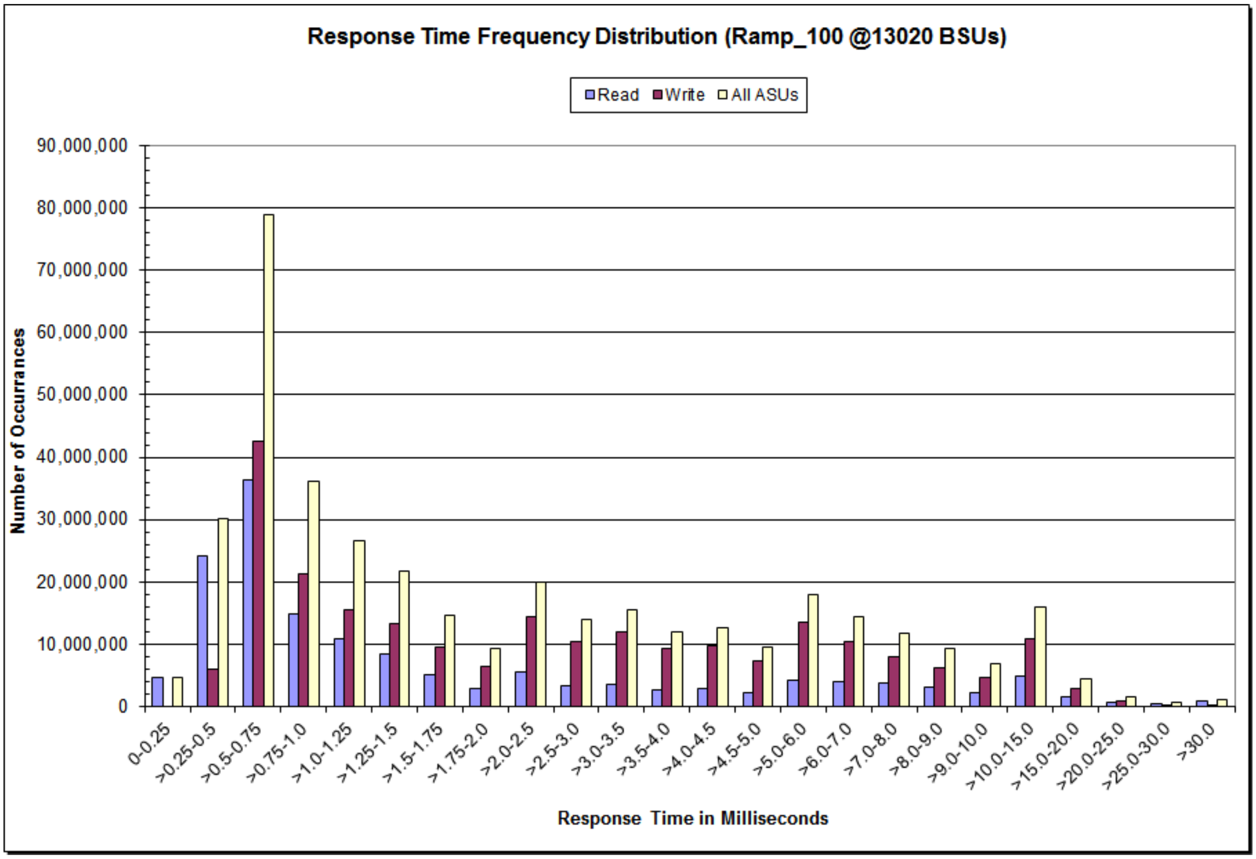
IOPS Test Run –Response Time Frequency Distribution Data

Response Time (ms)	0-0.25	>0.25-0.5	>0.5-0.75	>0.75-1.0	>1.0-1.25	>1.25-1.5	>1.5-1.75	>1.75-2.0
Read	4,717,520	24,154,004	36,469,865	14,913,536	10,915,487	8,424,785	5,083,450	2,971,350
Write	0	6,051,022	42,508,865	21,332,604	15,633,011	13,345,040	9,542,387	6,357,095
All ASUs	4,717,520	30,205,026	78,978,730	36,246,140	26,548,498	21,769,825	14,625,837	9,328,445
ASU1	3,464,231	22,804,988	53,243,445	23,321,702	17,163,329	13,824,673	8,984,994	5,574,270
ASU2	1,253,289	4,638,469	6,375,903	2,989,559	2,109,103	1,737,487	1,200,026	801,261
ASU3	0	2,761,569	19,359,382	9,934,879	7,276,066	6,207,665	4,440,817	2,952,914

Response Time (ms)	>2.0-2.5	>2.5-3.0	>3.0-3.5	>3.5-4.0	>4.0-4.5	>4.5-5.0	>5.0-6.0	>6.0-7.0
Read	5,654,180	3,454,466	3,501,689	2,644,858	2,843,495	2,172,670	4,299,644	3,976,502
Write	14,331,050	10,554,547	12,001,783	9,327,054	9,862,913	7,367,882	13,576,496	10,479,665
All ASUs	19,985,230	14,009,013	15,503,472	11,971,912	12,706,408	9,540,552	17,876,140	14,456,167
ASU1	11,488,702	7,674,901	8,164,260	6,143,093	6,392,879	4,713,207	8,631,236	6,846,964
ASU2	1,858,905	1,445,214	1,778,013	1,496,871	1,731,108	1,402,558	2,927,957	2,724,700
ASU3	6,637,623	4,888,898	5,561,199	4,331,948	4,582,421	3,424,787	6,316,947	4,884,503

Response Time (ms)	>7.0-8.0	>8.0-9.0	>9.0-10.0	>10.0-15.0	>15.0-20.0	>20.0-25.0	>25.0-30.0	>30.0
Read	3,752,755	3,135,862	2,207,857	4,975,169	1,674,418	765,676	411,922	922,555
Write	8,084,625	6,188,615	4,674,596	10,994,777	2,875,194	862,617	293,518	301,924
All ASUs	11,837,380	9,324,477	6,882,453	15,969,946	4,549,612	1,628,293	705,440	1,224,479
ASU1	5,596,898	4,463,799	3,284,505	7,456,852	2,074,041	723,705	304,449	451,783
ASU2	2,461,508	1,960,004	1,400,627	3,296,440	1,081,471	477,997	254,717	625,803
ASU3	3,778,974	2,900,674	2,197,321	5,216,654	1,394,100	426,591	146,274	146,893

IOPS Test Run –Response Time Frequency Distribution Graph



IOPS Test Run – I/O Request Information

I/O Requests Completed in the Measurement Interval	I/O Requests Completed with Response Time = or < 30 ms	I/O Requests Completed with Response Time > 30 ms
390,590,995	389,366,516	1,224,479

IOPS Test Run – Measured Intensity Multiplier and Coefficient of Variation

Clause 3.4.3

IM – Intensity Multiplier: *The ratio of I/Os for each I/O stream relative to the total I/Os for all I/O streams (ASU1-1 – ASU3-1) as required by the benchmark specification.*

Clauses 5.1.10 and 5.3.15.2

MIM – Measured Intensity Multiplier: *The Measured Intensity Multiplier represents the ratio of measured I/Os for each I/O stream relative to the total I/Os measured for all I/O streams (ASU1-1 – ASU3-1). This value may differ from the corresponding Expected Intensity Multiplier by no more than 5%.*

Clause 5.3.15.3

COV – Coefficient of Variation: *This measure of variation for the Measured Intensity Multiplier cannot exceed 0.2.*

	ASU1-1	ASU1-2	ASU1-3	ASU1-4	ASU2-1	ASU2-2	ASU2-3	ASU3-1
IM	***0.0350***	***0.2810***	***0.0700***	***0.2100***	***0.0180***	***0.0700***	***0.0350***	***0.2810***
MIM	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
COV	0.001	0.000	0.001	0.000	0.002	0.001	0.001	0.000

Primary Metrics Test – Response Time Ramp Test Phase

Clause 5.4.4.3

The Response Time Ramp Test Phase consists of five Test Runs, one each at 95%, 90%, 80%, 50%, and 10% of the load point (100%) used to generate the SPC-1 IOPS™ primary metric. Each of the five Test Runs has a Measurement Interval of ten (10) minutes. The Response Time Ramp Test Phase immediately follows the IOPS Test Phase without any interruption or manual intervention.

The five Response Time Ramp Test Runs, in conjunction with the IOPS Test Run (100%), demonstrate the relationship between Average Response Time and I/O Request Throughput for the Tested Storage Configuration (TSC) as illustrated in the response time/throughput curve on page 16.

In addition, the Average Response Time measured during the 10% Test Run is the value for the SPC-1 LRT™ metric. That value represents the Average Response Time of a lightly loaded TSC.

Clause 9.4.3.7.4

The following content shall appear in the FDR for the Response Time Ramp Phase:

1. *A Response Time Ramp Distribution.*
2. *The human readable Test Run Results File produced by the Workload Generator for each Test Run within the Response Time Ramp Test Phase.*
3. *For the 10% Load Level Test Run (SPC-1 LRT™ metric) an Average Response Time Distribution.*
4. *A listing or screen image of all input parameters supplied to the Workload Generator.*

SPC-1 Workload Generator Input Parameters

The SPC-1 Workload Generator input parameters for the Sustainability, IOPS, Response Time Ramp, Repeatability, and Persistence Test Runs are documented in Appendix E: SPC-1 Workload Generator Input Parameters on Page 103.

Response Time Ramp Test Results File

A link to each test result file generated from each Response Time Ramp Test Run list listed below.

95% Load Level

90% Load Level

80% Load Level

50% Load Level

10% Load Level

Response Time Ramp Distribution (IOPS) Data

The five Test Runs that comprise the Response Time Ramp Phase are executed at 95%, 90%, 80%, 50%, and 10% of the Business Scaling Unit (BSU) load level used to produce the SPC-1 IOPS™ primary metric. The 100% BSU load level is included in the following Response Time Ramp data table and graph for completeness.

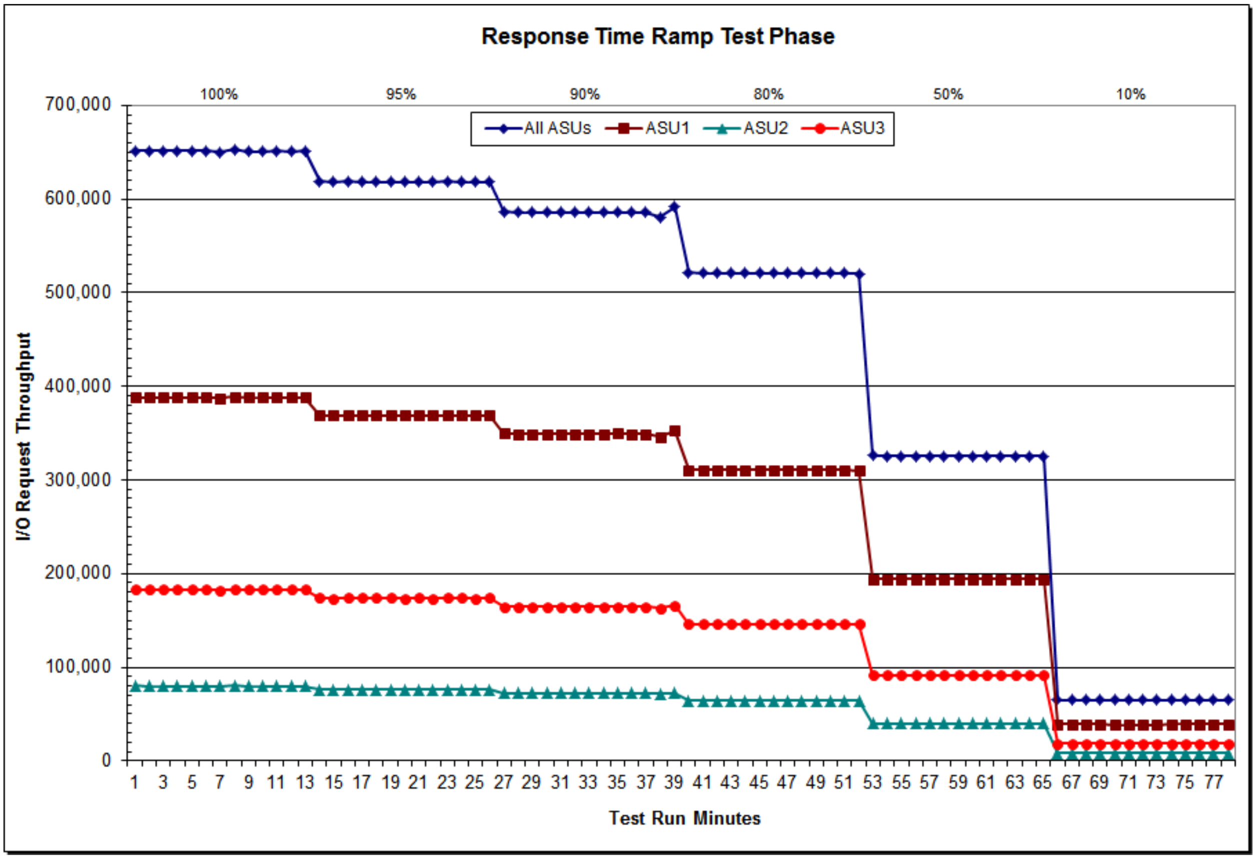
100% Load Level: 13,020 BSUs	**Start**	**Stop**	**Interval**	**Duration**	**95% Load Level: 12,369 BSUs**	**Start**	**Stop**	**Interval**	**Duration**
Start-Up/Ramp-Up	0:07:48	0:10:49	0-3	0:03:01	**Start-Up/Ramp-Up**	0:32:17	0:35:18	0-3	0:03:01
Measurement Interval	0:10:49	0:20:50	3-12	0:10:01	**Measurement Interval**	0:35:18	0:45:20	3-12	0:10:02
(60 second intervals)	**All ASUs**	**ASU-1**	**ASU-2**	**ASU-3**	*(60 second intervals)*	**All ASUs**	**ASU-1**	**ASU-2**	**ASU-3**
0	651,402.87	388,196.70	80,179.65	183,026.52	**0**	618,754.55	368,749.95	76,122.43	173,882.17
1	651,021.90	388,006.28	80,085.25	182,930.37	**1**	618,318.75	368,498.93	76,060.28	173,759.53
2	651,074.23	388,127.15	80,071.20	182,875.88	**2**	618,689.88	368,672.15	76,122.73	173,895.00
3	650,978.60	388,071.12	79,988.57	182,918.92	**3**	618,489.22	368,643.22	76,072.92	173,773.08
4	650,981.33	387,905.95	80,069.42	183,005.97	**4**	618,484.55	368,667.33	76,044.65	173,772.57
5	651,067.82	388,084.70	80,050.95	182,932.17	**5**	618,398.83	368,549.63	76,065.50	173,783.70
6	649,519.55	387,142.95	79,875.43	182,501.17	**6**	618,434.55	368,666.08	76,038.58	173,729.88
7	652,570.98	388,974.12	80,267.97	183,328.90	**7**	618,411.45	368,555.60	76,084.02	173,771.83
8	650,907.72	387,803.07	80,049.52	183,055.13	**8**	618,446.02	368,748.85	75,990.80	173,706.37
9	650,371.60	387,648.78	79,948.97	182,773.85	**9**	618,531.67	368,684.52	76,050.57	173,796.58
10	651,381.12	388,263.33	80,080.33	183,037.45	**10**	618,347.85	368,515.08	76,016.35	173,816.42
11	650,903.90	387,943.78	80,090.37	182,869.75	**11**	618,362.62	368,566.93	76,058.57	173,737.12
12	651,196.15	388,058.30	80,068.58	183,069.27	**12**	618,467.35	368,605.42	76,019.13	173,842.80
Average	***650,987.88***	***387,989.61***	***80,049.01***	***182,949.26***	***Average***	***618,437.41***	***368,620.27***	***76,044.11***	***173,773.04***

90% Load Level: 11,718 BSUs	**Start**	**Stop**	**Interval**	**Duration**	**80% Load Level: 10,416 BSUs**	**Start**	**Stop**	**Interval**	**Duration**
Start-Up/Ramp-Up	0:55:54	0:58:55	0-3	0:03:01	**Start-Up/Ramp-Up**	1:19:46	1:22:47	0-3	0:03:01
Measurement Interval	0:58:55	1:08:56	3-12	0:10:01	**Measurement Interval**	1:22:47	1:32:51	3-12	0:10:04
(60 second intervals)	**All ASUs**	**ASU-1**	**ASU-2**	**ASU-3**	*(60 second intervals)*	**All ASUs**	**ASU-1**	**ASU-2**	**ASU-3**
0	586,378.50	349,508.62	72,108.28	164,761.60	**0**	521,351.12	310,797.27	64,077.50	146,476.35
1	585,886.42	349,202.03	72,055.40	164,628.98	**1**	520,904.82	310,446.55	64,093.45	146,364.82
2	585,910.68	349,184.55	72,103.73	164,622.40	**2**	520,845.33	310,401.23	64,087.45	146,356.65
3	585,739.15	349,082.98	72,057.08	164,599.08	**3**	520,804.17	310,370.88	63,984.25	146,449.03
4	585,841.13	349,097.92	72,100.47	164,642.75	**4**	520,626.18	310,315.85	63,977.12	146,333.22
5	585,944.83	349,205.57	72,135.15	164,604.12	**5**	520,759.93	310,377.02	64,077.65	146,305.27
6	585,888.35	349,187.75	72,015.37	164,685.23	**6**	520,775.23	310,342.32	64,111.05	146,321.87
7	586,014.93	349,276.42	72,036.55	164,701.97	**7**	520,644.27	310,328.73	64,050.38	146,265.15
8	585,917.25	349,298.73	72,013.22	164,605.30	**8**	520,716.65	310,280.58	64,074.05	146,362.02
9	585,896.08	349,227.77	72,078.50	164,589.82	**9**	520,820.78	310,401.13	64,090.90	146,328.75
10	585,748.18	349,084.18	72,027.62	164,636.38	**10**	520,756.47	310,345.68	64,088.58	146,322.20
11	580,486.92	345,948.33	71,401.27	163,137.32	**11**	520,839.85	310,474.40	64,045.20	146,320.25
12	591,354.40	352,478.35	72,684.83	166,191.22	**12**	520,312.82	310,072.15	63,985.25	146,255.42
Average	***585,883.12***	***349,188.80***	***72,055.01***	***164,639.32***	***Average***	***520,705.64***	***310,330.88***	***64,048.44***	***146,326.32***

Response Time Ramp Distribution (IOPS) Data *(continued)*

50% Load Level: **6,510 BSUs**	**Start**	**Stop**	**Interval**	**Duration**	**10% Load Level:** **1,302 BSUs**	**Start**	**Stop**	**Interval**	**Duration**
Start-Up/Ramp-Up	1:43:11	1:46:12	0-3	0:03:01	**Start-Up/Ramp-Up**	2:05:41	2:08:42	0-3	0:03:01
Measurement Interval	1:46:12	1:56:13	3-12	0:10:01	**Measurement Interval**	2:08:42	2:18:43	3-12	0:10:01
(60 second intervals)	**All ASUs**	**ASU-1**	**ASU-2**	**ASU-3**	*(60 second intervals)*	**All ASUs**	**ASU-1**	**ASU-2**	**ASU-3**
0	325,783.03	194,205.50	40,060.22	91,517.32	**0**	65,158.73	38,843.65	8,022.33	18,292.75
1	325,586.30	194,077.28	40,073.83	91,435.18	**1**	65,127.53	38,812.57	8,003.82	18,311.15
2	325,474.33	194,002.88	40,018.98	91,452.47	**2**	65,111.85	38,807.02	7,994.03	18,310.80
3	325,470.60	194,016.90	39,982.37	91,471.33	**3**	65,172.27	38,864.38	8,008.48	18,299.40
4	325,524.48	194,015.45	40,009.55	91,499.48	**4**	65,044.55	38,747.60	8,021.05	18,275.90
5	325,400.85	193,899.30	40,030.30	91,471.25	**5**	65,055.62	38,767.87	8,009.12	18,278.63
6	325,430.58	193,879.00	40,064.20	91,487.38	**6**	65,129.78	38,787.83	8,014.67	18,327.28
7	325,558.35	193,987.40	40,052.32	91,518.63	**7**	65,069.48	38,796.65	7,989.13	18,283.70
8	325,523.63	194,006.43	40,006.73	91,510.47	**8**	65,089.70	38,815.48	7,987.97	18,286.25
9	325,445.68	193,913.53	40,056.30	91,475.85	**9**	65,099.37	38,804.63	8,017.95	18,276.78
10	325,495.73	193,973.98	40,005.22	91,516.53	**10**	65,078.28	38,809.40	7,991.87	18,277.02
11	325,633.15	194,073.98	40,029.35	91,529.82	**11**	65,120.98	38,813.60	8,019.70	18,287.68
12	325,446.85	194,010.97	40,005.90	91,429.98	**12**	65,123.73	38,815.63	8,014.32	18,293.78
Average	***325,492.99***	***193,977.70***	***40,024.22***	***91,491.07***	***Average***	***65,098.38***	***38,802.31***	***8,007.43***	***18,288.64***

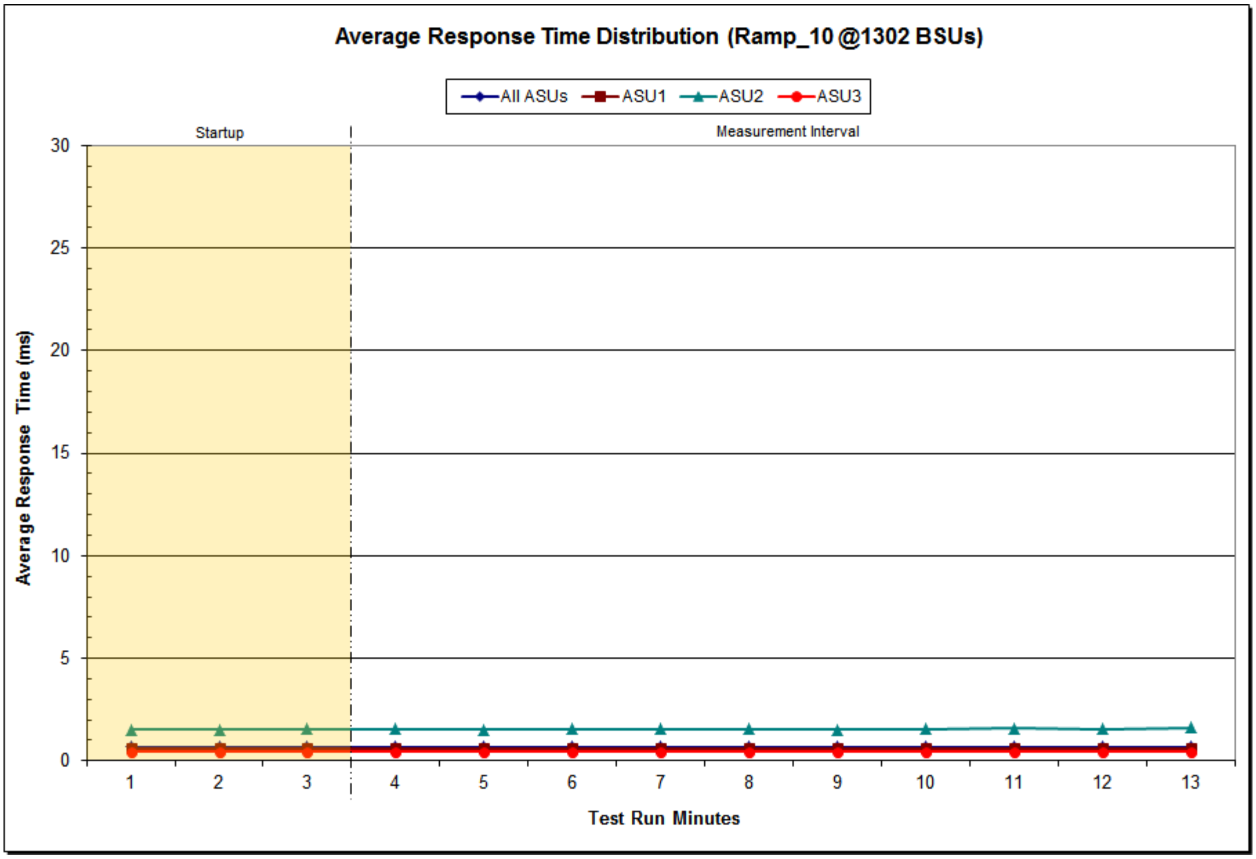
Response Time Ramp Distribution (IOPS) Graph



SPC-1 LRT™ Average Response Time (ms) Distribution Data

1,302 BSUs	**Start**	**Stop**	**Interval**	**Duration**
Start-Up/Ramp-Up	2:05:41	2:08:42	0-2	0:03:01
Measurement Interval	2:08:42	2:18:43	3-12	0:10:01
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	0.67	0.60	1.51	0.44
1	0.67	0.60	1.52	0.44
2	0.68	0.61	1.56	0.45
3	0.68	0.60	1.57	0.44
4	0.67	0.60	1.51	0.44
5	0.68	0.61	1.55	0.44
6	0.68	0.61	1.55	0.45
7	0.68	0.61	1.54	0.45
8	0.67	0.60	1.51	0.44
9	0.68	0.61	1.56	0.45
10	0.68	0.60	1.58	0.45
11	0.68	0.61	1.56	0.45
12	0.68	0.60	1.61	0.45
Average	***0.68***	***0.61***	***1.55***	***0.45***

SPC-1 LRT™ Average Response Time (ms) Distribution Graph



SPC-1 LRT™ (10%) – Measured Intensity Multiplier and Coefficient of Variation

Clause 3.4.3

IM – Intensity Multiplier: *The ratio of I/Os for each I/O stream relative to the total I/Os for all I/O streams (ASU1-1 – ASU3-1) as required by the benchmark specification.*

Clauses 5.1.10 and 5.3.15.2

MIM – Measured Intensity Multiplier: *The Measured Intensity Multiplier represents the ratio of measured I/Os for each I/O stream relative to the total I/Os measured for all I/O streams (ASU1-1 – ASU3-1). This value may differ from the corresponding Expected Intensity Multiplier by no more than 5%.*

Clause 5.3.15.3

COV – Coefficient of Variation: *This measure of variation for the Measured Intensity Multiplier cannot exceed 0.2.*

	ASU1-1	ASU1-2	ASU1-3	ASU1-4	ASU2-1	ASU2-2	ASU2-3	ASU3-1
IM	***0.0350***	***0.2810***	***0.0700***	***0.2100***	***0.0180***	***0.0700***	***0.0350***	***0.2810***
MIM	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2809
COV	0.003	0.001	0.003	0.001	0.002	0.002	0.003	0.001

Repeatability Test

Clause 5.4.5

The Repeatability Test demonstrates the repeatability and reproducibility of the SPC-1 IOPS™ primary metric and the SPC-1 LRT™ metric generated in earlier Test Runs.

There are two identical Repeatability Test Phases. Each Test Phase contains two Test Runs. Each of the Test Runs will have a Measurement Interval of no less than ten (10) minutes. The two Test Runs in each Test Phase will be executed without interruption or any type of manual intervention.

The first Test Run in each Test Phase is executed at the 10% load point. The Average Response Time from each of the Test Runs is compared to the SPC-1 LRT™ metric. Each Average Response Time value must be less than the SPC-1 LRT™ metric plus 5% or less than the SPC-1 LRT™ metric plus one (1) millisecond (ms).

The second Test Run in each Test Phase is executed at the 100% load point. The I/O Request Throughput from the Test Runs is compared to the SPC-1 IOPS™ primary metric. Each I/O Request Throughput value must be greater than the SPC-1 IOPS™ primary metric minus 5%. In addition, the Average Response Time for each Test Run cannot exceed 30 milliseconds.

If any of the above constraints are not met, the benchmark measurement is invalid.

Clause 9.4.3.7.5

The following content shall appear in the FDR for each Test Run in the two Repeatability Test Phases:

1. *A table containing the results of the Repeatability Test.*
2. *An I/O Request Throughput Distribution graph and table.*
3. *An Average Response Time Distribution graph and table.*
4. *The human readable Test Run Results File produced by the Workload Generator.*
5. *A listing or screen image of all input parameters supplied to the Workload Generator.*

SPC-1 Workload Generator Input Parameters

The SPC-1 Workload Generator input parameters for the Sustainability, IOPS, Response Time Ramp, Repeatability, and Persistence Test Runs are documented in Appendix E: SPC-1 Workload Generator Input Parameters on Page 103.

Repeatability Test Results File

The values for the SPC-1 IOPS™, SPC-1 LRT™, and the Repeatability Test measurements are listed in the tables below.

	SPC-1 IOPS™
Primary Metrics	***650,987.88***
Repeatability Test Phase 1	650,993.35
Repeatability Test Phase 2	651,024.29

The SPC-1 IOPS™ values in the above table were generated using 100% of the specified Business Scaling Unit (BSU) load level. Each of the Repeatability Test Phase values for SPC-1 IOPS™ must greater than 95% of the reported SPC-1 IOPS™ Primary Metric.

	SPC-1 LRT™
Primary Metrics	***0.68 ms***
Repeatability Test Phase 1	0.68 ms
Repeatability Test Phase 2	0.68 ms

The average response time values in the SPC-1 LRT™ column were generated using 10% of the specified Business Scaling Unit (BSU) load level. Each of the Repeatability Test Phase values for SPC-1 LRT™ must be less than 105% of the reported SPC-1 LRT™ Primary Metric or less than the reported SPC-1 LRT™ Primary Metric plus one (1) millisecond (ms).

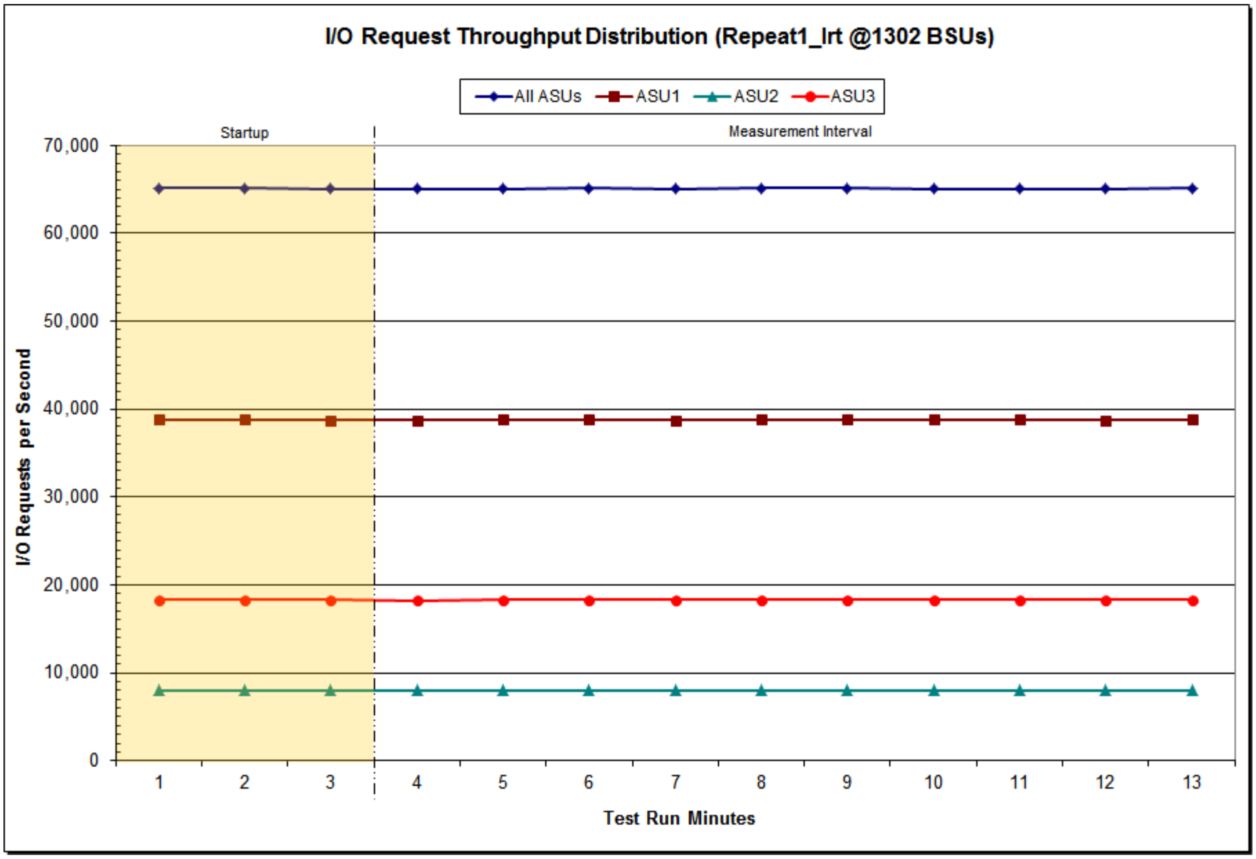
A link to the test result file generated from each Repeatability Test Run is listed below.

Repeatability Test Phase 1, Test Run 1 (LRT)
Repeatability Test Phase 1, Test Run 2 (IOPS)
Repeatability Test Phase 2, Test Run 1 (LRT)
Repeatability Test Phase 2, Test Run 2 (IOPS)

Repeatability 1 LRT – I/O Request Throughput Distribution Data

1,302 BSUs	**Start**	**Stop**	**Interval**	**Duration**
Start-Up/Ramp-Up	2:28:24	2:31:24	0-2	0:03:00
Measurement Interval	2:31:24	2:41:25	3-12	0:10:01
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	65,151.42	38,852.95	8,022.45	18,276.02
1	65,148.88	38,803.90	8,028.98	18,316.00
2	65,080.08	38,773.23	8,019.88	18,286.97
3	65,058.22	38,765.37	8,017.82	18,275.03
4	65,074.70	38,792.02	8,004.78	18,277.90
5	65,165.52	38,811.83	8,018.45	18,335.23
6	65,041.47	38,751.65	7,992.43	18,297.38
7	65,111.75	38,796.82	8,016.32	18,298.62
8	65,153.80	38,836.43	8,005.57	18,311.80
9	65,080.08	38,793.87	8,009.53	18,276.68
10	65,090.55	38,798.37	8,000.10	18,292.08
11	65,041.07	38,741.58	8,005.83	18,293.65
12	65,140.87	38,820.25	8,015.68	18,304.93
Average	***65,095.80***	***38,790.82***	***8,008.65***	***18,296.33***

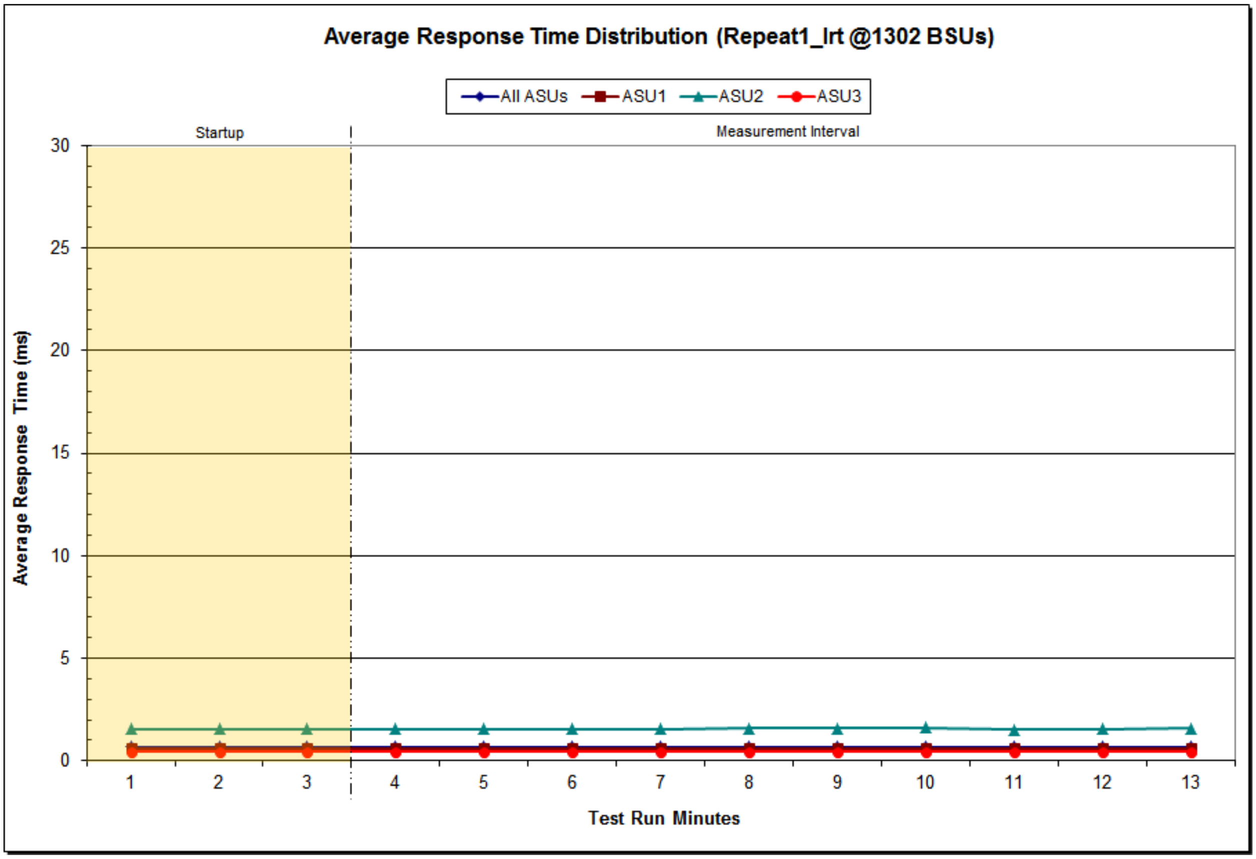
Repeatability 1 LRT – I/O Request Throughput Distribution Graph



Repeatability 1 LRT –Average Response Time (ms) Distribution Data

1,302 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	2:28:24	2:31:24	0-2	0:03:00
Measurement Interval	2:31:24	2:41:25	3-12	0:10:01
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	0.67	0.60	1.54	0.44
1	0.68	0.60	1.54	0.45
2	0.67	0.60	1.56	0.44
3	0.67	0.60	1.55	0.44
4	0.68	0.61	1.54	0.45
5	0.68	0.61	1.57	0.46
6	0.68	0.60	1.56	0.44
7	0.68	0.60	1.58	0.45
8	0.69	0.61	1.60	0.45
9	0.68	0.60	1.60	0.45
10	0.67	0.60	1.52	0.44
11	0.68	0.60	1.55	0.45
12	0.68	0.60	1.59	0.45
Average	***0.68***	***0.60***	***1.57***	***0.45***

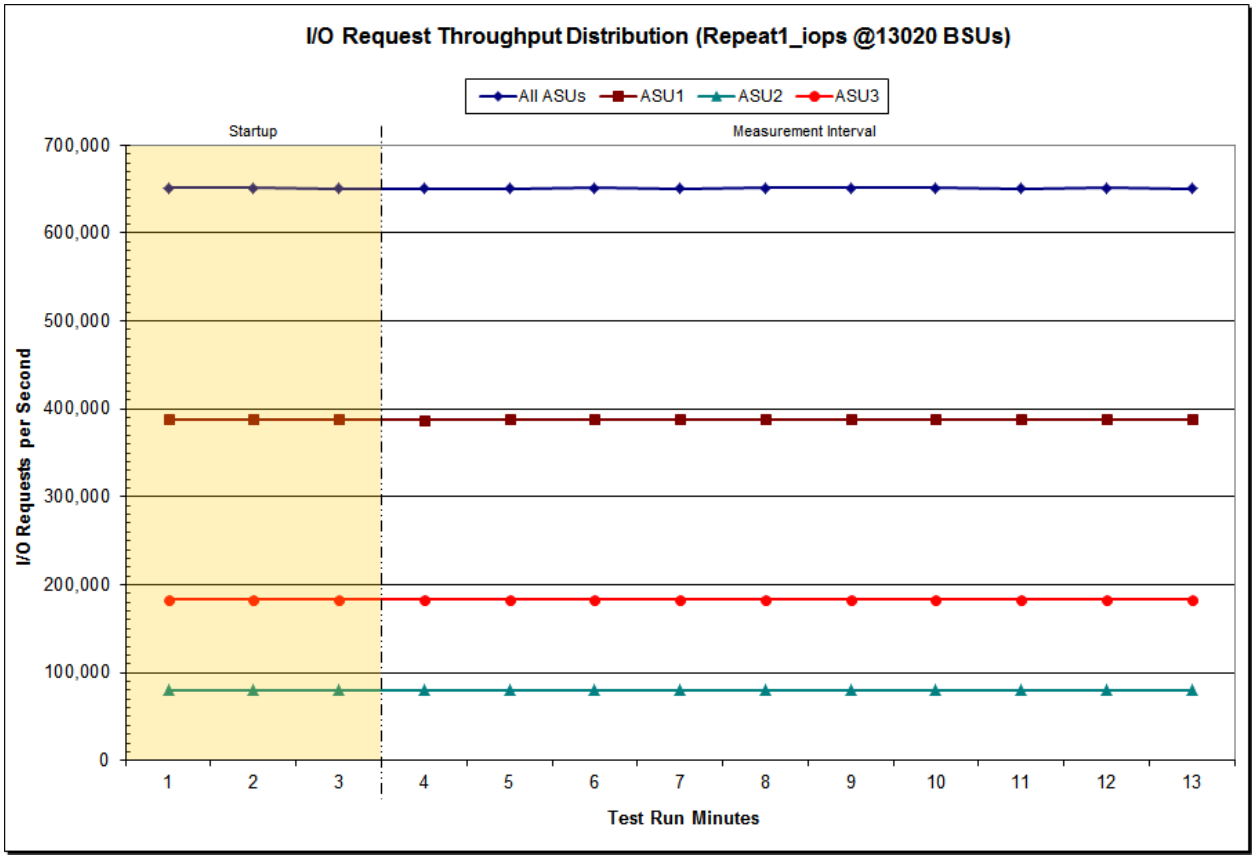
Repeatability 1 LRT –Average Response Time (ms) Distribution Graph



Repeatability 1 IOPS – I/O Request Throughput Distribution Data

13,020 BSUs	**Start**	**Stop**	**Interval**	**Duration**
Start-Up/Ramp-Up	2:52:00	2:55:01	0-2	0:03:01
Measurement Interval	2:55:01	3:05:02	3-12	0:10:01
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	651,613.12	388,412.75	80,164.72	183,035.65
1	651,084.82	388,100.33	80,023.42	182,961.07
2	650,942.37	388,061.22	80,031.78	182,849.37
3	650,743.17	387,800.37	80,095.88	182,846.92
4	650,949.98	387,941.03	80,057.52	182,951.43
5	651,068.98	388,087.82	79,991.55	182,989.62
6	650,881.48	388,016.22	80,069.12	182,796.15
7	651,057.87	387,919.65	80,117.95	183,020.27
8	651,092.25	388,047.55	80,070.03	182,974.67
9	651,072.97	388,155.58	79,999.57	182,917.82
10	650,992.22	388,132.60	80,027.82	182,831.80
11	651,185.07	388,006.02	80,149.83	183,029.22
12	650,889.55	388,014.50	80,033.90	182,841.15
Average	***650,993.35***	***388,012.13***	***80,061.32***	***182,919.90***

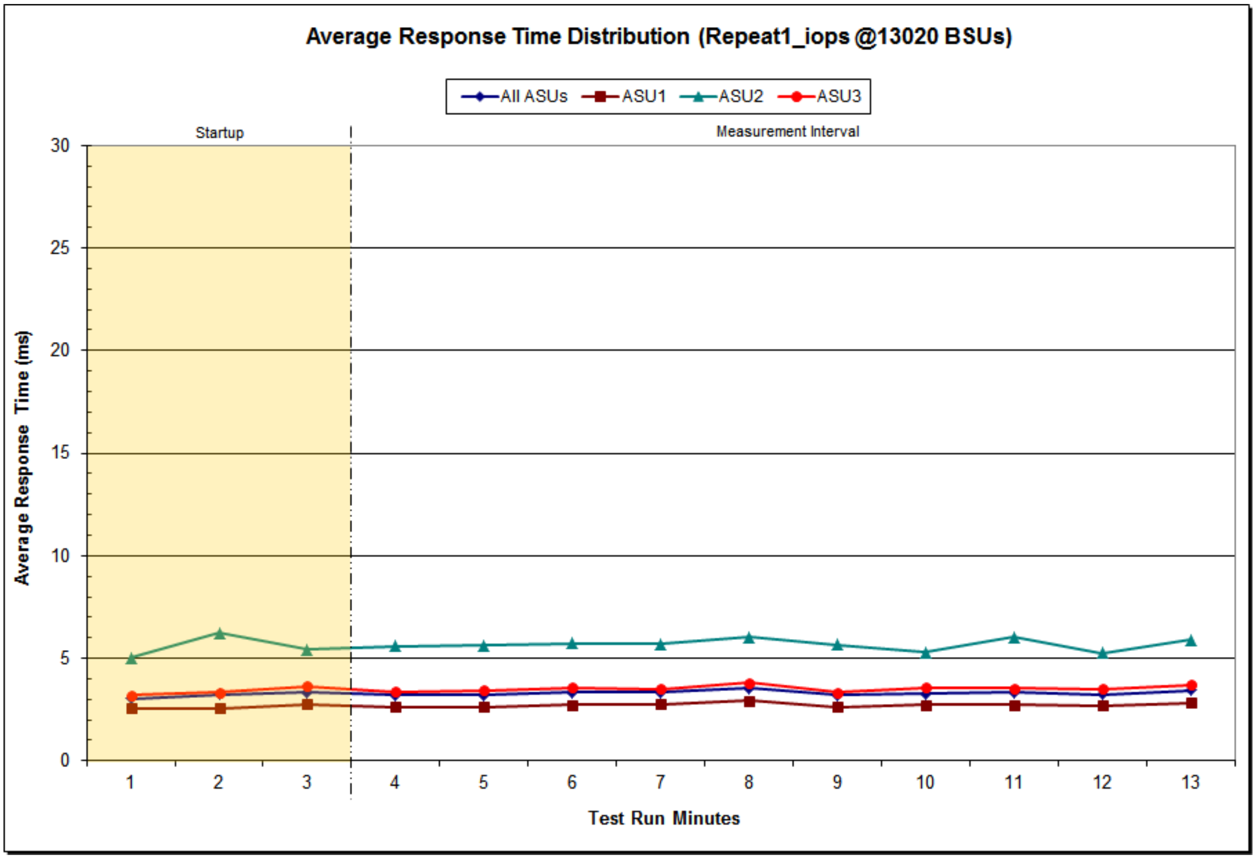
Repeatability 1 IOPS – I/O Request Throughput Distribution Graph



Repeatability 1 IOPS –Average Response Time (ms) Distribution Data

13,020 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	2:52:00	2:55:01	0-2	0:03:01
Measurement Interval	2:55:01	3:05:02	3-12	0:10:01
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	3.04	2.55	5.01	3.20
1	3.23	2.56	6.24	3.32
2	3.34	2.76	5.43	3.63
3	3.22	2.64	5.60	3.39
4	3.22	2.62	5.63	3.42
5	3.33	2.72	5.73	3.57
6	3.33	2.77	5.69	3.49
7	3.55	2.92	6.01	3.82
8	3.20	2.62	5.65	3.35
9	3.27	2.72	5.28	3.57
10	3.35	2.72	6.03	3.53
11	3.23	2.68	5.25	3.51
12	3.45	2.83	5.90	3.69
Average	***3.31***	***2.72***	***5.68***	***3.53***

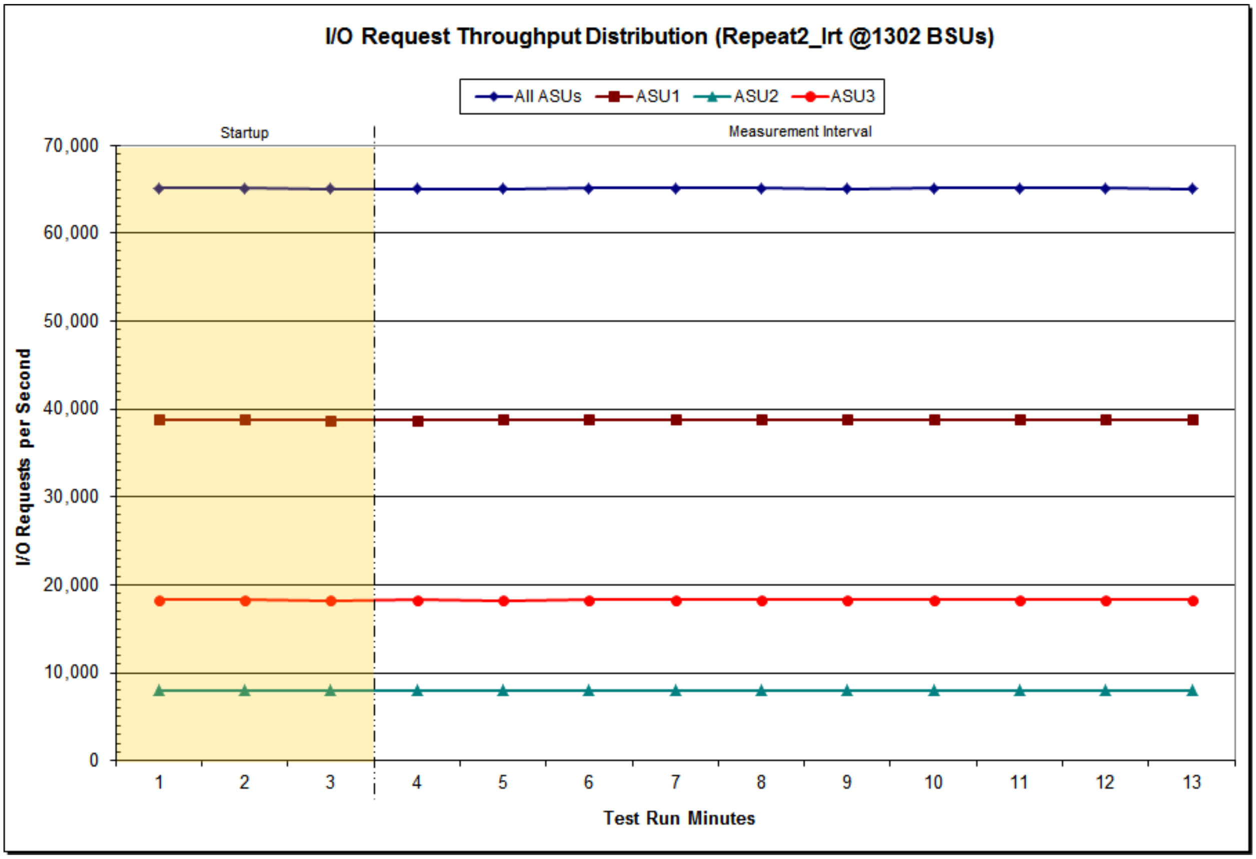
Repeatability 1 IOPS –Average Response Time (ms) Distribution Graph



Repeatability 2 LRT – I/O Request Throughput Distribution Data

1,302 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	3:14:14	3:17:14	0-2	0:03:00
Measurement Interval	3:17:14	3:27:15	3-12	0:10:01
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	65,166.72	38,822.75	8,029.25	18,314.72
1	65,107.70	38,825.55	8,002.52	18,279.63
2	65,028.05	38,780.45	7,987.98	18,259.62
3	65,090.45	38,767.87	8,005.50	18,317.08
4	65,057.02	38,785.55	7,998.47	18,273.00
5	65,113.62	38,804.65	8,008.45	18,300.52
6	65,139.30	38,796.08	8,020.90	18,322.32
7	65,154.27	38,807.60	8,025.85	18,320.82
8	65,071.87	38,789.28	7,994.28	18,288.30
9	65,146.83	38,801.50	8,012.85	18,332.48
10	65,127.05	38,829.73	7,992.95	18,304.37
11	65,105.55	38,807.85	8,010.88	18,286.82
12	65,069.55	38,791.05	7,990.98	18,287.52
Average	***65,107.55***	***38,798.12***	***8,006.11***	***18,303.32***

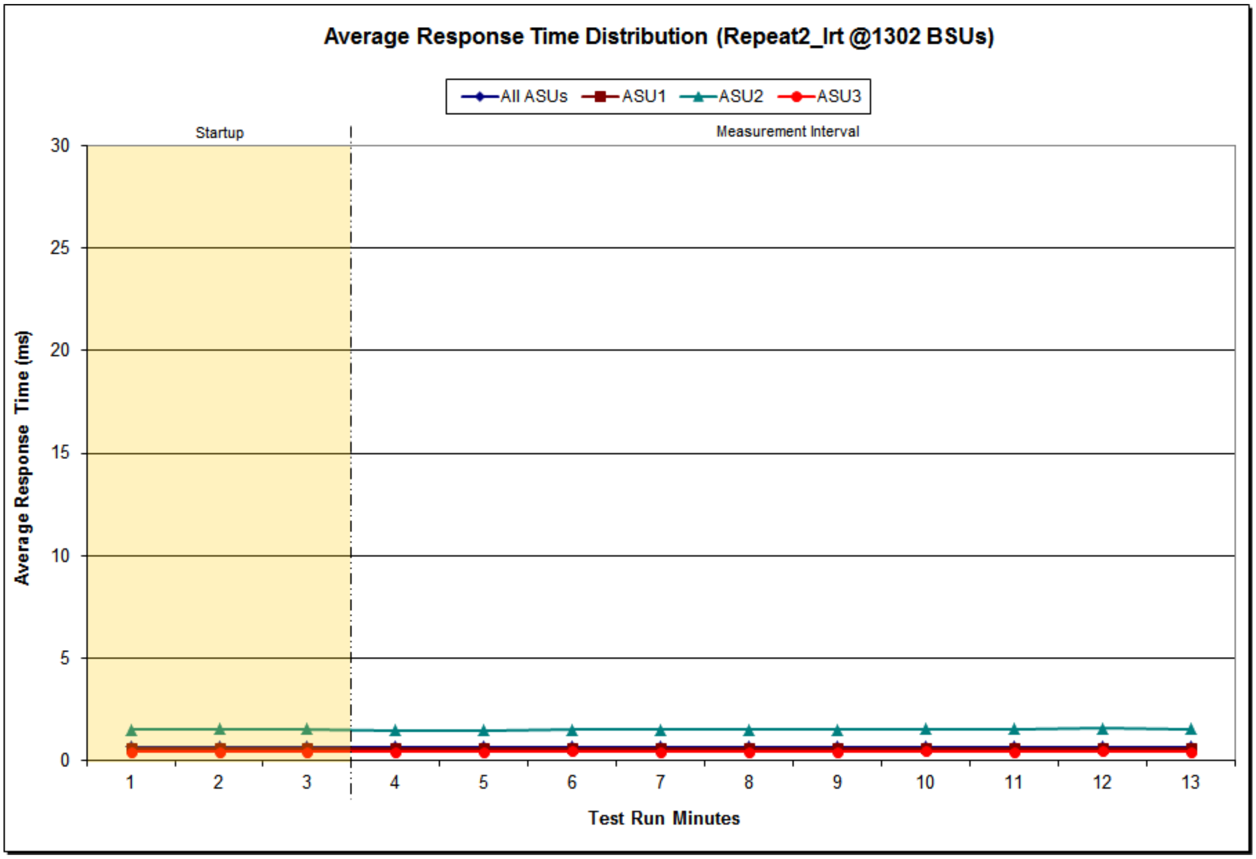
Repeatability 2 LRT – I/O Request Throughput Distribution Graph



Repeatability 2 LRT –Average Response Time (ms) Distribution Data

1,302 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	3:14:14	3:17:14	0-2	0:03:00
Measurement Interval	3:17:14	3:27:15	3-12	0:10:01
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	0.67	0.60	1.53	0.43
1	0.68	0.60	1.54	0.46
2	0.68	0.60	1.55	0.45
3	0.66	0.60	1.49	0.44
4	0.66	0.60	1.49	0.44
5	0.68	0.61	1.52	0.47
6	0.67	0.60	1.52	0.44
7	0.68	0.61	1.54	0.46
8	0.68	0.61	1.53	0.45
9	0.69	0.61	1.55	0.46
10	0.68	0.61	1.54	0.45
11	0.69	0.61	1.58	0.46
12	0.68	0.61	1.56	0.44
Average	***0.68***	***0.61***	***1.53***	***0.45***

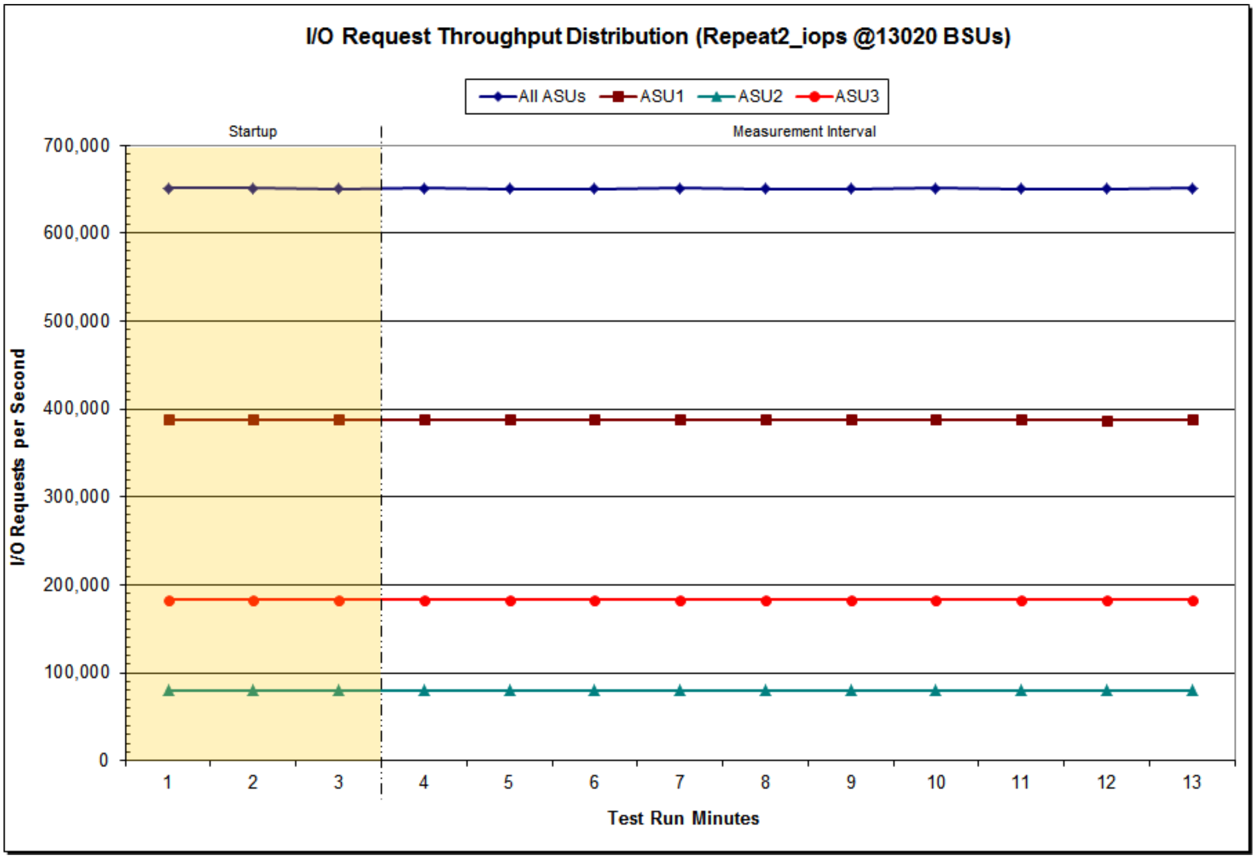
Repeatability 2 LRT –Average Response Time (ms) Distribution Graph



Repeatability 2 IOPS – I/O Request Throughput Distribution Data

13,020 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	3:38:13	3:41:14	0-2	0:03:01
Measurement Interval	3:41:14	3:51:15	3-12	0:10:01
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	651,507.67	388,261.63	80,130.82	183,115.22
1	651,081.45	388,012.22	80,098.48	182,970.75
2	650,980.83	387,933.23	80,081.83	182,965.77
3	651,170.15	388,106.78	80,068.83	182,994.53
4	650,977.58	388,004.52	80,093.97	182,879.10
5	650,959.12	387,989.15	80,061.80	182,908.17
6	651,194.07	387,981.90	80,123.98	183,088.18
7	650,928.88	387,993.20	80,093.73	182,841.95
8	650,892.95	387,940.68	80,022.92	182,929.35
9	651,106.37	388,068.37	80,054.60	182,983.40
10	650,947.52	388,000.13	80,041.02	182,906.37
11	650,701.20	387,817.05	80,004.12	182,880.03
12	651,365.10	388,251.53	80,133.00	182,980.57
Average	***651,024.29***	***388,015.33***	***80,069.80***	***182,939.17***

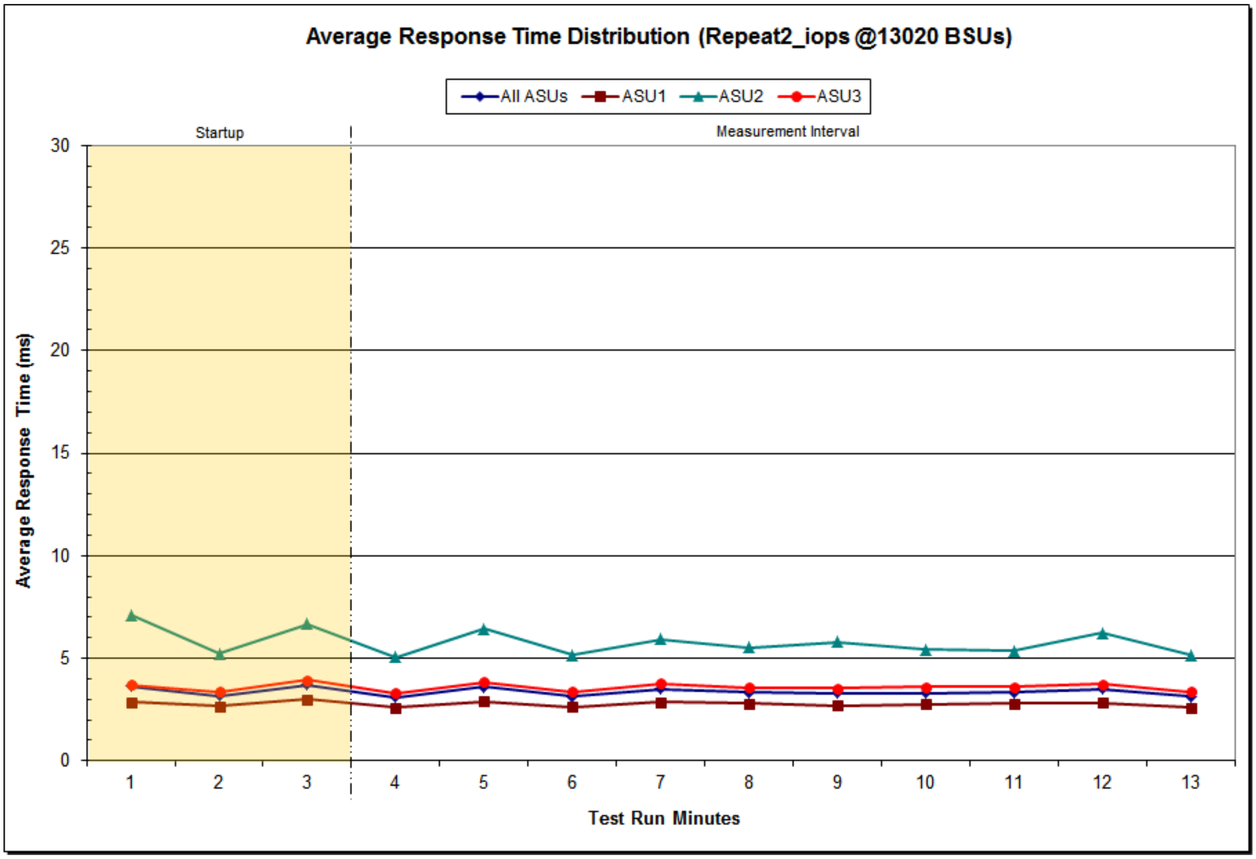
Repeatability 2 IOPS – I/O Request Throughput Distribution Graph



Repeatability 2 IOPS –Average Response Time (ms) Distribution Data

13,020 BSUs	**Start**	**Stop**	**Interval**	**Duration**
Start-Up/Ramp-Up	3:38:13	3:41:14	0-2	0:03:01
Measurement Interval	3:41:14	3:51:15	3-12	0:10:01
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	3.63	2.86	7.12	3.72
1	3.18	2.67	5.22	3.39
2	3.71	2.99	6.68	3.95
3	3.09	2.58	5.06	3.31
4	3.60	2.91	6.45	3.83
5	3.14	2.62	5.14	3.38
6	3.49	2.87	5.92	3.76
7	3.35	2.79	5.53	3.58
8	3.32	2.70	5.80	3.55
9	3.32	2.75	5.43	3.60
10	3.33	2.78	5.35	3.60
11	3.50	2.82	6.24	3.74
12	3.13	2.61	5.14	3.36
Average	***3.33***	***2.74***	***5.61***	***3.57***

Repeatability 2 IOPS –Average Response Time (ms) Distribution Graph



Repeatability 1 (LRT)
Measured Intensity Multiplier and Coefficient of Variation

Clause 3.4.3

IM – Intensity Multiplier: *The ratio of I/Os for each I/O stream relative to the total I/Os for all I/O streams (ASU1-1 – ASU3-1) as required by the benchmark specification.*

Clauses5.1.10 and 5.3.15.2

MIM – Measured Intensity Multiplier: *The Measured Intensity Multiplier represents the ratio of measured I/Os for each I/O stream relative to the total I/Os measured for all I/O streams (ASU1-1 – ASU3-1). This value may differ from the corresponding Expected Intensity Multiplier by no more than 5%.*

Clause 5.3.15.3

COV – Coefficient of Variation: *This measure of variation for the Measured Intensity Multiplier cannot exceed 0.2.*

	ASU1-1	ASU1-2	ASU1-3	ASU1-4	ASU2-1	ASU2-2	ASU2-3	ASU3-1
IM	***0.0350***	***0.2810***	***0.0700***	***0.2100***	***0.0180***	***0.0700***	***0.0350***	***0.2810***
MIM	0.0350	0.2809	0.0700	0.2100	0.0180	0.0700	0.0350	0.2811
COV	0.003	0.001	0.001	0.001	0.003	0.001	0.002	0.001

Repeatability 1 (IOPS)
Measured Intensity Multiplier and Coefficient of Variation

	ASU1-1	ASU1-2	ASU1-3	ASU1-4	ASU2-1	ASU2-2	ASU2-3	ASU3-1
IM	***0.0350***	***0.2810***	***0.0700***	***0.2100***	***0.0180***	***0.0700***	***0.0350***	***0.2810***
MIM	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
COV	0.001	0.000	0.001	0.000	0.001	0.001	0.001	0.000

Repeatability 2 (LRT)
Measured Intensity Multiplier and Coefficient of Variation

	ASU1-1	ASU1-2	ASU1-3	ASU1-4	ASU2-1	ASU2-2	ASU2-3	ASU3-1
IM	***0.0350***	***0.2810***	***0.0700***	***0.2100***	***0.0180***	***0.0700***	***0.0350***	***0.2810***
MIM	0.0350	0.2809	0.0700	0.2099	0.0180	0.0700	0.0349	0.2811
COV	0.002	0.001	0.001	0.001	0.004	0.002	0.003	0.001

Repeatability 2 (IOPS)
Measured Intensity Multiplier and Coefficient of Variation

	ASU1-1	ASU1-2	ASU1-3	ASU1-4	ASU2-1	ASU2-2	ASU2-3	ASU3-1
IM	***0.0350***	***0.2810***	***0.0700***	***0.2100***	***0.0180***	***0.0700***	***0.0350***	***0.2810***
MIM	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
COV	0.001	0.000	0.001	0.000	0.001	0.000	0.001	0.000

Data Persistence Test

Clause 6

The Data Persistence Test demonstrates the Tested Storage Configuration (TSC):

- *Is capable of maintain data integrity across a power cycle.*
- *Ensures the transfer of data between Logical Volumes and host systems occurs without corruption or loss.*

The SPC-1 Workload Generator will write 16 block I/O requests at random over the total Addressable Storage Capacity of the TSC for ten (10) minutes at a minimum of 25% of the load used to generate the SPC-1 IOPS™ primary metric. The bit pattern selected to be written to each block as well as the address of the block will be retained in a log file.

The Tested Storage Configuration (TSC) will be shutdown and restarted using a power off/power on cycle at the end of the above sequence of write operations. In addition, any caches employing battery backup must be flushed/emptied.

The SPC-1 Workload Generator will then use the above log file to verify each block written contains the correct bit pattern.

Clause 9.4.3.8

The following content shall appear in this section of the FDR:

1. *A listing or screen image of all input parameters supplied to the Workload Generator.*
2. *For the successful Data Persistence Test Run, a table illustrating key results. The content, appearance, and format of this table are specified in Table 9-12. Information displayed in this table shall be obtained from the Test Run Results File referenced below in #3.*
3. *For the successful Data Persistence Test Run, the human readable Test Run Results file produced by the Workload Generator (may be contained in an appendix).*

SPC-1 Workload Generator Input Parameters

The SPC-1 Workload Generator input parameters for the Sustainability, IOPS, Response Time Ramp, Repeatability, and Persistence Test Runs are documented in Appendix E: SPC-1 Workload Generator Input Parameters on Page 103.

Data Persistence Test Results File

A link to each test result file generated from each Data Persistence Test is listed below.

Persistence 1 Test Results File

Persistence 2 Test Results File

Data Persistence Test Results

Data Persistence Test Results	
Data Persistence Test Run Number: 1	
Total Number of Logical Blocks Written	3,887,421
Total Number of Logical Blocks Verified	3,852,564
Total Number of Logical Blocks that Failed Verification	0
Time Duration for Writing Test Logical Blocks	5 minutes
Size in bytes of each Logical Block	1024
Number of Failed I/O Requests in the process of the Test	0

If approved by the SPC Auditor, the SPC-2 Persistence Test may be used to meet the SPC-1 persistence requirements. Both the SPC-1 and SPC-2 Persistence Tests provide the same level of functionality and verification of data integrity. The SPC-2 Persistence Test may be easily configured to address an SPC-1 storage configuration. The SPC-2 Persistence Test extends the size of storage configurations that may be tested and significantly reduces the test duration of such configurations.

The SPC-2 Persistence Test was approved for use in this set of audited measurements.

In some cases the same address was the target of multiple writes, which resulted in more Logical Blocks Written than Logical Blocks Verified. In the case of multiple writes to the same address, the pattern written and verified must be associated with the last write to that address.

PRICED STORAGE CONFIGURATION AVAILABILITY DATE

Clause 9.4.3.9

The committed delivery data for general availability (Availability Date) of all products that comprise the Priced Storage Configuration must be reported. When the Priced Storage Configuration includes products or components with different availability dates, the reported Availability Date for the Priced Storage Configuration must be the date at which all components are committed to be available.

The Huawei OceanStor™ 6800 V3 as documented in this Full Disclosure Report is currently available for customer purchase and shipment.

PRICING INFORMATION

Clause 9.4.3.3.6

The Executive Summary shall contain a pricing spreadsheet as documented in Clause 8.3.1.

Pricing information may be found in the Priced Storage Configuration Pricing section on page 17.

TESTED STORAGE CONFIGURATION (TSC) AND PRICED STORAGE CONFIGURATION DIFFERENCES

Clause 9.4.3.3.8

The Executive Summary shall contain a list of all differences between the Tested Storage Configuration (TSC) and the Priced Storage Configuration.

A list of all differences between the Tested Storage Configuration (TSC) and Priced Storage Configuration may be found in the Executive Summary portion of this document on page 17.

ANOMALIES OR IRREGULARITIES

Clause 9.4.3.10

The FDR shall include a clear and complete description of any anomalies or irregularities encountered in the course of executing the SPC-1 benchmark that may in any way call into question the accuracy, verifiability, or authenticity of information published in this FDR.

The SPC-2 Persistence Test Run 1 *(write phase)* specified 410 Streams rather than the required minimum of 434 Streams. That small difference in the number of Streams did not materially affect the results of the Persistence Test.

APPENDIX A: SPC-1 GLOSSARY

"Decimal" *(powers of ten)* Measurement Units

In the storage industry, the terms "kilo", "mega", "giga", "tera", "peta", and "exa" are commonly used prefixes for computing performance and capacity. For the purposes of the SPC workload definitions, all of the following terms are defined in "powers of ten" measurement units.

A kilobyte (KB) is equal to 1,000 (10^3) bytes.

A megabyte (MB) is equal to 1,000,000 (10^6) bytes.

A gigabyte (GB) is equal to 1,000,000,000 (10^9) bytes.

A terabyte (TB) is equal to 1,000,000,000,000 (10^{12}) bytes.

A petabyte (PB) is equal to 1,000,000,000,000,000 (10^{15}) bytes

An exabyte (EB) is equal to 1,000,000,000,000,000,000 (10^{18}) bytes

"Binary" *(powers of two)* Measurement Units

The sizes reported by many operating system components use "powers of two" measurement units rather than "power of ten" units. The following standardized definitions and terms are also valid and may be used in this document.

A kibibyte (KiB) is equal to 1,024 (2^{10}) bytes.

A mebibyte (MiB) is equal to 1,048,576 (2^{20}) bytes.

A gigibyte (GiB) is equal to 1,073,741,824 (2^{30}) bytes.

A tebibyte (TiB) is equal to 1,099,511,627,776 (2^{40}) bytes.

A pebibyte (PiB) is equal to 1,125,899,906,842,624 (2^{50}) bytes.

An exbibyte (EiB) is equal to 1,152,921,504,606,846,967 (2^{60}) bytes.

SPC-1 Data Repository Definitions

Total ASU Capacity: The total storage capacity read and written in the course of executing the SPC-1 benchmark.

Application Storage Unit (ASU): The logical interface between the storage and SPC-1 Workload Generator. The three ASUs (Data, User, and Log) are typically implemented on one or more Logical Volume.

Logical Volume: The division of Addressable Storage Capacity into individually addressable logical units of storage used in the SPC-1 benchmark. Each Logical Volume is implemented as a single, contiguous address space.

Addressable Storage Capacity: The total storage (sum of Logical Volumes) that can be read and written by application programs such as the SPC-1 Workload Generator.

Configured Storage Capacity: This capacity includes the Addressable Storage Capacity and any other storage (parity disks, hot spares, etc.) necessary to implement the Addressable Storage Capacity.

Physical Storage Capacity: The formatted capacity of all storage devices physically present in the Tested Storage Configuration (TSC).

Data Protection Overhead: The storage capacity required to implement the selected level of data protection.

Required Storage: The amount of Configured Storage Capacity required to implement the Addressable Storage Configuration, excluding the storage required for the three ASUs.

Global Storage Overhead: The amount of Physical Storage Capacity that is required for storage subsystem use and unavailable for use by application programs.

Total Unused Storage: The amount of storage capacity available for use by application programs but not included in the Total ASU Capacity.

SPC-1 Data Protection Levels

Protected 1: The single point of failure of any ***storage device*** in the configuration will not result in permanent loss of access to or integrity of the SPC-1 Data Repository.

Protected 2: The single point of failure of any ***component*** in the configuration will not result in permanent loss of access to or integrity of the SPC-1 Data Repository.

SPC-1 Test Execution Definitions

Average Response Time: The sum of the Response Times for all Measured I/O Requests divided by the total number of Measured I/O Requests.

Completed I/O Request: An I/O Request with a Start Time and a Completion Time (see "I/O Completion Types" below).

Completion Time: The time recorded by the Workload Generator when an I/O Request is satisfied by the TSC as signaled by System Software.

Data Rate: The data transferred in all Measured I/O Requests in an SPC-1 Test Run divided by the length of the Test Run in seconds.

Expected I/O Count: For any given I/O Stream and Test Phase, the product of 50 times the BSU level, the duration of the Test Phase in seconds, and the Intensity Multiplier for that I/O Stream.

Failed I/O Request: Any I/O Request issued by the Workload Generator that could not be completed or was signaled as failed by System Software. A Failed I/O Request has no Completion Time (see "I/O Completion Types" below).

I/O Request Throughput: The total number of Measured I/O requests in an SPC-1 Test Run divided by the duration of the Measurement Interval in seconds.

In-Flight I/O Request: An I/O Request issued by the I/O Command Generator to the TSC that has a recorded Start Time, but does not complete within the Measurement Interval (see "I/O Completion Types" below).

Measured I/O Request: A Completed I/O Request with a Completion Time occurring within the Measurement Interval (see "I/O Completion Types" below).

Measured Intensity Multiplier: The percentage of all Measured I/O Requests that were issued by a given I/O Stream.

Measurement Interval: The finite and contiguous time period, after the TSC has reached Steady State, when data is collected by a Test Sponsor to generate an SPC-1 test result or support an SPC-1 test result.

Ramp-Up: The time required for the Benchmark Configuration (BC) to produce Steady State throughput after the Workload Generator begins submitting I/O Requests to the TSC for execution.

Ramp-Down: The time required for the BC to complete all I/O Requests issued by the Workload Generator. The Ramp-Down period begins when the Workload Generator ceases to issue new I/O Requests to the TSC.

Response Time: The Response Time of a Measured I/O Request is its Completion Time minus its Start Time.

Start Time: The time recorded by the Workload Generator when an I/O Request is submitted, by the Workload Generator, to the System Software for execution on the Tested Storage Configuration (TSC).

Start-Up: The period that begins after the Workload Generator starts to submit I/O requests to the TSC and ends at the beginning of the Measurement Interval.

Shut-Down: The period between the end of the Measurement Interval and the time when all I/O Requests issued by the Workload Generator have completed or failed.

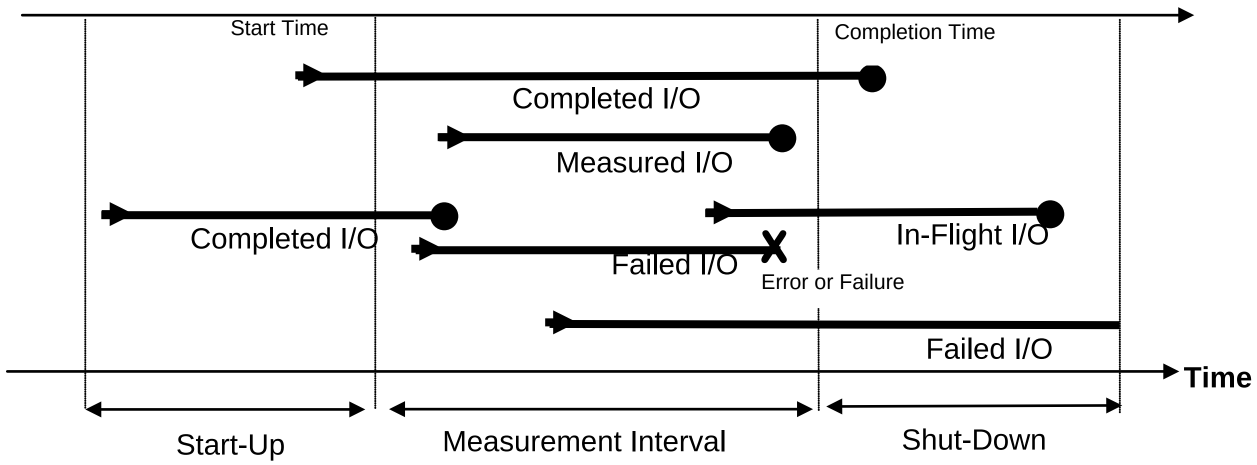
Steady State: The consistent and sustainable throughput of the TSC. During this period the load presented to the TSC by the Workload Generator is constant.

Test: A collection of Test Phases and or Test Runs sharing a common objective.

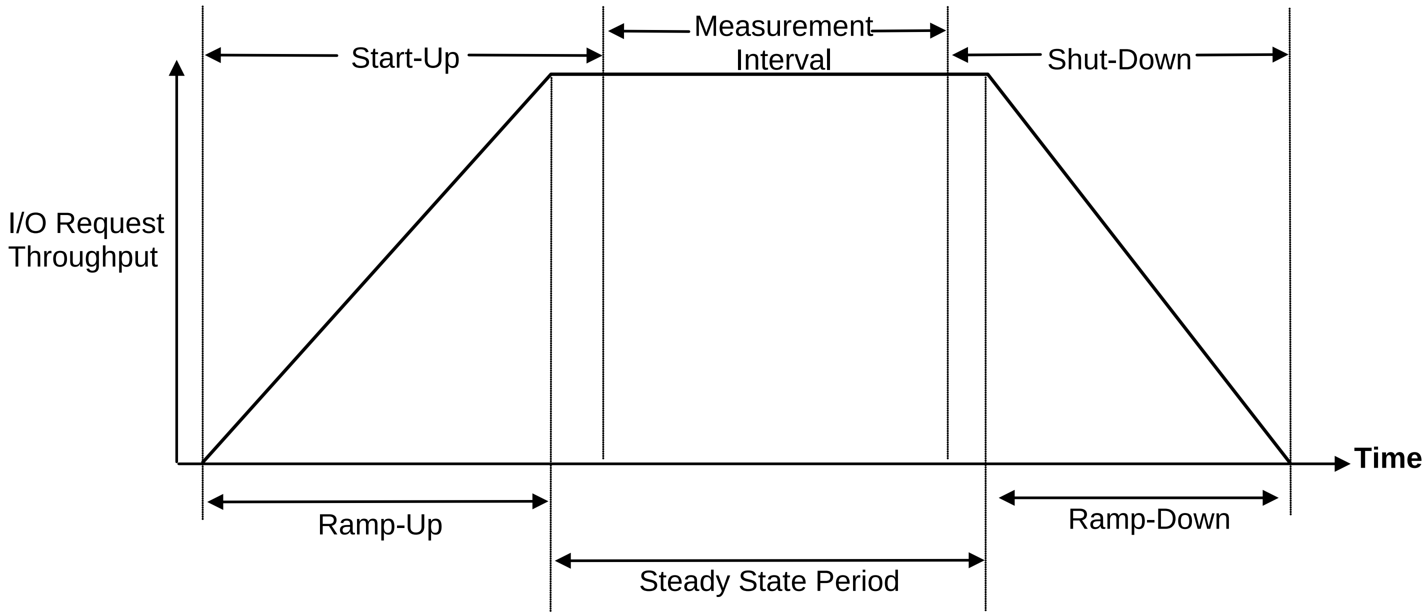
Test Run: The execution of SPC-1 for the purpose of producing or supporting an SPC-1 test result. SPC-1 Test Runs may have a finite and measured Ramp-Up period, Start-Up period, Shut-Down period, and Ramp-Down period as illustrated in the "SPC-1 Test Run Components" below. All SPC-1 Test Runs shall have a Steady State period and a Measurement Interval.

Test Phase: A collection of one or more SPC-1 Test Runs sharing a common objective and intended to be run in a specific sequence.

I/O Completion Types



SPC-1 Test Run Components



APPENDIX B: CUSTOMER TUNABLE PARAMETERS AND OPTIONS

Red Hat Enterprise Linux 5.5 (64-bit)

Change the I/O scheduler from ***cfq*** to ***noop*** on each Host System, which will result in all incoming I/O requests inserted into a simple, unordered FIFO queue. This change was done by the execution of the **scheduler.sh** script as documented in *Appendix C: Tested Storage Configuration (TSC) Creation*.

Huawei FusionServer RH2288 V2 Host Systems

The CPU frequency scaling policy was changed on each Host System from ***ondemand*** to ***performance***, which prevents a reduction in the CPU frequency when a server is lightly loaded. This change was done by the execution of the **chg_cpu.sh** script as documented in *Appendix C: Tested Storage Configuration (TSC) Creation*.

APPENDIX C: TESTED STORAGE CONFIGURATION (TSC) CREATION

The scripts referenced in Steps 2 and 3 appear in the section, Referenced Scripts.

Step 1: Create Mapping View, LUN Group, Host Group and Host

Execute the following commands using the OceanStor 6800 V3 CLI from one of the Host Systems to complete the following:

- Create one ***mapping_view*** (***MappingView1***)
- Create one ***lun_group*** (***LunGroup1***)
- Create one ***host_group*** (***HostGroup1***)
- Create one ***host*** (***Host1***)
- Add ***Host1*** to ***HostGroup1***
- Add ***HostGroup1*** and ***LunGroup1*** to ***MappingView1***
- Add the FC ports' WWN to ***Host1***

```
create mapping_view name=MappingView1 mapping_view_id=1
create lun_group name=LunGroup1 lun_group_id=1
create host_group name=HostGroup1 host_group_id=1
create host name=Host1 operating_system=Linux host_id=1

add host_group host host_group_id=1 host_id_list=1
add mapping_view host_group mapping_view_id=1 host_group_id=1
add mapping_view lun_group mapping_view_id=1 lun_group_id=1

add host initiator host_id=1 initiator_type=FC wwn=21000024ff2c952a
add host initiator host_id=1 initiator_type=FC wwn=21000024ff2c952b
add host initiator host_id=1 initiator_type=FC wwn=21000024ff35e7da
add host initiator host_id=1 initiator_type=FC wwn=21000024ff35e7db
add host initiator host_id=1 initiator_type=FC wwn=21000024ff36e6e2
add host initiator host_id=1 initiator_type=FC wwn=21000024ff36e6e3
add host initiator host_id=1 initiator_type=FC wwn=21000024ff372908
add host initiator host_id=1 initiator_type=FC wwn=21000024ff372909
add host initiator host_id=1 initiator_type=FC wwn=21000024ff455e92
add host initiator host_id=1 initiator_type=FC wwn=21000024ff455e93
add host initiator host_id=1 initiator_type=FC wwn=21000024ff49af3a
add host initiator host_id=1 initiator_type=FC wwn=21000024ff49af3b
add host initiator host_id=1 initiator_type=FC wwn=21000024ff49af78
add host initiator host_id=1 initiator_type=FC wwn=21000024ff49af79
add host initiator host_id=1 initiator_type=FC wwn=21000024ff4a4d56
add host initiator host_id=1 initiator_type=FC wwn=21000024ff4a4d57
add host initiator host_id=1 initiator_type=FC wwn=21000024ff536922
add host initiator host_id=1 initiator_type=FC wwn=21000024ff536923
add host initiator host_id=1 initiator_type=FC wwn=21000024ff536a2c
add host initiator host_id=1 initiator_type=FC wwn=21000024ff536a2d
add host initiator host_id=1 initiator_type=FC wwn=21000024ff540b8e
add host initiator host_id=1 initiator_type=FC wwn=21000024ff540b8f
add host initiator host_id=1 initiator_type=FC wwn=21000024ff543b04
add host initiator host_id=1 initiator_type=FC wwn=21000024ff543b05
```

Step 2: Create Disk Domains, Storage Pools, LUNs and Deploy SSDs

Execute the **mklun.sh** script on one of the Host Systems, which has **expect** installed to complete the following:

- Create 48 disk domains
- Change the disk domains' HDD (tier1) hot spare strategy from **High** to **Low**
- Create 48 storage pools
  (one storage pool per disk domain using all available capacity)
- Create 144 LUNs
  (4 LUNs in storage pools 0-7, 8-15, 24-31, 32-39 and 1 LUN in the remaining 16 storage pools)
- Add the SSDs to sixteen disk domains (0-7, 24-31) for the SmartTier functionality
- Change the disk domains' SSD (tier0) hot spare strategy from **High** to **Low**
- Add the SSDs to sixteen storage pools (0-7, 24-31) for the SmartTier functionality
- Add the 144 LUNs to ***LunGroup1***

*Note: **Expect** is a Unix automation and testing tool, written by Don Libes as an extension to the Tcl scripting language, for interactive applications such as telnet, ftp, passwd, fsck, rlogin, tip, ssh, and others. It uses Unix pseudo terminals to wrap up subprocesses transparently, allowing the automation of arbitrary applications that are accessed over a terminal. Expect is an open source tool can be downloaded at the following location: http://www.nist.gov/el/msid/expect.cfm*

Step 3: Create Volumes on the Master Host System

Execute the **mkvolume.sh** script on the Master Host System to create 144 logical volumes as follows:

1. **Create Physical Volume**

   Create 144 physical volumes using the **pvcreate** command.

2. **Create Volumes Groups**

   Create three volume groups (**vg1**, **vg2** and **vg3**) using the **vgcreate** command as follows:

   Create **vg1** using the following physical volumes: /dev/sdb, /dev/sdc, /dev/sdd, /dev/sde, /dev/sdf, /dev/sdg, /dev/sdh, /dev/sdi, /dev/sdj, /dev/sdk, /dev/sdl, /dev/sdm, /dev/sdn, /dev/sdo, /dev/sdp, /dev/sdq, /dev/sdr, /dev/sds, /dev/sdt, /dev/sdu, /dev/sdv, /dev/sdw, /dev/sdx, /dev/sdy, /dev/sdz, /dev/sdaa, /dev/sdab, /dev/sdac, /dev/sdad, /dev/sdae, /dev/sdaf, /dev/sdag, /dev/sdah, /dev/sdai, /dev/sdaj, /dev/sdak, /dev/sdal, /dev/sdam, /dev/sdan, /dev/sdao, /dev/sdap, /dev/sdaq, /dev/sdar, /dev/sdas, /dev/sdat, /dev/sdau, /dev/sdav, /dev/sdaw, /dev/sdax, /dev/sday, /dev/sdaz, /dev/sdba, /dev/sdbb, /dev/sdbc, /dev/sdbd, /dev/sdbe, /dev/sdbf, /dev/sdbg, /dev/sdbh, /dev/sdbi, /dev/sdbj, /dev/sdbk, /dev/sdbl, /dev/sdbm

   Create **vg2** using the following physical volumes: /dev/sdbn, /dev/sdbo, /dev/sdbp, /dev/sdbq, /dev/sdbr, /dev/sdbs, /dev/sdbt, /dev/sdbu, /dev/sdbv, /dev/sdbw, /dev/sdbx, /dev/sdby, /dev/sdbz, /dev/sdca, /dev/sdcb, /dev/sdcc, /dev/sdcd, /dev/sdce, /dev/sdcf, /dev/sdcg, /dev/sdch, /dev/sdci, /dev/sdcj, /dev/sdck, /dev/sdcl, /dev/sdcm, /dev/sdcn, /dev/sdco, /dev/sdcp, /dev/sdcq, /dev/sdcr, /dev/sdcs, /dev/sdct, /dev/sdcu, /dev/sdcv, /dev/sdcw, /dev/sdcx, /dev/sdcy, /dev/sdcz,

/dev/sdda, /dev/sddb, /dev/sddc, /dev/sddd, /dev/sdde, /dev/sddf, /dev/sddg, /dev/sddh, /dev/sddi, /dev/sddj, /dev/sddk, /dev/sddl, /dev/sddm, /dev/sddn, /dev/sddo, /dev/sddp, /dev/sddq, /dev/sddr, /dev/sdds, /dev/sddt, /dev/sddu, /dev/sddv, /dev/sddw, /dev/sddx, /dev/sddy

Create **vg3** using the following physical volumes: /dev/sddz, /dev/sdea, /dev/sdeb, /dev/sdec, /dev/sded, /dev/sdee, /dev/sdef, /dev/sdeg, /dev/sdeh, /dev/sdei, /dev/sdej, /dev/sdek, /dev/sdel, /dev/sdem, /dev/sden, /dev/sdeo

3. **Create Logical Volumes**
   - Create 64 logical volumes, each with a capacity of 1,575 GiB, on **vg1** for ASU-1.
   - Create 64 logical volumes, each with a capacity of 1,575 GiB, on **vg2** for ASU-2.
   - Create 16 logical volumes, each with a capacity of 1,400 GiB, on **vg3** for ASU-3.
4. **Scan Physical Volumes, Volume Groups, Logical Volumes and activate each Logical Volume**

   Execute the **lv_scan.sh** on the Slave Host Systems to scan the physical volumes, volume groups and logical volumes. In addition, the script will make each logical volume available *(activate)*.

Step 3: Change the Scheduler on each Host System

Execute the **scheduler.sh** script on each Host System to change the scheduler of each block device from ***cfq*** to ***noop***.

Step 4: Change the CPU Frequency Scaling Policy

Execute the **chg_cpu.sh** script on each Host System to change the CPU frequency scaling policy from ***ondemand*** to ***performance***, which prevents a reduction in the CPU frequency when a server is lightly loaded.

Referenced Scripts

mklun.sh

```
#!/bin/bash

stor=100.124.7.101
stor_user=admin
stor_pswd=Admin@storage0

export LANG=C

echo "creating LUN ..."

expect <<__END_CREATE_LUN
     spawn ssh $stor_user@$stor
     set timeout 60
     expect {
            -re "assword" { send "$stor_pswd\r" }
            -re "yes/no" { send "yes\r"; exp_continue }
     }
     expect ">"

            # ------create disk_domain--------
```

```
                send "create disk_domain name=ASU010 disk_list=DAE000.0-24,DAE030.0-
4 disk_domain_id=0\r"
                expect ">"

                send "create disk_domain name=ASU011 disk_list=DAE001.0-24,DAE030.5-
9 disk_domain_id=1\r"
                expect ">"

                send "create disk_domain name=ASU012 disk_list=DAE002.0-
24,DAE030.10-14 disk_domain_id=2\r"
                expect ">"

                send "create disk_domain name=ASU013 disk_list=DAE010.0-
24,DAE030.15-19 disk_domain_id=3\r"
                expect ">"

                send "create disk_domain name=ASU014 disk_list=DAE011.0-
24,DAE030.20-24 disk_domain_id=4\r"
                expect ">"

                send "create disk_domain name=ASU015 disk_list=DAE012.0-24,DAE031.0-
4 disk_domain_id=5\r"
                expect ">"

                send "create disk_domain name=ASU016 disk_list=DAE020.0-24,DAE031.5-
9 disk_domain_id=6\r"
                expect ">"

                send "create disk_domain name=ASU017 disk_list=DAE021.0-
24,DAE031.10-14 disk_domain_id=7\r"
                expect ">"



                send "create disk_domain name=ASU020 disk_list=DAE040.0-24,DAE070.0-
6 disk_domain_id=8\r"
                expect ">"

                send "create disk_domain name=ASU021 disk_list=DAE041.0-24,DAE070.7-
13 disk_domain_id=9\r"
                expect ">"

                send "create disk_domain name=ASU022 disk_list=DAE042.0-
24,DAE070.14-20 disk_domain_id=10\r"
                expect ">"

                send "create disk_domain name=ASU023 disk_list=DAE050.0-24,DAE071.0-
6 disk_domain_id=11\r"
                expect ">"

                send "create disk_domain name=ASU024 disk_list=DAE051.0-24,DAE071.7-
13 disk_domain_id=12\r"
                expect ">"

                send "create disk_domain name=ASU025 disk_list=DAE052.0-
24,DAE071.14-20 disk_domain_id=13\r"
                expect ">"

                send "create disk_domain name=ASU026 disk_list=DAE060.0-
24,DAE070.21-24,DAE071.21-23 disk_domain_id=14\r"
                expect ">"
                send "create disk_domain name=ASU027 disk_list=DAE061.0-
24,DAE071.24,DAE031.15-20 disk_domain_id=15\r"

                expect ">"


                send "create disk_domain name=ASU030 disk_list=DAE080.0-11
disk_domain_id=16\r"
                expect ">"

                send "create disk_domain name=ASU031 disk_list=DAE080.12-23
disk_domain_id=17\r"
                expect ">"

                send "create disk_domain name=ASU032 disk_list=DAE081.0-11
disk_domain_id=18\r"
                expect ">"

                send "create disk_domain name=ASU033 disk_list=DAE081.12-23
disk_domain_id=19\r"
                expect ">"

                send "create disk_domain name=ASU034 disk_list=DAE082.0-11
disk_domain_id=20\r"
                expect ">"

                send "create disk_domain name=ASU035 disk_list=DAE082.12-23
disk_domain_id=21\r"
                expect ">"

                send "create disk_domain name=ASU036 disk_list=DAE090.0-11
disk_domain_id=22\r"
                expect ">"

                send "create disk_domain name=ASU037 disk_list=DAE090.12-
16,DAE082.24,DAE081.24,DAE080.24,DAE031.21-24 disk_domain_id=23\r"
            expect ">"


            send "create disk_domain name=ASU110 disk_list=DAE0140.0-24,DAE0170.0-4
disk_domain_id=24\r"
            expect ">"

                send "create disk_domain name=ASU111 disk_list=DAE0141.0-
24,DAE0170.5-9 disk_domain_id=25\r"
                expect ">"

                send "create disk_domain name=ASU112 disk_list=DAE0142.0-
24,DAE0170.10-14 disk_domain_id=26\r"

                expect ">"
                send "create disk_domain name=ASU113 disk_list=DAE0150.0-
24,DAE0170.15-19 disk_domain_id=27\r"

                expect ">"
                send "create disk_domain name=ASU114 disk_list=DAE0151.0-
24,DAE0170.20-24 disk_domain_id=28\r"

                expect ">"
                send "create disk_domain name=ASU115 disk_list=DAE0152.0-
24,DAE0171.0-4 disk_domain_id=29\r"

                expect ">"
                send "create disk_domain name=ASU116 disk_list=DAE0160.0-
24,DAE0171.5-9 disk_domain_id=30\r"

                expect ">"
                send "create disk_domain name=ASU117 disk_list=DAE0161.0-
24,DAE0171.10-14 disk_domain_id=31\r"

                expect ">"


                send "create disk_domain name=ASU120 disk_list=DAE0180.0-
24,DAE01B0.0-6 disk_domain_id=32\r"

                expect ">"
                send "create disk_domain name=ASU121 disk_list=DAE0181.0-
24,DAE01B0.7-13 disk_domain_id=33\r"

                expect ">"
                send "create disk_domain name=ASU122 disk_list=DAE0182.0-
24,DAE01B0.14-20 disk_domain_id=34\r"

                expect ">"
                send "create disk_domain name=ASU123 disk_list=DAE0190.0-
24,DAE01B1.0-6 disk_domain_id=35\r"

                expect ">"
                send "create disk_domain name=ASU124 disk_list=DAE0191.0-
24,DAE01B1.7-13 disk_domain_id=36\r"
                expect ">"
                send "create disk_domain name=ASU125 disk_list=DAE0192.0-
24,DAE01B1.14-20 disk_domain_id=37\r"

                expect ">"
                send "create disk_domain name=ASU126 disk_list=DAE01A0.0-
24,DAE01B0.21-24,DAE01B1.21-23 disk_domain_id=38\r"

                expect ">"
                send "create disk_domain name=ASU127 disk_list=DAE01A1.0-
24,DAE01B1.24,DAE0171.15-20 disk_domain_id=39\r"

                expect ">"


                send "create disk_domain name=ASU130 disk_list=DAE01C0.0-11
disk_domain_id=40\r"

                expect ">"
                send "create disk_domain name=ASU131 disk_list=DAE01C0.12-23
disk_domain_id=41\r"

                expect ">"
                send "create disk_domain name=ASU132 disk_list=DAE01C1.0-11
disk_domain_id=42\r"

                expect ">"
                send "create disk_domain name=ASU133 disk_list=DAE01C1.12-23
disk_domain_id=43\r"

                expect ">"
                send "create disk_domain name=ASU134 disk_list=DAE01C2.0-11
disk_domain_id=44\r"

                expect ">"
                send "create disk_domain name=ASU135 disk_list=DAE01C2.12-23
disk_domain_id=45\r"

                expect ">"
                send "create disk_domain name=ASU136 disk_list=DAE01D0.0-11
disk_domain_id=46\r"

                expect ">"
                send "create disk_domain name=ASU137 disk_list=DAE01D0.12-
16,DAE01C2.24,DAE01C1.24,DAE01C0.24,DAE0171.21-24 disk_domain_id=47\r"
                expect ">"



            # ------change disk_domain tier1_hotspare_strategy from high to low -----
---
            for { set domainid 0 } { \$domainid <= 47 } { incr domainid } {
            send "change disk_domain general disk_domain_id=\$domianid
tier1_hotspare_strategy=low\r"
            expect "y/n";send "y\r"
            expect ">"
            }


            # ------create storage_pool --------
                send "create storage_pool name=ASU010 disk_type=SAS capacity=7162GB
pool_id=0 disk_domain_id=0 raid_level=RAID10\r"
                expect ">"

                send "create storage_pool name=ASU110 disk_type=SAS capacity=7162GB
pool_id=24 disk_domain_id=24 raid_level=RAID10\r"
                expect ">"

                for { set poolid 1 } { \$poolid <= 7 } { incr poolid } {

                send "create storage_pool name=ASU01\$poolid disk_type=SAS
capacity=7171GB pool_id=\$poolid disk_domain_id=\$poolid raid_level=RAID10\r"
                expect ">"
                send "create storage_pool name=ASU11\$poolid disk_type=SAS
capacity=7171GB pool_id=[expr \$poolid + 24] disk_domain_id=[expr \$poolid + 24]
raid_level=RAID10\r"
                expect ">"
                }

                for { set poolid 0 } { \$poolid <= 7 } { incr poolid } {


                send "create storage_pool name=ASU02\$poolid disk_type=SAS
capacity=7684GB pool_id=[expr \$poolid + 8] disk_domain_id=[expr \$poolid + 8]
raid_level=RAID10\r"

                expect ">"
                send "create storage_pool name=ASU12\$poolid disk_type=SAS
capacity=7684GB pool_id=[expr \$poolid + 32] disk_domain_id=[expr \$poolid + 32]
raid_level=RAID10\r"

                expect ">"


                send "create storage_pool name=ASU03\$poolid disk disk_type=SAS
capacity=2816GB pool_id=[expr \$poolid + 16 disk_domain_id=[expr \$poolid + 16]
raid_level=RAID10\r"
```

```
                expect ">"
                send "create storage_pool name=ASU13\$poolid disk disk_type=SAS
capacity=2816GB pool_id=[expr \$poolid + 40 disk_domain_id=[expr \$poolid + 40]
raid_level=RAID10\r"
                expect ">"
                }



            # ------create lun --------
            send "create lun name=ASU010 number=4 pool_id=0 capacity=1790GB
owner_controller=0A\r"
            expect ">"

                send "create lun name=ASU011 number=4 pool_id=1 capacity=1792GB
owner_controller=0B\r"

                expect ">"
                send "create lun name=ASU012 number=4 pool_id=2 capacity=1792GB
owner_controller=0A\r"

                expect ">"
                send "create lun name=ASU013 number=4 pool_id=3 capacity=1792GB
owner_controller=0B\r"

                expect ">"
                send "create lun name=ASU014 number=4 pool_id=4 capacity=1792GB
owner_controller=0A\r"

                expect ">"
                send "create lun name=ASU015 number=4 pool_id=5 capacity=1792GB
owner_controller=0B\r"

                expect ">"
                send "create lun name=ASU016 number=4 pool_id=6 capacity=1792GB
owner_controller=0A\r"

                expect ">"
                send "create lun name=ASU017 number=4 pool_id=7 capacity=1792GB
owner_controller=0B\r"

                expect ">"


                send "create lun name=ASU020 number=4 pool_id=8 capacity=1921GB
owner_controller=0A\r"

                expect ">"
                send "create lun name=ASU021 number=4 pool_id=9 capacity=1921GB
owner_controller=0B\r"

                expect ">"
                send "create lun name=ASU022 number=4 pool_id=10 capacity=1921GB
owner_controller=0A\r"

                expect ">"
                send "create lun name=ASU023 number=4 pool_id=11 capacity=1921GB
owner_controller=0B\r"

                expect ">"
                send "create lun name=ASU024 number=4 pool_id=12 capacity=1921GB
owner_controller=0A\r"
```

```
                 expect ">"
                 send "create lun name=ASU025 number=4 pool_id=13 capacity=1921GB
 owner_controller=0B\r"

                 expect ">"
                 send "create lun name=ASU026 number=4 pool_id=14 capacity=1921GB
 owner_controller=0A\r"

                 expect ">"
                 send "create lun name=ASU027 number=4 pool_id=15 capacity=1921GB
 owner_controller=0B\r"

                 expect ">"


                 send "create lun name=ASU030 pool_id=16 capacity=2816GB
 owner_controller=0A\r"

                 expect ">"
                 send "create lun name=ASU031 pool_id=17 capacity=2816GB
 owner_controller=0B\r"

                 expect ">"
                 send "create lun name=ASU032 pool_id=18 capacity=2816GB
 owner_controller=0A\r"

                 expect ">"
                 send "create lun name=ASU033 pool_id=19 capacity=2816GB
 owner_controller=0B\r"

                 expect ">"
                 send "create lun name=ASU034 pool_id=20 capacity=2816GB
 owner_controller=0A\r"

                 expect ">"
                 send "create lun name=ASU035 pool_id=21 capacity=2816GB
 owner_controller=0B\r"

                 expect ">"
                 send "create lun name=ASU036 pool_id=22 capacity=2816GB
 owner_controller=0A\r"

                 expect ">"
                 send "create lun name=ASU037 pool_id=23 capacity=2816GB
 owner_controller=0B\r"
                 expect ">"

                 send "create lun name=ASU110 number=4 pool_id=24 capacity=1790GB
 owner_controller=0C\r"

                 expect ">"
                 send "create lun name=ASU111 number=4 pool_id=25 capacity=1792GB
 owner_controller=0D\r"

                 expect ">"
                 send "create lun name=ASU112 number=4 pool_id=26 capacity=1792GB
 owner_controller=0C\r"

                 expect ">"
                 send "create lun name=ASU113 number=4 pool_id=27 capacity=1792GB
 owner_controller=0D\r"

                 expect ">"
```

```
                send "create lun name=ASU114 number=4 pool_id=28 capacity=1792GB
 owner_controller=0C\r"

                expect ">"
                send "create lun name=ASU115 number=4 pool_id=29 capacity=1792GB
 owner_controller=0D\r"

                expect ">"
                send "create lun name=ASU116 number=4 pool_id=30 capacity=1792GB
 owner_controller=0C\r"

                expect ">"
                send "create lun name=ASU117 number=4 pool_id=31 capacity=1792GB
 owner_controller=0D\r"

                expect ">"


                send "create lun name=ASU120 number=4 pool_id=32 capacity=1921GB
 owner_controller=0C\r"

                expect ">"
                send "create lun name=ASU121 number=4 pool_id=33 capacity=1921GB
 owner_controller=0D\r"

                expect ">"
                send "create lun name=ASU122 number=4 pool_id=34 capacity=1921GB
 owner_controller=0C\r"

                expect ">"
                send "create lun name=ASU123 number=4 pool_id=35 capacity=1921GB
 owner_controller=0D\r"

                expect ">"
                send "create lun name=ASU124 number=4 pool_id=36 capacity=1921GB
 owner_controller=0C\r"

                expect ">"
                send "create lun name=ASU125 number=4 pool_id=37 capacity=1921GB
 owner_controller=0D\r"

                expect ">"
                send "create lun name=ASU126 number=4 pool_id=38 capacity=1921GB
 owner_controller=0C\r"

                expect ">"
                send "create lun name=ASU127 number=4 pool_id=39 capacity=1921GB
 owner_controller=0D\r"

                expect ">"


                send "create lun name=ASU130 pool_id=40 capacity=2816GB
 owner_controller=0C\r"

                expect ">"
                send "create lun name=ASU131 pool_id=41 capacity=2816GB
 owner_controller=0D\r"

                expect ">"
                send "create lun name=ASU132 pool_id=42 capacity=2816GB
 owner_controller=0C\r"
```

```
                expect ">"
                send "create lun name=ASU133 pool_id=43 capacity=2816GB
 owner_controller=0D\r"

                expect ">"
                send "create lun name=ASU134 pool_id=44 capacity=2816GB
 owner_controller=0C\r"

                expect ">"
                send "create lun name=ASU135 pool_id=45 capacity=2816GB
 owner_controller=0D\r"

                expect ">"
                send "create lun name=ASU136 pool_id=46 capacity=2816GB
 owner_controller=0C\r"

                expect ">"
                send "create lun name=ASU137 pool_id=47 capacity=2816GB
 owner_controller=0D\r"
                expect ">"



            # ------add ssds to disk_domain--------
            send "add disk_domain disk disk_domain_id=0 disk_list=DAE022.0-9\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=1 disk_list=DAE022.10-
 19\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=2 disk_list=DAE032.0-9\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=3 disk_list=DAE032.10-
 19\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=4 disk_list=DAE062.0-9\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=5 disk_list=DAE062.10-
 19\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=6 disk_list=DAE072.0-9\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=7 disk_list=DAE072.10-
 19\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=24 disk_list=DAE0162.0-
 9\r"
```

```
                expect ">"

                send "add disk_domain disk disk_domain_id=25 disk_list=DAE0162.10-
19\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=26 disk_list=DAE0172.0-
9\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=27 disk_list=DAE0172.10-
19\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=28 disk_list=DAE01A2.0-
9\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=29 disk_list=DAE01A2.10-
19\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=30 disk_list=DAE01B2.0-
9\r"

                expect ">"

                send "add disk_domain disk disk_domain_id=31 disk_list=DAE01B2.10-
19\r"

                expect ">"


                #--------change disk_domain tier0_hotspare_strategy from high to
low-----

                for { set domainid 0 } { \$domainid <= 7 } { incr domainid } {

                send "change disk_domain general disk_domain_id=\$domainid
tier0_hotspare_strategy=low\r"
                expect ">"
                send "change disk_domain general disk_domain_id=[expr \$domainid +
24] tier0_hotspare_strategy=low\r"
                expect ">"
                }


                # ------add ssds to storage_pool-------
                for { set poolid 0 } { \$poolid <= 7 } { incr pool } {

                send "add storage_pool tier pool_id=\$poolid disk_type=SSD
capacity=781GB raid_level=RAID10\r"
                expect ">"
                send "add storage_pool tier pool_id=[expr \$poolid + 24]
disk_type=SSD capacity=781GB raid_level=RAID10\r"
                expect ">"
                }
```

```
            # ------ add all luns to lun_group--------
               send "add lun_group lun lun_group_id=1
lun_id_list=0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,2
7,28,29,30,31\r"
               expect ">"

               sedn "add lun_group lun lun_group_id=1
lun_id_list=72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,
96,97,98,99,100,101,102,103\r"
               expect ">"

               send "add lun_group lun lun_group_id=1
lun_id_list=32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,
56,57,58,59,60,61,62,63\r"
               expect ">"

               send "add lun_group lun lun_group_id=1
lun_id_list=104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,
122,123,124,125,126,127,128,129,130,131,132,133,134,135\r"
               expect ">"

               send "add lun_group lun lun_group_id=1
lun_id_list=64,65,66,67,68,69,70,71,136,137,138,139,140,141,142,143\r"
            expect ">"


            send "exit\r"
            expect "(y/n):"
            send "y\r"
            expect EOF
__END_CREATE_LUN
```

lv_scan.sh

```
#!/bin/bash
pvscan
vgscan
lvscan
lvchange -a y /dev/vg1/asu100
lvchange -a y /dev/vg1/asu101
lvchange -a y /dev/vg1/asu102
lvchange -a y /dev/vg1/asu103
lvchange -a y /dev/vg1/asu104
lvchange -a y /dev/vg1/asu105
lvchange -a y /dev/vg1/asu106
lvchange -a y /dev/vg1/asu107
lvchange -a y /dev/vg1/asu108
lvchange -a y /dev/vg1/asu109
lvchange -a y /dev/vg1/asu110
lvchange -a y /dev/vg1/asu111
lvchange -a y /dev/vg1/asu112
lvchange -a y /dev/vg1/asu113
lvchange -a y /dev/vg1/asu114
lvchange -a y /dev/vg1/asu115
lvchange -a y /dev/vg1/asu116
lvchange -a y /dev/vg1/asu117
lvchange -a y /dev/vg1/asu118
lvchange -a y /dev/vg1/asu119
lvchange -a y /dev/vg1/asu120
lvchange -a y /dev/vg1/asu121
lvchange -a y /dev/vg1/asu122
lvchange -a y /dev/vg1/asu123
lvchange -a y /dev/vg1/asu124
lvchange -a y /dev/vg1/asu125
lvchange -a y /dev/vg1/asu126
lvchange -a y /dev/vg1/asu127
lvchange -a y /dev/vg1/asu128
lvchange -a y /dev/vg1/asu129
lvchange -a y /dev/vg1/asu130
lvchange -a y /dev/vg1/asu131
lvchange -a y /dev/vg1/asu132
lvchange -a y /dev/vg1/asu133
lvchange -a y /dev/vg1/asu134
lvchange -a y /dev/vg1/asu135
lvchange -a y /dev/vg1/asu136
lvchange -a y /dev/vg1/asu137
lvchange -a y /dev/vg1/asu138
lvchange -a y /dev/vg1/asu139
lvchange -a y /dev/vg1/asu140
lvchange -a y /dev/vg1/asu141
lvchange -a y /dev/vg1/asu142
lvchange -a y /dev/vg1/asu143
lvchange -a y /dev/vg1/asu144
lvchange -a y /dev/vg1/asu145
lvchange -a y /dev/vg1/asu146
lvchange -a y /dev/vg1/asu147
lvchange -a y /dev/vg1/asu148
lvchange -a y /dev/vg1/asu149
lvchange -a y /dev/vg1/asu150
lvchange -a y /dev/vg1/asu151
lvchange -a y /dev/vg1/asu152
lvchange -a y /dev/vg1/asu153
lvchange -a y /dev/vg1/asu154
lvchange -a y /dev/vg1/asu155
lvchange -a y /dev/vg1/asu156
lvchange -a y /dev/vg1/asu157
lvchange -a y /dev/vg1/asu158
lvchange -a y /dev/vg1/asu159
lvchange -a y /dev/vg1/asu160
lvchange -a y /dev/vg1/asu161
lvchange -a y /dev/vg1/asu162
lvchange -a y /dev/vg1/asu163


lvchange -a y /dev/vg2/asu200
lvchange -a y /dev/vg2/asu201
lvchange -a y /dev/vg2/asu202
lvchange -a y /dev/vg2/asu203
lvchange -a y /dev/vg2/asu204
lvchange -a y /dev/vg2/asu205
lvchange -a y /dev/vg2/asu206
lvchange -a y /dev/vg2/asu207
lvchange -a y /dev/vg2/asu208
lvchange -a y /dev/vg2/asu209
lvchange -a y /dev/vg2/asu210
lvchange -a y /dev/vg2/asu211
lvchange -a y /dev/vg2/asu212
lvchange -a y /dev/vg2/asu213
lvchange -a y /dev/vg2/asu214
lvchange -a y /dev/vg2/asu215
lvchange -a y /dev/vg2/asu216
lvchange -a y /dev/vg2/asu217
lvchange -a y /dev/vg2/asu218
lvchange -a y /dev/vg2/asu219
lvchange -a y /dev/vg2/asu220
lvchange -a y /dev/vg2/asu221
lvchange -a y /dev/vg2/asu222
lvchange -a y /dev/vg2/asu223
lvchange -a y /dev/vg2/asu224
lvchange -a y /dev/vg2/asu225
lvchange -a y /dev/vg2/asu226
lvchange -a y /dev/vg2/asu227
lvchange -a y /dev/vg2/asu228
lvchange -a y /dev/vg2/asu229
lvchange -a y /dev/vg2/asu230
lvchange -a y /dev/vg2/asu231
lvchange -a y /dev/vg2/asu232
lvchange -a y /dev/vg2/asu233
lvchange -a y /dev/vg2/asu234
lvchange -a y /dev/vg2/asu235
lvchange -a y /dev/vg2/asu236
lvchange -a y /dev/vg2/asu237
lvchange -a y /dev/vg2/asu238
lvchange -a y /dev/vg2/asu239
lvchange -a y /dev/vg2/asu240
lvchange -a y /dev/vg2/asu241
lvchange -a y /dev/vg2/asu242
lvchange -a y /dev/vg2/asu243
lvchange -a y /dev/vg2/asu244
lvchange -a y /dev/vg2/asu245
lvchange -a y /dev/vg2/asu246
lvchange -a y /dev/vg2/asu247
lvchange -a y /dev/vg2/asu248
lvchange -a y /dev/vg2/asu249
lvchange -a y /dev/vg2/asu250
lvchange -a y /dev/vg2/asu251
lvchange -a y /dev/vg2/asu252
lvchange -a y /dev/vg2/asu253
lvchange -a y /dev/vg2/asu254
lvchange -a y /dev/vg2/asu255
lvchange -a y /dev/vg2/asu256
lvchange -a y /dev/vg2/asu257
lvchange -a y /dev/vg2/asu258
lvchange -a y /dev/vg2/asu259
lvchange -a y /dev/vg2/asu260
lvchange -a y /dev/vg2/asu261
lvchange -a y /dev/vg2/asu262
lvchange -a y /dev/vg2/asu263


lvchange -a y /dev/vg3/asu300
lvchange -a y /dev/vg3/asu301
lvchange -a y /dev/vg3/asu302
lvchange -a y /dev/vg3/asu303
lvchange -a y /dev/vg3/asu304
lvchange -a y /dev/vg3/asu305
lvchange -a y /dev/vg3/asu306
lvchange -a y /dev/vg3/asu307
lvchange -a y /dev/vg3/asu308
lvchange -a y /dev/vg3/asu309
lvchange -a y /dev/vg3/asu310
lvchange -a y /dev/vg3/asu311
lvchange -a y /dev/vg3/asu312
lvchange -a y /dev/vg3/asu313
lvchange -a y /dev/vg3/asu314
lvchange -a y /dev/vg3/asu315
```

scheduler.sh

```
echo noop > /sys/block/sdb/queue/scheduler
echo noop > /sys/block/sdc/queue/scheduler
echo noop > /sys/block/sdd/queue/scheduler
echo noop > /sys/block/sde/queue/scheduler
echo noop > /sys/block/sdf/queue/scheduler
echo noop > /sys/block/sdg/queue/scheduler
echo noop > /sys/block/sdh/queue/scheduler
echo noop > /sys/block/sdi/queue/scheduler
echo noop > /sys/block/sdj/queue/scheduler
echo noop > /sys/block/sdk/queue/scheduler
echo noop > /sys/block/sdl/queue/scheduler
echo noop > /sys/block/sdm/queue/scheduler
echo noop > /sys/block/sdn/queue/scheduler
echo noop > /sys/block/sdo/queue/scheduler
echo noop > /sys/block/sdp/queue/scheduler
echo noop > /sys/block/sdq/queue/scheduler
echo noop > /sys/block/sdr/queue/scheduler
echo noop > /sys/block/sds/queue/scheduler
echo noop > /sys/block/sdt/queue/scheduler
echo noop > /sys/block/sdu/queue/scheduler
echo noop > /sys/block/sdv/queue/scheduler
echo noop > /sys/block/sdw/queue/scheduler
echo noop > /sys/block/sdx/queue/scheduler
echo noop > /sys/block/sdy/queue/scheduler
echo noop > /sys/block/sdz/queue/scheduler
echo noop > /sys/block/sdaa/queue/scheduler
echo noop > /sys/block/sdab/queue/scheduler
echo noop > /sys/block/sdac/queue/scheduler
```

```
  echo noop > /sys/block/sdad/queue/scheduler
  echo noop > /sys/block/sdae/queue/scheduler
  echo noop > /sys/block/sdaf/queue/scheduler
  echo noop > /sys/block/sdag/queue/scheduler
  echo noop > /sys/block/sdah/queue/scheduler
  echo noop > /sys/block/sdai/queue/scheduler
  echo noop > /sys/block/sdaj/queue/scheduler
  echo noop > /sys/block/sdak/queue/scheduler
  echo noop > /sys/block/sdal/queue/scheduler
  echo noop > /sys/block/sdam/queue/scheduler
  echo noop > /sys/block/sdan/queue/scheduler
  echo noop > /sys/block/sdao/queue/scheduler
  echo noop > /sys/block/sdap/queue/scheduler
  echo noop > /sys/block/sdaq/queue/scheduler
  echo noop > /sys/block/sdar/queue/scheduler
  echo noop > /sys/block/sdas/queue/scheduler
  echo noop > /sys/block/sdat/queue/scheduler
  echo noop > /sys/block/sdau/queue/scheduler
  echo noop > /sys/block/sdav/queue/scheduler
  echo noop > /sys/block/sdaw/queue/scheduler
  echo noop > /sys/block/sdax/queue/scheduler
  echo noop > /sys/block/sday/queue/scheduler
  echo noop > /sys/block/sdaz/queue/scheduler
  echo noop > /sys/block/sdba/queue/scheduler
  echo noop > /sys/block/sdbb/queue/scheduler
  echo noop > /sys/block/sdbc/queue/scheduler
  echo noop > /sys/block/sdbd/queue/scheduler
  echo noop > /sys/block/sdbe/queue/scheduler
  echo noop > /sys/block/sdbf/queue/scheduler
  echo noop > /sys/block/sdbg/queue/scheduler
  echo noop > /sys/block/sdbh/queue/scheduler
  echo noop > /sys/block/sdbi/queue/scheduler
  echo noop > /sys/block/sdbj/queue/scheduler
  echo noop > /sys/block/sdbk/queue/scheduler
  echo noop > /sys/block/sdbl/queue/scheduler
  echo noop > /sys/block/sdbm/queue/scheduler
  echo noop > /sys/block/sdbn/queue/scheduler
  echo noop > /sys/block/sdbo/queue/scheduler
  echo noop > /sys/block/sdbp/queue/scheduler
  echo noop > /sys/block/sdbq/queue/scheduler
  echo noop > /sys/block/sdbr/queue/scheduler
  echo noop > /sys/block/sdbs/queue/scheduler
  echo noop > /sys/block/sdbt/queue/scheduler
  echo noop > /sys/block/sdbu/queue/scheduler
  echo noop > /sys/block/sdbv/queue/scheduler
  echo noop > /sys/block/sdbw/queue/scheduler
  echo noop > /sys/block/sdbx/queue/scheduler
  echo noop > /sys/block/sdby/queue/scheduler
  echo noop > /sys/block/sdbz/queue/scheduler
  echo noop > /sys/block/sdca/queue/scheduler
  echo noop > /sys/block/sdcb/queue/scheduler
  echo noop > /sys/block/sdcc/queue/scheduler
  echo noop > /sys/block/sdcd/queue/scheduler
  echo noop > /sys/block/sdce/queue/scheduler
  echo noop > /sys/block/sdcf/queue/scheduler
  echo noop > /sys/block/sdcg/queue/scheduler
  echo noop > /sys/block/sdch/queue/scheduler
  echo noop > /sys/block/sdci/queue/scheduler
  echo noop > /sys/block/sdcj/queue/scheduler
  echo noop > /sys/block/sdck/queue/scheduler
  echo noop > /sys/block/sdcl/queue/scheduler
  echo noop > /sys/block/sdcm/queue/scheduler
  echo noop > /sys/block/sdcn/queue/scheduler
```

```
echo noop > /sys/block/sdco/queue/scheduler
echo noop > /sys/block/sdcp/queue/scheduler
echo noop > /sys/block/sdcq/queue/scheduler
echo noop > /sys/block/sdcr/queue/scheduler
echo noop > /sys/block/sdcs/queue/scheduler
echo noop > /sys/block/sdct/queue/scheduler
echo noop > /sys/block/sdcu/queue/scheduler
echo noop > /sys/block/sdcv/queue/scheduler
echo noop > /sys/block/sdcw/queue/scheduler
echo noop > /sys/block/sdcx/queue/scheduler
echo noop > /sys/block/sdcy/queue/scheduler
echo noop > /sys/block/sdcz/queue/scheduler
echo noop > /sys/block/sdda/queue/scheduler
echo noop > /sys/block/sddb/queue/scheduler
echo noop > /sys/block/sddc/queue/scheduler
echo noop > /sys/block/sddd/queue/scheduler
echo noop > /sys/block/sdde/queue/scheduler
echo noop > /sys/block/sddf/queue/scheduler
echo noop > /sys/block/sddg/queue/scheduler
echo noop > /sys/block/sddh/queue/scheduler
echo noop > /sys/block/sddi/queue/scheduler
echo noop > /sys/block/sddj/queue/scheduler
echo noop > /sys/block/sddk/queue/scheduler
echo noop > /sys/block/sddl/queue/scheduler
echo noop > /sys/block/sddm/queue/scheduler
echo noop > /sys/block/sddn/queue/scheduler
echo noop > /sys/block/sddo/queue/scheduler
echo noop > /sys/block/sddp/queue/scheduler
echo noop > /sys/block/sddq/queue/scheduler
echo noop > /sys/block/sddr/queue/scheduler
echo noop > /sys/block/sdds/queue/scheduler
echo noop > /sys/block/sddt/queue/scheduler
echo noop > /sys/block/sddu/queue/scheduler
echo noop > /sys/block/sddv/queue/scheduler
echo noop > /sys/block/sddw/queue/scheduler
echo noop > /sys/block/sddx/queue/scheduler
echo noop > /sys/block/sddy/queue/scheduler
echo noop > /sys/block/sddz/queue/scheduler
echo noop > /sys/block/sdea/queue/scheduler
echo noop > /sys/block/sdeb/queue/scheduler
echo noop > /sys/block/sdec/queue/scheduler
echo noop > /sys/block/sded/queue/scheduler
echo noop > /sys/block/sdee/queue/scheduler
echo noop > /sys/block/sdef/queue/scheduler
echo noop > /sys/block/sdeg/queue/scheduler
echo noop > /sys/block/sdeh/queue/scheduler
echo noop > /sys/block/sdei/queue/scheduler
echo noop > /sys/block/sdej/queue/scheduler
echo noop > /sys/block/sdek/queue/scheduler
echo noop > /sys/block/sdel/queue/scheduler
echo noop > /sys/block/sdem/queue/scheduler
echo noop > /sys/block/sden/queue/scheduler
echo noop > /sys/block/sdeo/queue/scheduler
```

chg_cpu.sh

```
#!/bin/bash

echo performance > /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu1/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu2/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu3/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu4/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu5/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu6/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu7/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu8/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu9/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu10/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu11/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu12/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu13/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu14/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu15/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu16/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu17/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu18/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu19/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu20/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu21/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu22/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu23/cpufreq/scaling_governor
```

APPENDIX D: SPC-1 WORKLOAD GENERATOR STORAGE COMMANDS AND PARAMETERS

ASU Pre-Fill

The content of command and parameter file, used in this benchmark to execute the required ASU pre-fill, is listed below.

```
hd=default,vdbench=/root/vdbench,user=root,shell=ssh
hd=hd1,system=host1
hd=hd2,system=host2
hd=hd3,system=host3
hd=hd4,system=host4
hd=hd5,system=host5
hd=hd6,system=host6


sd=default,openflags=o_direct,threads=8
sd=sd1,host=hd1,lun=/dev/vg1/asu100,size=1691143372800
sd=sd2,host=hd1,lun=/dev/vg1/asu101,size=1691143372800
sd=sd3,host=hd1,lun=/dev/vg1/asu102,size=1691143372800
sd=sd4,host=hd1,lun=/dev/vg1/asu103,size=1691143372800
sd=sd5,host=hd1,lun=/dev/vg1/asu104,size=1691143372800
sd=sd6,host=hd1,lun=/dev/vg1/asu105,size=1691143372800
sd=sd7,host=hd1,lun=/dev/vg1/asu106,size=1691143372800
sd=sd8,host=hd1,lun=/dev/vg1/asu107,size=1691143372800
sd=sd9,host=hd1,lun=/dev/vg1/asu108,size=1691143372800
sd=sd10,host=hd2,lun=/dev/vg1/asu109,size=1691143372800
sd=sd11,host=hd2,lun=/dev/vg1/asu110,size=1691143372800
sd=sd12,host=hd2,lun=/dev/vg1/asu111,size=1691143372800
sd=sd13,host=hd2,lun=/dev/vg1/asu112,size=1691143372800
sd=sd14,host=hd2,lun=/dev/vg1/asu113,size=1691143372800
sd=sd15,host=hd2,lun=/dev/vg1/asu114,size=1691143372800
sd=sd16,host=hd2,lun=/dev/vg1/asu115,size=1691143372800
sd=sd17,host=hd2,lun=/dev/vg1/asu116,size=1691143372800
sd=sd18,host=hd2,lun=/dev/vg1/asu117,size=1691143372800
sd=sd19,host=hd2,lun=/dev/vg1/asu118,size=1691143372800
sd=sd20,host=hd2,lun=/dev/vg1/asu119,size=1691143372800
sd=sd21,host=hd3,lun=/dev/vg1/asu120,size=1691143372800
sd=sd22,host=hd3,lun=/dev/vg1/asu121,size=1691143372800
sd=sd23,host=hd3,lun=/dev/vg1/asu122,size=1691143372800
sd=sd24,host=hd3,lun=/dev/vg1/asu123,size=1691143372800
sd=sd25,host=hd3,lun=/dev/vg1/asu124,size=1691143372800
sd=sd26,host=hd3,lun=/dev/vg1/asu125,size=1691143372800
sd=sd27,host=hd3,lun=/dev/vg1/asu126,size=1691143372800
sd=sd28,host=hd3,lun=/dev/vg1/asu127,size=1691143372800
sd=sd29,host=hd3,lun=/dev/vg1/asu128,size=1691143372800
sd=sd30,host=hd3,lun=/dev/vg1/asu129,size=1691143372800
sd=sd31,host=hd4,lun=/dev/vg1/asu130,size=1691143372800
sd=sd32,host=hd4,lun=/dev/vg1/asu131,size=1691143372800
sd=sd33,host=hd4,lun=/dev/vg1/asu132,size=1691143372800
sd=sd34,host=hd4,lun=/dev/vg1/asu133,size=1691143372800
sd=sd35,host=hd4,lun=/dev/vg1/asu134,size=1691143372800
sd=sd36,host=hd4,lun=/dev/vg1/asu135,size=1691143372800
sd=sd37,host=hd4,lun=/dev/vg1/asu136,size=1691143372800
sd=sd38,host=hd4,lun=/dev/vg1/asu137,size=1691143372800
sd=sd39,host=hd4,lun=/dev/vg1/asu138,size=1691143372800
sd=sd40,host=hd4,lun=/dev/vg1/asu139,size=1691143372800
sd=sd41,host=hd5,lun=/dev/vg1/asu140,size=1691143372800
sd=sd42,host=hd5,lun=/dev/vg1/asu141,size=1691143372800
```

```
sd=sd43,host=hd5,lun=/dev/vg1/asu142,size=1691143372800
sd=sd44,host=hd5,lun=/dev/vg1/asu143,size=1691143372800
sd=sd45,host=hd5,lun=/dev/vg1/asu144,size=1691143372800
sd=sd46,host=hd5,lun=/dev/vg1/asu145,size=1691143372800
sd=sd47,host=hd5,lun=/dev/vg1/asu146,size=1691143372800
sd=sd48,host=hd5,lun=/dev/vg1/asu147,size=1691143372800
sd=sd49,host=hd5,lun=/dev/vg1/asu148,size=1691143372800
sd=sd50,host=hd5,lun=/dev/vg1/asu149,size=1691143372800
sd=sd51,host=hd6,lun=/dev/vg1/asu150,size=1691143372800
sd=sd52,host=hd6,lun=/dev/vg1/asu151,size=1691143372800
sd=sd53,host=hd6,lun=/dev/vg1/asu152,size=1691143372800
sd=sd54,host=hd6,lun=/dev/vg1/asu153,size=1691143372800
sd=sd55,host=hd6,lun=/dev/vg1/asu154,size=1691143372800
sd=sd56,host=hd6,lun=/dev/vg1/asu155,size=1691143372800
sd=sd57,host=hd6,lun=/dev/vg1/asu156,size=1691143372800
sd=sd58,host=hd6,lun=/dev/vg1/asu157,size=1691143372800
sd=sd59,host=hd6,lun=/dev/vg1/asu158,size=1691143372800
sd=sd60,host=hd6,lun=/dev/vg1/asu159,size=1691143372800
sd=sd61,host=hd1,lun=/dev/vg1/asu160,size=1691143372800
sd=sd62,host=hd2,lun=/dev/vg1/asu161,size=1691143372800
sd=sd63,host=hd3,lun=/dev/vg1/asu162,size=1691143372800
sd=sd64,host=hd4,lun=/dev/vg1/asu163,size=1691143372800

sd=sd65,host=hd1,lun=/dev/vg2/asu200,size=1691143372800
sd=sd66,host=hd1,lun=/dev/vg2/asu201,size=1691143372800
sd=sd67,host=hd1,lun=/dev/vg2/asu202,size=1691143372800
sd=sd68,host=hd1,lun=/dev/vg2/asu203,size=1691143372800
sd=sd69,host=hd1,lun=/dev/vg2/asu204,size=1691143372800
sd=sd70,host=hd1,lun=/dev/vg2/asu205,size=1691143372800
sd=sd71,host=hd1,lun=/dev/vg2/asu206,size=1691143372800
sd=sd72,host=hd1,lun=/dev/vg2/asu207,size=1691143372800
sd=sd73,host=hd1,lun=/dev/vg2/asu208,size=1691143372800
sd=sd74,host=hd2,lun=/dev/vg2/asu209,size=1691143372800
sd=sd75,host=hd2,lun=/dev/vg2/asu210,size=1691143372800
sd=sd76,host=hd2,lun=/dev/vg2/asu211,size=1691143372800
sd=sd77,host=hd2,lun=/dev/vg2/asu212,size=1691143372800
sd=sd78,host=hd2,lun=/dev/vg2/asu213,size=1691143372800
sd=sd79,host=hd2,lun=/dev/vg2/asu214,size=1691143372800
sd=sd80,host=hd2,lun=/dev/vg2/asu215,size=1691143372800
sd=sd81,host=hd2,lun=/dev/vg2/asu216,size=1691143372800
sd=sd82,host=hd2,lun=/dev/vg2/asu217,size=1691143372800
sd=sd83,host=hd2,lun=/dev/vg2/asu218,size=1691143372800
sd=sd84,host=hd2,lun=/dev/vg2/asu219,size=1691143372800
sd=sd85,host=hd3,lun=/dev/vg2/asu220,size=1691143372800
sd=sd86,host=hd3,lun=/dev/vg2/asu221,size=1691143372800
sd=sd87,host=hd3,lun=/dev/vg2/asu222,size=1691143372800
sd=sd88,host=hd3,lun=/dev/vg2/asu223,size=1691143372800
sd=sd89,host=hd3,lun=/dev/vg2/asu224,size=1691143372800
sd=sd90,host=hd3,lun=/dev/vg2/asu225,size=1691143372800
sd=sd91,host=hd3,lun=/dev/vg2/asu226,size=1691143372800
sd=sd92,host=hd3,lun=/dev/vg2/asu227,size=1691143372800
sd=sd93,host=hd3,lun=/dev/vg2/asu228,size=1691143372800
sd=sd94,host=hd3,lun=/dev/vg2/asu229,size=1691143372800
sd=sd95,host=hd4,lun=/dev/vg2/asu230,size=1691143372800
sd=sd96,host=hd4,lun=/dev/vg2/asu231,size=1691143372800
sd=sd97,host=hd4,lun=/dev/vg2/asu232,size=1691143372800
sd=sd98,host=hd4,lun=/dev/vg2/asu233,size=1691143372800
sd=sd99,host=hd4,lun=/dev/vg2/asu234,size=1691143372800
sd=sd100,host=hd4,lun=/dev/vg2/asu235,size=1691143372800
sd=sd101,host=hd4,lun=/dev/vg2/asu236,size=1691143372800
sd=sd102,host=hd4,lun=/dev/vg2/asu237,size=1691143372800
sd=sd103,host=hd4,lun=/dev/vg2/asu238,size=1691143372800
sd=sd104,host=hd4,lun=/dev/vg2/asu239,size=1691143372800
```

```
sd=sd105,host=hd5,lun=/dev/vg2/asu240,size=1691143372800
sd=sd106,host=hd5,lun=/dev/vg2/asu241,size=1691143372800
sd=sd107,host=hd5,lun=/dev/vg2/asu242,size=1691143372800
sd=sd108,host=hd5,lun=/dev/vg2/asu243,size=1691143372800
sd=sd109,host=hd5,lun=/dev/vg2/asu244,size=1691143372800
sd=sd110,host=hd5,lun=/dev/vg2/asu245,size=1691143372800
sd=sd111,host=hd5,lun=/dev/vg2/asu246,size=1691143372800
sd=sd112,host=hd5,lun=/dev/vg2/asu247,size=1691143372800
sd=sd113,host=hd5,lun=/dev/vg2/asu248,size=1691143372800
sd=sd114,host=hd5,lun=/dev/vg2/asu249,size=1691143372800
sd=sd115,host=hd6,lun=/dev/vg2/asu250,size=1691143372800
sd=sd116,host=hd6,lun=/dev/vg2/asu251,size=1691143372800
sd=sd117,host=hd6,lun=/dev/vg2/asu252,size=1691143372800
sd=sd118,host=hd6,lun=/dev/vg2/asu253,size=1691143372800
sd=sd119,host=hd6,lun=/dev/vg2/asu254,size=1691143372800
sd=sd120,host=hd6,lun=/dev/vg2/asu255,size=1691143372800
sd=sd121,host=hd6,lun=/dev/vg2/asu256,size=1691143372800
sd=sd122,host=hd6,lun=/dev/vg2/asu257,size=1691143372800
sd=sd123,host=hd6,lun=/dev/vg2/asu258,size=1691143372800
sd=sd124,host=hd6,lun=/dev/vg2/asu259,size=1691143372800
sd=sd125,host=hd1,lun=/dev/vg2/asu260,size=1691143372800
sd=sd126,host=hd2,lun=/dev/vg2/asu261,size=1691143372800
sd=sd127,host=hd3,lun=/dev/vg2/asu262,size=1691143372800
sd=sd128,host=hd4,lun=/dev/vg2/asu263,size=1691143372800

sd=sd129,host=hd1,lun=/dev/vg3/asu300,size=1503238553600
sd=sd130,host=hd1,lun=/dev/vg3/asu301,size=1503238553600
sd=sd131,host=hd2,lun=/dev/vg3/asu302,size=1503238553600
sd=sd132,host=hd2,lun=/dev/vg3/asu303,size=1503238553600
sd=sd133,host=hd3,lun=/dev/vg3/asu304,size=1503238553600
sd=sd134,host=hd3,lun=/dev/vg3/asu305,size=1503238553600
sd=sd135,host=hd4,lun=/dev/vg3/asu306,size=1503238553600
sd=sd136,host=hd4,lun=/dev/vg3/asu307,size=1503238553600
sd=sd137,host=hd5,lun=/dev/vg3/asu308,size=1503238553600
sd=sd138,host=hd5,lun=/dev/vg3/asu309,size=1503238553600
sd=sd139,host=hd6,lun=/dev/vg3/asu310,size=1503238553600
sd=sd140,host=hd6,lun=/dev/vg3/asu311,size=1503238553600
sd=sd141,host=hd5,lun=/dev/vg3/asu312,size=1503238553600
sd=sd142,host=hd5,lun=/dev/vg3/asu313,size=1503238553600
sd=sd143,host=hd6,lun=/dev/vg3/asu314,size=1503238553600
sd=sd144,host=hd6,lun=/dev/vg3/asu315,size=1503238553600

wd=wd1,sd=sd*,rdpct=0,seekpct=-1,xfersize=256K
rd=PREPASU1,wd=wd1,iorate=max,elapsed=3600000,interval=10
```

Common Command Lines

The following command lines appear at the beginning of each command and parameter files. The command lines are only listed below to eliminate redundancy.

```
sd=asu1_0,lun=/dev/vg1/asu100,size=1691143372800
sd=asu1_1,lun=/dev/vg1/asu101,size=1691143372800
sd=asu1_2,lun=/dev/vg1/asu102,size=1691143372800
sd=asu1_3,lun=/dev/vg1/asu103,size=1691143372800
sd=asu1_4,lun=/dev/vg1/asu104,size=1691143372800
sd=asu1_5,lun=/dev/vg1/asu105,size=1691143372800
sd=asu1_6,lun=/dev/vg1/asu106,size=1691143372800
sd=asu1_7,lun=/dev/vg1/asu107,size=1691143372800
sd=asu1_8,lun=/dev/vg1/asu108,size=1691143372800
sd=asu1_9,lun=/dev/vg1/asu109,size=1691143372800
sd=asu1_10,lun=/dev/vg1/asu110,size=1691143372800
sd=asu1_11,lun=/dev/vg1/asu111,size=1691143372800
sd=asu1_12,lun=/dev/vg1/asu112,size=1691143372800
sd=asu1_13,lun=/dev/vg1/asu113,size=1691143372800
sd=asu1_14,lun=/dev/vg1/asu114,size=1691143372800
sd=asu1_15,lun=/dev/vg1/asu115,size=1691143372800
sd=asu1_16,lun=/dev/vg1/asu116,size=1691143372800
sd=asu1_17,lun=/dev/vg1/asu117,size=1691143372800
sd=asu1_18,lun=/dev/vg1/asu118,size=1691143372800
sd=asu1_19,lun=/dev/vg1/asu119,size=1691143372800
sd=asu1_20,lun=/dev/vg1/asu120,size=1691143372800
sd=asu1_21,lun=/dev/vg1/asu121,size=1691143372800
sd=asu1_22,lun=/dev/vg1/asu122,size=1691143372800
sd=asu1_23,lun=/dev/vg1/asu123,size=1691143372800
sd=asu1_24,lun=/dev/vg1/asu124,size=1691143372800
sd=asu1_25,lun=/dev/vg1/asu125,size=1691143372800
sd=asu1_26,lun=/dev/vg1/asu126,size=1691143372800
sd=asu1_27,lun=/dev/vg1/asu127,size=1691143372800
sd=asu1_28,lun=/dev/vg1/asu128,size=1691143372800
sd=asu1_29,lun=/dev/vg1/asu129,size=1691143372800
sd=asu1_30,lun=/dev/vg1/asu130,size=1691143372800
sd=asu1_31,lun=/dev/vg1/asu131,size=1691143372800
sd=asu1_32,lun=/dev/vg1/asu132,size=1691143372800
sd=asu1_33,lun=/dev/vg1/asu133,size=1691143372800
sd=asu1_34,lun=/dev/vg1/asu134,size=1691143372800
sd=asu1_35,lun=/dev/vg1/asu135,size=1691143372800
sd=asu1_36,lun=/dev/vg1/asu136,size=1691143372800
sd=asu1_37,lun=/dev/vg1/asu137,size=1691143372800
sd=asu1_38,lun=/dev/vg1/asu138,size=1691143372800
sd=asu1_39,lun=/dev/vg1/asu139,size=1691143372800
sd=asu1_40,lun=/dev/vg1/asu140,size=1691143372800
sd=asu1_41,lun=/dev/vg1/asu141,size=1691143372800
sd=asu1_42,lun=/dev/vg1/asu142,size=1691143372800
sd=asu1_43,lun=/dev/vg1/asu143,size=1691143372800
sd=asu1_44,lun=/dev/vg1/asu144,size=1691143372800
sd=asu1_45,lun=/dev/vg1/asu145,size=1691143372800
sd=asu1_46,lun=/dev/vg1/asu146,size=1691143372800
sd=asu1_47,lun=/dev/vg1/asu147,size=1691143372800
sd=asu1_48,lun=/dev/vg1/asu148,size=1691143372800
sd=asu1_49,lun=/dev/vg1/asu149,size=1691143372800
sd=asu1_50,lun=/dev/vg1/asu150,size=1691143372800
sd=asu1_51,lun=/dev/vg1/asu151,size=1691143372800
sd=asu1_52,lun=/dev/vg1/asu152,size=1691143372800
sd=asu1_53,lun=/dev/vg1/asu153,size=1691143372800
sd=asu1_54,lun=/dev/vg1/asu154,size=1691143372800
sd=asu1_55,lun=/dev/vg1/asu155,size=1691143372800
```

```
sd=asu1_56,lun=/dev/vg1/asu156,size=1691143372800
sd=asu1_57,lun=/dev/vg1/asu157,size=1691143372800
sd=asu1_58,lun=/dev/vg1/asu158,size=1691143372800
sd=asu1_59,lun=/dev/vg1/asu159,size=1691143372800
sd=asu1_60,lun=/dev/vg1/asu160,size=1691143372800
sd=asu1_61,lun=/dev/vg1/asu161,size=1691143372800
sd=asu1_62,lun=/dev/vg1/asu162,size=1691143372800
sd=asu1_63,lun=/dev/vg1/asu163,size=1691143372800


sd=asu2_0,lun=/dev/vg2/asu200,size=1691143372800
sd=asu2_1,lun=/dev/vg2/asu201,size=1691143372800
sd=asu2_2,lun=/dev/vg2/asu202,size=1691143372800
sd=asu2_3,lun=/dev/vg2/asu203,size=1691143372800
sd=asu2_4,lun=/dev/vg2/asu204,size=1691143372800
sd=asu2_5,lun=/dev/vg2/asu205,size=1691143372800
sd=asu2_6,lun=/dev/vg2/asu206,size=1691143372800
sd=asu2_7,lun=/dev/vg2/asu207,size=1691143372800
sd=asu2_8,lun=/dev/vg2/asu208,size=1691143372800
sd=asu2_9,lun=/dev/vg2/asu209,size=1691143372800
sd=asu2_10,lun=/dev/vg2/asu210,size=1691143372800
sd=asu2_11,lun=/dev/vg2/asu211,size=1691143372800
sd=asu2_12,lun=/dev/vg2/asu212,size=1691143372800
sd=asu2_13,lun=/dev/vg2/asu213,size=1691143372800
sd=asu2_14,lun=/dev/vg2/asu214,size=1691143372800
sd=asu2_15,lun=/dev/vg2/asu215,size=1691143372800
sd=asu2_16,lun=/dev/vg2/asu216,size=1691143372800
sd=asu2_17,lun=/dev/vg2/asu217,size=1691143372800
sd=asu2_18,lun=/dev/vg2/asu218,size=1691143372800
sd=asu2_19,lun=/dev/vg2/asu219,size=1691143372800
sd=asu2_20,lun=/dev/vg2/asu220,size=1691143372800
sd=asu2_21,lun=/dev/vg2/asu221,size=1691143372800
sd=asu2_22,lun=/dev/vg2/asu222,size=1691143372800
sd=asu2_23,lun=/dev/vg2/asu223,size=1691143372800
sd=asu2_24,lun=/dev/vg2/asu224,size=1691143372800
sd=asu2_25,lun=/dev/vg2/asu225,size=1691143372800
sd=asu2_26,lun=/dev/vg2/asu226,size=1691143372800
sd=asu2_27,lun=/dev/vg2/asu227,size=1691143372800
sd=asu2_28,lun=/dev/vg2/asu228,size=1691143372800
sd=asu2_29,lun=/dev/vg2/asu229,size=1691143372800
sd=asu2_30,lun=/dev/vg2/asu230,size=1691143372800
sd=asu2_31,lun=/dev/vg2/asu231,size=1691143372800
sd=asu2_32,lun=/dev/vg2/asu232,size=1691143372800
sd=asu2_33,lun=/dev/vg2/asu233,size=1691143372800
sd=asu2_34,lun=/dev/vg2/asu234,size=1691143372800
sd=asu2_35,lun=/dev/vg2/asu235,size=1691143372800
sd=asu2_36,lun=/dev/vg2/asu236,size=1691143372800
sd=asu2_37,lun=/dev/vg2/asu237,size=1691143372800
sd=asu2_38,lun=/dev/vg2/asu238,size=1691143372800
sd=asu2_39,lun=/dev/vg2/asu239,size=1691143372800
sd=asu2_40,lun=/dev/vg2/asu240,size=1691143372800
sd=asu2_41,lun=/dev/vg2/asu241,size=1691143372800
sd=asu2_42,lun=/dev/vg2/asu242,size=1691143372800
sd=asu2_43,lun=/dev/vg2/asu243,size=1691143372800
sd=asu2_44,lun=/dev/vg2/asu244,size=1691143372800
sd=asu2_45,lun=/dev/vg2/asu245,size=1691143372800
sd=asu2_46,lun=/dev/vg2/asu246,size=1691143372800
sd=asu2_47,lun=/dev/vg2/asu247,size=1691143372800
sd=asu2_48,lun=/dev/vg2/asu248,size=1691143372800
sd=asu2_49,lun=/dev/vg2/asu249,size=1691143372800
sd=asu2_50,lun=/dev/vg2/asu250,size=1691143372800
sd=asu2_51,lun=/dev/vg2/asu251,size=1691143372800
sd=asu2_52,lun=/dev/vg2/asu252,size=1691143372800
```

```
sd=asu2_53,lun=/dev/vg2/asu253,size=1691143372800
sd=asu2_54,lun=/dev/vg2/asu254,size=1691143372800
sd=asu2_55,lun=/dev/vg2/asu255,size=1691143372800
sd=asu2_56,lun=/dev/vg2/asu256,size=1691143372800
sd=asu2_57,lun=/dev/vg2/asu257,size=1691143372800
sd=asu2_58,lun=/dev/vg2/asu258,size=1691143372800
sd=asu2_59,lun=/dev/vg2/asu259,size=1691143372800
sd=asu2_60,lun=/dev/vg2/asu260,size=1691143372800
sd=asu2_61,lun=/dev/vg2/asu261,size=1691143372800
sd=asu2_62,lun=/dev/vg2/asu262,size=1691143372800
sd=asu2_63,lun=/dev/vg2/asu263,size=1691143372800


sd=asu3_0,lun=/dev/vg3/asu300,size=1503238553600
sd=asu3_1,lun=/dev/vg3/asu301,size=1503238553600
sd=asu3_2,lun=/dev/vg3/asu302,size=1503238553600
sd=asu3_3,lun=/dev/vg3/asu303,size=1503238553600
sd=asu3_4,lun=/dev/vg3/asu304,size=1503238553600
sd=asu3_5,lun=/dev/vg3/asu305,size=1503238553600
sd=asu3_6,lun=/dev/vg3/asu306,size=1503238553600
sd=asu3_7,lun=/dev/vg3/asu307,size=1503238553600
sd=asu3_8,lun=/dev/vg3/asu308,size=1503238553600
sd=asu3_9,lun=/dev/vg3/asu309,size=1503238553600
sd=asu3_10,lun=/dev/vg3/asu310,size=1503238553600
sd=asu3_11,lun=/dev/vg3/asu311,size=1503238553600
sd=asu3_12,lun=/dev/vg3/asu312,size=1503238553600
sd=asu3_13,lun=/dev/vg3/asu313,size=1503238553600
sd=asu3_14,lun=/dev/vg3/asu314,size=1503238553600
sd=asu3_15,lun=/dev/vg3/asu315,size=1503238553600
```

Primary Metrics and Repeatability Tests

The content of SPC-1 Workload Generator command and parameter file used in this benchmark to execute the Primary Metrics *(Sustainability Test Phase, IOPS Test Phase, and Response Time Ramp Test Phase)* and Repeatability *(Repeatability Test Phase 1 and Repeatability Test Phase 2)* Tests is listed below.

```
host=master
slaves=(slave1,slave2,slave3,slave4,slave5,slave6,slave7,slave8,slave9,slave10,slave
11,slave12,slave13,slave14,slave15,slave16,slave17,slave18,slave19,slave20,slave21,s
lave22,slave23,slave24,slave25,slave26,slave27,slave28,slave29,slave30,slave31,slave
32,slave33,slave34,slave35,slave36,slave37,slave38,slave39,slave40,slave41,slave42,s
lave43,slave44,slave45,slave46,slave47,slave48,slave49,slave50,slave51,slave52,slave
53,slave54,slave55,slave56,slave57,slave58,slave59,slave60,slave61,slave62,slave63,s
lave64,slave65,slave66,slave67,slave68,slave69,slave70,slave71,slave72,slave73,slave
74,slave75,slave76,slave77,slave78,slave79,slave80,slave81,slave82,slave83,slave84,s
lave85,slave86,slave87,slave88,slave89,slave90,slave91,slave92,slave93,slave94,slave
95,slave96,slave97,slave98,slave99,slave100,slave101,slave102,slave103,slave104,slav
e105,slave106,slave107,slave108,slave109,slave110,slave111,slave112,slave113,slave11
4,slave115,slave116,slave117,slave118,slave119,slave120,slave121,slave122,slave123,s
lave124,slave125,slave126,slave127,slave128,slave129,slave130,slave131,slave132)
```

common commands

SPC-1 Persistence Test Run 1

The content of SPC-1 Workload Generator command and parameter file, used in this benchmark to execute a reduced level SPC-1 Persistence Test Run 1, is listed below.

```
common commands
```

Slave JVMs

Each Slave JVM was invoked with a command and parameter file similar to the example listed below. The only difference in each file was **host** parameter value, which was unique to each Slave JVM, e.g. **slave1**…**slave132**.

```
master=host1
host=slave1

common commands
```

SPC-2 Persistence Test

If approved by the SPC Auditor, the SPC-2 Persistence Test may be used to meet the SPC-1 persistence requirements. Both the SPC-1 and SPC-2 Persistence Tests provide the same level of functionality and verification of data integrity.

Common Command Lines – SPC-2 Persistence Test

The following command lines appear at the beginning of each command and parameter file for the two SPC-2 Persistence Test Runs. The command lines are only listed below to eliminate redundancy

```
host=localhost,spc2="/root/spc2",shell=spc2,jvms=10,maxstreams=100
host=(100.124.7.12,coltrane1),spc2="/root/spc2",shell=spc2,jvms=10,maxstreams=100
host=(100.124.7.13,coltrane2),spc2="/root/spc2",shell=spc2,jvms=10,maxstreams=100

sd=default,host=localhost
sd=sd1,lun=/dev/vg1/asu100,size=1691143372800
sd=sd2,lun=/dev/vg1/asu101,size=1691143372800
sd=sd3,lun=/dev/vg1/asu102,size=1691143372800
sd=sd4,lun=/dev/vg1/asu103,size=1691143372800
sd=sd5,lun=/dev/vg1/asu104,size=1691143372800
sd=sd6,lun=/dev/vg1/asu105,size=1691143372800
sd=sd7,lun=/dev/vg1/asu106,size=1691143372800
sd=sd8,lun=/dev/vg1/asu107,size=1691143372800
sd=sd9,lun=/dev/vg1/asu108,size=1691143372800
sd=sd10,lun=/dev/vg1/asu109,size=1691143372800
sd=sd11,lun=/dev/vg1/asu110,size=1691143372800
sd=sd12,lun=/dev/vg1/asu111,size=1691143372800
sd=sd13,lun=/dev/vg1/asu112,size=1691143372800
sd=sd14,lun=/dev/vg1/asu113,size=1691143372800
sd=sd15,lun=/dev/vg1/asu114,size=1691143372800
```

```
sd=sd16,lun=/dev/vg1/asu115,size=1691143372800
sd=sd17,lun=/dev/vg1/asu116,size=1691143372800
sd=sd18,lun=/dev/vg1/asu117,size=1691143372800
sd=sd19,lun=/dev/vg1/asu118,size=1691143372800
sd=sd20,lun=/dev/vg1/asu119,size=1691143372800
sd=sd21,lun=/dev/vg1/asu120,size=1691143372800
sd=sd22,lun=/dev/vg1/asu121,size=1691143372800
sd=sd23,lun=/dev/vg1/asu122,size=1691143372800
sd=sd24,lun=/dev/vg1/asu123,size=1691143372800
sd=sd25,lun=/dev/vg1/asu124,size=1691143372800
sd=sd26,lun=/dev/vg1/asu125,size=1691143372800
sd=sd27,lun=/dev/vg1/asu126,size=1691143372800
sd=sd28,lun=/dev/vg1/asu127,size=1691143372800
sd=sd29,lun=/dev/vg1/asu128,size=1691143372800
sd=sd30,lun=/dev/vg1/asu129,size=1691143372800
sd=sd31,lun=/dev/vg1/asu130,size=1691143372800
sd=sd32,lun=/dev/vg1/asu131,size=1691143372800
sd=sd33,lun=/dev/vg1/asu132,size=1691143372800
sd=sd34,lun=/dev/vg1/asu133,size=1691143372800
sd=sd35,lun=/dev/vg1/asu134,size=1691143372800
sd=sd36,lun=/dev/vg1/asu135,size=1691143372800
sd=sd37,lun=/dev/vg1/asu136,size=1691143372800
sd=sd38,lun=/dev/vg1/asu137,size=1691143372800
sd=sd39,lun=/dev/vg1/asu138,size=1691143372800
sd=sd40,lun=/dev/vg1/asu139,size=1691143372800
sd=sd41,lun=/dev/vg1/asu140,size=1691143372800
sd=sd42,lun=/dev/vg1/asu141,size=1691143372800
sd=sd43,lun=/dev/vg1/asu142,size=1691143372800
sd=sd44,lun=/dev/vg1/asu143,size=1691143372800
sd=sd45,lun=/dev/vg1/asu144,size=1691143372800
sd=sd46,lun=/dev/vg1/asu145,size=1691143372800
sd=sd47,lun=/dev/vg1/asu146,size=1691143372800
sd=sd48,lun=/dev/vg1/asu147,size=1691143372800
sd=sd49,lun=/dev/vg1/asu148,size=1691143372800
sd=sd50,lun=/dev/vg1/asu149,size=1691143372800
sd=sd51,lun=/dev/vg1/asu150,size=1691143372800
sd=sd52,lun=/dev/vg1/asu151,size=1691143372800
sd=sd53,lun=/dev/vg1/asu152,size=1691143372800
sd=sd54,lun=/dev/vg1/asu153,size=1691143372800
sd=sd55,lun=/dev/vg1/asu154,size=1691143372800
sd=sd56,lun=/dev/vg1/asu155,size=1691143372800
sd=sd57,lun=/dev/vg1/asu156,size=1691143372800
sd=sd58,lun=/dev/vg1/asu157,size=1691143372800
sd=sd59,lun=/dev/vg1/asu158,size=1691143372800
sd=sd60,lun=/dev/vg1/asu159,size=1691143372800
sd=sd61,lun=/dev/vg1/asu160,size=1691143372800
sd=sd62,lun=/dev/vg1/asu161,size=1691143372800
sd=sd63,lun=/dev/vg1/asu162,size=1691143372800
sd=sd64,lun=/dev/vg1/asu163,size=1691143372800

sd=sd65,lun=/dev/vg2/asu200,size=1691143372800
sd=sd66,lun=/dev/vg2/asu201,size=1691143372800
sd=sd67,lun=/dev/vg2/asu202,size=1691143372800
sd=sd68,lun=/dev/vg2/asu203,size=1691143372800
sd=sd69,lun=/dev/vg2/asu204,size=1691143372800
sd=sd70,lun=/dev/vg2/asu205,size=1691143372800
sd=sd71,lun=/dev/vg2/asu206,size=1691143372800
sd=sd72,lun=/dev/vg2/asu207,size=1691143372800
sd=sd73,lun=/dev/vg2/asu208,size=1691143372800
sd=sd74,lun=/dev/vg2/asu209,size=1691143372800
sd=sd75,lun=/dev/vg2/asu210,size=1691143372800
sd=sd76,lun=/dev/vg2/asu211,size=1691143372800
sd=sd77,lun=/dev/vg2/asu212,size=1691143372800
```

```
  sd=sd78,lun=/dev/vg2/asu213,size=1691143372800
  sd=sd79,lun=/dev/vg2/asu214,size=1691143372800
  sd=sd80,lun=/dev/vg2/asu215,size=1691143372800
  sd=sd81,lun=/dev/vg2/asu216,size=1691143372800
  sd=sd82,lun=/dev/vg2/asu217,size=1691143372800
  sd=sd83,lun=/dev/vg2/asu218,size=1691143372800
  sd=sd84,lun=/dev/vg2/asu219,size=1691143372800
  sd=sd85,lun=/dev/vg2/asu220,size=1691143372800
  sd=sd86,lun=/dev/vg2/asu221,size=1691143372800
  sd=sd87,lun=/dev/vg2/asu222,size=1691143372800
  sd=sd88,lun=/dev/vg2/asu223,size=1691143372800
  sd=sd89,lun=/dev/vg2/asu224,size=1691143372800
  sd=sd90,lun=/dev/vg2/asu225,size=1691143372800
  sd=sd91,lun=/dev/vg2/asu226,size=1691143372800
  sd=sd92,lun=/dev/vg2/asu227,size=1691143372800
  sd=sd93,lun=/dev/vg2/asu228,size=1691143372800
  sd=sd94,lun=/dev/vg2/asu229,size=1691143372800
  sd=sd95,lun=/dev/vg2/asu230,size=1691143372800
  sd=sd96,lun=/dev/vg2/asu231,size=1691143372800
  sd=sd97,lun=/dev/vg2/asu232,size=1691143372800
  sd=sd98,lun=/dev/vg2/asu233,size=1691143372800
  sd=sd99,lun=/dev/vg2/asu234,size=1691143372800
  sd=sd100,lun=/dev/vg2/asu235,size=1691143372800
  sd=sd101,lun=/dev/vg2/asu236,size=1691143372800
  sd=sd102,lun=/dev/vg2/asu237,size=1691143372800
  sd=sd103,lun=/dev/vg2/asu238,size=1691143372800
  sd=sd104,lun=/dev/vg2/asu239,size=1691143372800
  sd=sd105,lun=/dev/vg2/asu240,size=1691143372800
  sd=sd106,lun=/dev/vg2/asu241,size=1691143372800
  sd=sd107,lun=/dev/vg2/asu242,size=1691143372800
  sd=sd108,lun=/dev/vg2/asu243,size=1691143372800
  sd=sd109,lun=/dev/vg2/asu244,size=1691143372800
  sd=sd110,lun=/dev/vg2/asu245,size=1691143372800
  sd=sd111,lun=/dev/vg2/asu246,size=1691143372800
  sd=sd112,lun=/dev/vg2/asu247,size=1691143372800
  sd=sd113,lun=/dev/vg2/asu248,size=1691143372800
  sd=sd114,lun=/dev/vg2/asu249,size=1691143372800
  sd=sd115,lun=/dev/vg2/asu250,size=1691143372800
  sd=sd116,lun=/dev/vg2/asu251,size=1691143372800
  sd=sd117,lun=/dev/vg2/asu252,size=1691143372800
  sd=sd118,lun=/dev/vg2/asu253,size=1691143372800
  sd=sd119,lun=/dev/vg2/asu254,size=1691143372800
  sd=sd120,lun=/dev/vg2/asu255,size=1691143372800
  sd=sd121,lun=/dev/vg2/asu256,size=1691143372800
  sd=sd122,lun=/dev/vg2/asu257,size=1691143372800
  sd=sd123,lun=/dev/vg2/asu258,size=1691143372800
  sd=sd124,lun=/dev/vg2/asu259,size=1691143372800
  sd=sd125,lun=/dev/vg2/asu260,size=1691143372800
  sd=sd126,lun=/dev/vg2/asu261,size=1691143372800
  sd=sd127,lun=/dev/vg2/asu262,size=1691143372800
  sd=sd128,lun=/dev/vg2/asu263,size=1691143372800

  sd=sd129,lun=/dev/vg3/asu300,size=1503238553600
  sd=sd130,lun=/dev/vg3/asu301,size=1503238553600
  sd=sd131,lun=/dev/vg3/asu302,size=1503238553600
  sd=sd132,lun=/dev/vg3/asu303,size=1503238553600
  sd=sd133,lun=/dev/vg3/asu304,size=1503238553600
  sd=sd134,lun=/dev/vg3/asu305,size=1503238553600
  sd=sd135,lun=/dev/vg3/asu306,size=1503238553600
  sd=sd136,lun=/dev/vg3/asu307,size=1503238553600
  sd=sd137,lun=/dev/vg3/asu308,size=1503238553600
  sd=sd138,lun=/dev/vg3/asu309,size=1503238553600
  sd=sd139,lun=/dev/vg3/asu310,size=1503238553600
```

```
sd=sd140,lun=/dev/vg3/asu311,size=1503238553600
sd=sd141,lun=/dev/vg3/asu312,size=1503238553600
sd=sd142,lun=/dev/vg3/asu313,size=1503238553600
sd=sd143,lun=/dev/vg3/asu314,size=1503238553600
sd=sd144,lun=/dev/vg3/asu315,size=1503238553600


sd=default,host=coltrane1
sd=sd1,lun=/dev/vg1/asu100,size=1691143372800
sd=sd2,lun=/dev/vg1/asu101,size=1691143372800
sd=sd3,lun=/dev/vg1/asu102,size=1691143372800
sd=sd4,lun=/dev/vg1/asu103,size=1691143372800
sd=sd5,lun=/dev/vg1/asu104,size=1691143372800
sd=sd6,lun=/dev/vg1/asu105,size=1691143372800
sd=sd7,lun=/dev/vg1/asu106,size=1691143372800
sd=sd8,lun=/dev/vg1/asu107,size=1691143372800
sd=sd9,lun=/dev/vg1/asu108,size=1691143372800
sd=sd10,lun=/dev/vg1/asu109,size=1691143372800
sd=sd11,lun=/dev/vg1/asu110,size=1691143372800
sd=sd12,lun=/dev/vg1/asu111,size=1691143372800
sd=sd13,lun=/dev/vg1/asu112,size=1691143372800
sd=sd14,lun=/dev/vg1/asu113,size=1691143372800
sd=sd15,lun=/dev/vg1/asu114,size=1691143372800
sd=sd16,lun=/dev/vg1/asu115,size=1691143372800
sd=sd17,lun=/dev/vg1/asu116,size=1691143372800
sd=sd18,lun=/dev/vg1/asu117,size=1691143372800
sd=sd19,lun=/dev/vg1/asu118,size=1691143372800
sd=sd20,lun=/dev/vg1/asu119,size=1691143372800
sd=sd21,lun=/dev/vg1/asu120,size=1691143372800
sd=sd22,lun=/dev/vg1/asu121,size=1691143372800
sd=sd23,lun=/dev/vg1/asu122,size=1691143372800
sd=sd24,lun=/dev/vg1/asu123,size=1691143372800
sd=sd25,lun=/dev/vg1/asu124,size=1691143372800
sd=sd26,lun=/dev/vg1/asu125,size=1691143372800
sd=sd27,lun=/dev/vg1/asu126,size=1691143372800
sd=sd28,lun=/dev/vg1/asu127,size=1691143372800
sd=sd29,lun=/dev/vg1/asu128,size=1691143372800
sd=sd30,lun=/dev/vg1/asu129,size=1691143372800
sd=sd31,lun=/dev/vg1/asu130,size=1691143372800
sd=sd32,lun=/dev/vg1/asu131,size=1691143372800
sd=sd33,lun=/dev/vg1/asu132,size=1691143372800
sd=sd34,lun=/dev/vg1/asu133,size=1691143372800
sd=sd35,lun=/dev/vg1/asu134,size=1691143372800
sd=sd36,lun=/dev/vg1/asu135,size=1691143372800
sd=sd37,lun=/dev/vg1/asu136,size=1691143372800
sd=sd38,lun=/dev/vg1/asu137,size=1691143372800
sd=sd39,lun=/dev/vg1/asu138,size=1691143372800
sd=sd40,lun=/dev/vg1/asu139,size=1691143372800
sd=sd41,lun=/dev/vg1/asu140,size=1691143372800
sd=sd42,lun=/dev/vg1/asu141,size=1691143372800
sd=sd43,lun=/dev/vg1/asu142,size=1691143372800
sd=sd44,lun=/dev/vg1/asu143,size=1691143372800
sd=sd45,lun=/dev/vg1/asu144,size=1691143372800
sd=sd46,lun=/dev/vg1/asu145,size=1691143372800
sd=sd47,lun=/dev/vg1/asu146,size=1691143372800
sd=sd48,lun=/dev/vg1/asu147,size=1691143372800
sd=sd49,lun=/dev/vg1/asu148,size=1691143372800
sd=sd50,lun=/dev/vg1/asu149,size=1691143372800
sd=sd51,lun=/dev/vg1/asu150,size=1691143372800
sd=sd52,lun=/dev/vg1/asu151,size=1691143372800
sd=sd53,lun=/dev/vg1/asu152,size=1691143372800
sd=sd54,lun=/dev/vg1/asu153,size=1691143372800
sd=sd55,lun=/dev/vg1/asu154,size=1691143372800
```

```
  sd=sd56,lun=/dev/vg1/asu155,size=1691143372800
  sd=sd57,lun=/dev/vg1/asu156,size=1691143372800
  sd=sd58,lun=/dev/vg1/asu157,size=1691143372800
  sd=sd59,lun=/dev/vg1/asu158,size=1691143372800
  sd=sd60,lun=/dev/vg1/asu159,size=1691143372800
  sd=sd61,lun=/dev/vg1/asu160,size=1691143372800
  sd=sd62,lun=/dev/vg1/asu161,size=1691143372800
  sd=sd63,lun=/dev/vg1/asu162,size=1691143372800
  sd=sd64,lun=/dev/vg1/asu163,size=1691143372800

  sd=sd65,lun=/dev/vg2/asu200,size=1691143372800
  sd=sd66,lun=/dev/vg2/asu201,size=1691143372800
  sd=sd67,lun=/dev/vg2/asu202,size=1691143372800
  sd=sd68,lun=/dev/vg2/asu203,size=1691143372800
  sd=sd69,lun=/dev/vg2/asu204,size=1691143372800
  sd=sd70,lun=/dev/vg2/asu205,size=1691143372800
  sd=sd71,lun=/dev/vg2/asu206,size=1691143372800
  sd=sd72,lun=/dev/vg2/asu207,size=1691143372800
  sd=sd73,lun=/dev/vg2/asu208,size=1691143372800
  sd=sd74,lun=/dev/vg2/asu209,size=1691143372800
  sd=sd75,lun=/dev/vg2/asu210,size=1691143372800
  sd=sd76,lun=/dev/vg2/asu211,size=1691143372800
  sd=sd77,lun=/dev/vg2/asu212,size=1691143372800
  sd=sd78,lun=/dev/vg2/asu213,size=1691143372800
  sd=sd79,lun=/dev/vg2/asu214,size=1691143372800
  sd=sd80,lun=/dev/vg2/asu215,size=1691143372800
  sd=sd81,lun=/dev/vg2/asu216,size=1691143372800
  sd=sd82,lun=/dev/vg2/asu217,size=1691143372800
  sd=sd83,lun=/dev/vg2/asu218,size=1691143372800
  sd=sd84,lun=/dev/vg2/asu219,size=1691143372800
  sd=sd85,lun=/dev/vg2/asu220,size=1691143372800
  sd=sd86,lun=/dev/vg2/asu221,size=1691143372800
  sd=sd87,lun=/dev/vg2/asu222,size=1691143372800
  sd=sd88,lun=/dev/vg2/asu223,size=1691143372800
  sd=sd89,lun=/dev/vg2/asu224,size=1691143372800
  sd=sd90,lun=/dev/vg2/asu225,size=1691143372800
  sd=sd91,lun=/dev/vg2/asu226,size=1691143372800
  sd=sd92,lun=/dev/vg2/asu227,size=1691143372800
  sd=sd93,lun=/dev/vg2/asu228,size=1691143372800
  sd=sd94,lun=/dev/vg2/asu229,size=1691143372800
  sd=sd95,lun=/dev/vg2/asu230,size=1691143372800
  sd=sd96,lun=/dev/vg2/asu231,size=1691143372800
  sd=sd97,lun=/dev/vg2/asu232,size=1691143372800
  sd=sd98,lun=/dev/vg2/asu233,size=1691143372800
  sd=sd99,lun=/dev/vg2/asu234,size=1691143372800
  sd=sd100,lun=/dev/vg2/asu235,size=1691143372800
  sd=sd101,lun=/dev/vg2/asu236,size=1691143372800
  sd=sd102,lun=/dev/vg2/asu237,size=1691143372800
  sd=sd103,lun=/dev/vg2/asu238,size=1691143372800
  sd=sd104,lun=/dev/vg2/asu239,size=1691143372800
  sd=sd105,lun=/dev/vg2/asu240,size=1691143372800
  sd=sd106,lun=/dev/vg2/asu241,size=1691143372800
  sd=sd107,lun=/dev/vg2/asu242,size=1691143372800
  sd=sd108,lun=/dev/vg2/asu243,size=1691143372800
  sd=sd109,lun=/dev/vg2/asu244,size=1691143372800
  sd=sd110,lun=/dev/vg2/asu245,size=1691143372800
  sd=sd111,lun=/dev/vg2/asu246,size=1691143372800
  sd=sd112,lun=/dev/vg2/asu247,size=1691143372800
  sd=sd113,lun=/dev/vg2/asu248,size=1691143372800
  sd=sd114,lun=/dev/vg2/asu249,size=1691143372800
  sd=sd115,lun=/dev/vg2/asu250,size=1691143372800
  sd=sd116,lun=/dev/vg2/asu251,size=1691143372800
  sd=sd117,lun=/dev/vg2/asu252,size=1691143372800
  sd=sd118,lun=/dev/vg2/asu253,size=1691143372800
  sd=sd119,lun=/dev/vg2/asu254,size=1691143372800
  sd=sd120,lun=/dev/vg2/asu255,size=1691143372800
  sd=sd121,lun=/dev/vg2/asu256,size=1691143372800
  sd=sd122,lun=/dev/vg2/asu257,size=1691143372800
  sd=sd123,lun=/dev/vg2/asu258,size=1691143372800
  sd=sd124,lun=/dev/vg2/asu259,size=1691143372800
  sd=sd125,lun=/dev/vg2/asu260,size=1691143372800
  sd=sd126,lun=/dev/vg2/asu261,size=1691143372800
  sd=sd127,lun=/dev/vg2/asu262,size=1691143372800
  sd=sd128,lun=/dev/vg2/asu263,size=1691143372800

  sd=sd129,lun=/dev/vg3/asu300,size=1503238553600
  sd=sd130,lun=/dev/vg3/asu301,size=1503238553600
  sd=sd131,lun=/dev/vg3/asu302,size=1503238553600
  sd=sd132,lun=/dev/vg3/asu303,size=1503238553600
  sd=sd133,lun=/dev/vg3/asu304,size=1503238553600
  sd=sd134,lun=/dev/vg3/asu305,size=1503238553600
  sd=sd135,lun=/dev/vg3/asu306,size=1503238553600
  sd=sd136,lun=/dev/vg3/asu307,size=1503238553600
  sd=sd137,lun=/dev/vg3/asu308,size=1503238553600
  sd=sd138,lun=/dev/vg3/asu309,size=1503238553600
  sd=sd139,lun=/dev/vg3/asu310,size=1503238553600
  sd=sd140,lun=/dev/vg3/asu311,size=1503238553600
  sd=sd141,lun=/dev/vg3/asu312,size=1503238553600
  sd=sd142,lun=/dev/vg3/asu313,size=1503238553600
  sd=sd143,lun=/dev/vg3/asu314,size=1503238553600
  sd=sd144,lun=/dev/vg3/asu315,size=1503238553600

  sd=default,host=coltrane2
  sd=sd1,lun=/dev/vg1/asu100,size=1691143372800
  sd=sd2,lun=/dev/vg1/asu101,size=1691143372800
  sd=sd3,lun=/dev/vg1/asu102,size=1691143372800
  sd=sd4,lun=/dev/vg1/asu103,size=1691143372800
  sd=sd5,lun=/dev/vg1/asu104,size=1691143372800
  sd=sd6,lun=/dev/vg1/asu105,size=1691143372800
  sd=sd7,lun=/dev/vg1/asu106,size=1691143372800
  sd=sd8,lun=/dev/vg1/asu107,size=1691143372800
  sd=sd9,lun=/dev/vg1/asu108,size=1691143372800
  sd=sd10,lun=/dev/vg1/asu109,size=1691143372800
  sd=sd11,lun=/dev/vg1/asu110,size=1691143372800
  sd=sd12,lun=/dev/vg1/asu111,size=1691143372800
  sd=sd13,lun=/dev/vg1/asu112,size=1691143372800
  sd=sd14,lun=/dev/vg1/asu113,size=1691143372800
  sd=sd15,lun=/dev/vg1/asu114,size=1691143372800
  sd=sd16,lun=/dev/vg1/asu115,size=1691143372800
  sd=sd17,lun=/dev/vg1/asu116,size=1691143372800
  sd=sd18,lun=/dev/vg1/asu117,size=1691143372800
  sd=sd19,lun=/dev/vg1/asu118,size=1691143372800
  sd=sd20,lun=/dev/vg1/asu119,size=1691143372800
  sd=sd21,lun=/dev/vg1/asu120,size=1691143372800
  sd=sd22,lun=/dev/vg1/asu121,size=1691143372800
  sd=sd23,lun=/dev/vg1/asu122,size=1691143372800
  sd=sd24,lun=/dev/vg1/asu123,size=1691143372800
  sd=sd25,lun=/dev/vg1/asu124,size=1691143372800
  sd=sd26,lun=/dev/vg1/asu125,size=1691143372800
  sd=sd27,lun=/dev/vg1/asu126,size=1691143372800
  sd=sd28,lun=/dev/vg1/asu127,size=1691143372800
  sd=sd29,lun=/dev/vg1/asu128,size=1691143372800
  sd=sd30,lun=/dev/vg1/asu129,size=1691143372800
  sd=sd31,lun=/dev/vg1/asu130,size=1691143372800
  sd=sd32,lun=/dev/vg1/asu131,size=1691143372800
  sd=sd33,lun=/dev/vg1/asu132,size=1691143372800
```

```
sd=sd34,lun=/dev/vg1/asu133,size=1691143372800
sd=sd35,lun=/dev/vg1/asu134,size=1691143372800
sd=sd36,lun=/dev/vg1/asu135,size=1691143372800
sd=sd37,lun=/dev/vg1/asu136,size=1691143372800
sd=sd38,lun=/dev/vg1/asu137,size=1691143372800
sd=sd39,lun=/dev/vg1/asu138,size=1691143372800
sd=sd40,lun=/dev/vg1/asu139,size=1691143372800
sd=sd41,lun=/dev/vg1/asu140,size=1691143372800
sd=sd42,lun=/dev/vg1/asu141,size=1691143372800
sd=sd43,lun=/dev/vg1/asu142,size=1691143372800
sd=sd44,lun=/dev/vg1/asu143,size=1691143372800
sd=sd45,lun=/dev/vg1/asu144,size=1691143372800
sd=sd46,lun=/dev/vg1/asu145,size=1691143372800
sd=sd47,lun=/dev/vg1/asu146,size=1691143372800
sd=sd48,lun=/dev/vg1/asu147,size=1691143372800
sd=sd49,lun=/dev/vg1/asu148,size=1691143372800
sd=sd50,lun=/dev/vg1/asu149,size=1691143372800
sd=sd51,lun=/dev/vg1/asu150,size=1691143372800
sd=sd52,lun=/dev/vg1/asu151,size=1691143372800
sd=sd53,lun=/dev/vg1/asu152,size=1691143372800
sd=sd54,lun=/dev/vg1/asu153,size=1691143372800
sd=sd55,lun=/dev/vg1/asu154,size=1691143372800
sd=sd56,lun=/dev/vg1/asu155,size=1691143372800
sd=sd57,lun=/dev/vg1/asu156,size=1691143372800
sd=sd58,lun=/dev/vg1/asu157,size=1691143372800
sd=sd59,lun=/dev/vg1/asu158,size=1691143372800
sd=sd60,lun=/dev/vg1/asu159,size=1691143372800
sd=sd61,lun=/dev/vg1/asu160,size=1691143372800
sd=sd62,lun=/dev/vg1/asu161,size=1691143372800
sd=sd63,lun=/dev/vg1/asu162,size=1691143372800
sd=sd64,lun=/dev/vg1/asu163,size=1691143372800

sd=sd65,lun=/dev/vg2/asu200,size=1691143372800
sd=sd66,lun=/dev/vg2/asu201,size=1691143372800
sd=sd67,lun=/dev/vg2/asu202,size=1691143372800
sd=sd68,lun=/dev/vg2/asu203,size=1691143372800
sd=sd69,lun=/dev/vg2/asu204,size=1691143372800
sd=sd70,lun=/dev/vg2/asu205,size=1691143372800
sd=sd71,lun=/dev/vg2/asu206,size=1691143372800
sd=sd72,lun=/dev/vg2/asu207,size=1691143372800
sd=sd73,lun=/dev/vg2/asu208,size=1691143372800
sd=sd74,lun=/dev/vg2/asu209,size=1691143372800
sd=sd75,lun=/dev/vg2/asu210,size=1691143372800
sd=sd76,lun=/dev/vg2/asu211,size=1691143372800
sd=sd77,lun=/dev/vg2/asu212,size=1691143372800
sd=sd78,lun=/dev/vg2/asu213,size=1691143372800
sd=sd79,lun=/dev/vg2/asu214,size=1691143372800
sd=sd80,lun=/dev/vg2/asu215,size=1691143372800
sd=sd81,lun=/dev/vg2/asu216,size=1691143372800
sd=sd82,lun=/dev/vg2/asu217,size=1691143372800
sd=sd83,lun=/dev/vg2/asu218,size=1691143372800
sd=sd84,lun=/dev/vg2/asu219,size=1691143372800
sd=sd85,lun=/dev/vg2/asu220,size=1691143372800
sd=sd86,lun=/dev/vg2/asu221,size=1691143372800
sd=sd87,lun=/dev/vg2/asu222,size=1691143372800
sd=sd88,lun=/dev/vg2/asu223,size=1691143372800
sd=sd89,lun=/dev/vg2/asu224,size=1691143372800
sd=sd90,lun=/dev/vg2/asu225,size=1691143372800
sd=sd91,lun=/dev/vg2/asu226,size=1691143372800
sd=sd92,lun=/dev/vg2/asu227,size=1691143372800
sd=sd93,lun=/dev/vg2/asu228,size=1691143372800
sd=sd94,lun=/dev/vg2/asu229,size=1691143372800
sd=sd95,lun=/dev/vg2/asu230,size=1691143372800
```

```
sd=sd96,lun=/dev/vg2/asu231,size=1691143372800
sd=sd97,lun=/dev/vg2/asu232,size=1691143372800
sd=sd98,lun=/dev/vg2/asu233,size=1691143372800
sd=sd99,lun=/dev/vg2/asu234,size=1691143372800
sd=sd100,lun=/dev/vg2/asu235,size=1691143372800
sd=sd101,lun=/dev/vg2/asu236,size=1691143372800
sd=sd102,lun=/dev/vg2/asu237,size=1691143372800
sd=sd103,lun=/dev/vg2/asu238,size=1691143372800
sd=sd104,lun=/dev/vg2/asu239,size=1691143372800
sd=sd105,lun=/dev/vg2/asu240,size=1691143372800
sd=sd106,lun=/dev/vg2/asu241,size=1691143372800
sd=sd107,lun=/dev/vg2/asu242,size=1691143372800
sd=sd108,lun=/dev/vg2/asu243,size=1691143372800
sd=sd109,lun=/dev/vg2/asu244,size=1691143372800
sd=sd110,lun=/dev/vg2/asu245,size=1691143372800
sd=sd111,lun=/dev/vg2/asu246,size=1691143372800
sd=sd112,lun=/dev/vg2/asu247,size=1691143372800
sd=sd113,lun=/dev/vg2/asu248,size=1691143372800
sd=sd114,lun=/dev/vg2/asu249,size=1691143372800
sd=sd115,lun=/dev/vg2/asu250,size=1691143372800
sd=sd116,lun=/dev/vg2/asu251,size=1691143372800
sd=sd117,lun=/dev/vg2/asu252,size=1691143372800
sd=sd118,lun=/dev/vg2/asu253,size=1691143372800
sd=sd119,lun=/dev/vg2/asu254,size=1691143372800
sd=sd120,lun=/dev/vg2/asu255,size=1691143372800
sd=sd121,lun=/dev/vg2/asu256,size=1691143372800
sd=sd122,lun=/dev/vg2/asu257,size=1691143372800
sd=sd123,lun=/dev/vg2/asu258,size=1691143372800
sd=sd124,lun=/dev/vg2/asu259,size=1691143372800
sd=sd125,lun=/dev/vg2/asu260,size=1691143372800
sd=sd126,lun=/dev/vg2/asu261,size=1691143372800
sd=sd127,lun=/dev/vg2/asu262,size=1691143372800
sd=sd128,lun=/dev/vg2/asu263,size=1691143372800

sd=sd129,lun=/dev/vg3/asu300,size=1503238553600
sd=sd130,lun=/dev/vg3/asu301,size=1503238553600
sd=sd131,lun=/dev/vg3/asu302,size=1503238553600
sd=sd132,lun=/dev/vg3/asu303,size=1503238553600
sd=sd133,lun=/dev/vg3/asu304,size=1503238553600
sd=sd134,lun=/dev/vg3/asu305,size=1503238553600
sd=sd135,lun=/dev/vg3/asu306,size=1503238553600
sd=sd136,lun=/dev/vg3/asu307,size=1503238553600
sd=sd137,lun=/dev/vg3/asu308,size=1503238553600
sd=sd138,lun=/dev/vg3/asu309,size=1503238553600
sd=sd139,lun=/dev/vg3/asu310,size=1503238553600
sd=sd140,lun=/dev/vg3/asu311,size=1503238553600
sd=sd141,lun=/dev/vg3/asu312,size=1503238553600
sd=sd142,lun=/dev/vg3/asu313,size=1503238553600
sd=sd143,lun=/dev/vg3/asu314,size=1503238553600
sd=sd144,lun=/dev/vg3/asu315,size=1503238553600


maxlatestart=1
reportinginterval=5
segmentlength=512m
```

SPC-2 Persistence Test Run 1 *(write phase)*

```
common_spc2_persist_commands

rd=default,rampup=360,periods=180,measurement=300,runout=0,rampdown=0,buffers=1
```

```
rd=default,rdpct=0,xfersize=1024k
rd=TR1-101s_SPC-2-persist-w,streams=410
```

SPC-2 Persistence Test Run 2 *(read phase)*

```
common_spc2_persist_commands

maxpersistenceerrors=10

rd=default,buffers=1,rdpct=100,xfersize=1024k
rd=TR1-5s_SPC-2-persist-r
```

APPENDIX E: SPC-1 WORKLOAD GENERATOR INPUT PARAMETERS

The 'master' script, **run.sh**, was invoked to execute the required ASU pre-fill, the 3 "Ramp-Up" Range Test Runs, Primary Metrics Test *(Sustainability Test Phase, IOPS Test Phase, and Response Time Ramp Test Phase)*, Repeatability Test *(Repeatability Test Phase 1 and Repeatability Test Phase 2)*, a reduced level SPC-1 Persistence Test Run 1 *(write phase)* and SPC-2 Persistence Test Run 1 *(write phase)* in an uninterrupted sequence.

The script also calculates the start and duration time for the SmartTier functionality. The **SmartTier_config.sh** script is invoked set the start and duration time for "hot" data migration to the configured SSD storage devices.

After the above test sequence completed, the script pauses until the required TSC power off/power on cycle is completed then executes SPC-2 Persistence Test Run 2 *(read phase)*.

The **run.sh** script also included the appropriate commands to capture the detailed TSC profile listings required for a Remote Audit.

run.sh

```
#!/bin/sh

JAVA="/usr/java/jre1.7.0_06/bin/java -Xms8192m -Xmx8192m -Xss512k"
EXEDIR=/root/V3_6800

expect shstorage.tcl > profile1_storage.log

date > profile1_volume.log
lvdisplay >> profile1_volume.log
date >> profile1_volume.log

echo "ASU prefill started......"
../vdbench/vdbench -f /root/V3_6800/prefilling.cfg -o /root/V3_6800/PreFill
echo "ASU prefill complete....."

time_now=`date +%s`
start_monitor_time=$(($time_now+1000))
start_relocation_time=$(($start_monitor_time+14460))
day_of_week_monitor=$(date +%a -d "1970-01-01 UTC $start_monitor_time seconds")
day_of_week_relocation=$(date +%a -d "1970-01-01 UTC $start_relocation_time
seconds")
start_monitor_time=$(date +%H:%M -d "1970-01-01 UTC $start_monitor_time seconds")
monitor_duration=04:00
start_relocation_time=$(date +%H:%M -d "1970-01-01 UTC $start_relocation_time
seconds")
relocation_duration=06:00

./SmartTier_config.sh $start_monitor_time $monitor_duration $start_relocation_time
$relocation_duration $day_of_week_monitor $day_of_week_relocation

N=1
for host in host1 host2 host3 host4 host5 host6
do
  ssh $host rm -rf $EXEDIR/output
  ssh $host rm -rf $EXEDIR/config
  ssh $host mkdir $EXEDIR/output
  ssh $host mkdir $EXEDIR/config
  for((i=1;i<=22;i++))
  do
```

```
    echo "start slave$N on $host"
    echo "master=host1" > $EXEDIR/config/slave$N.cfg
    echo "host=slave$N" >> $EXEDIR/config/slave$N.cfg


echo "sd=asu1_0,lun=/dev/vg1/asu100,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_1,lun=/dev/vg1/asu101,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_2,lun=/dev/vg1/asu102,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_3,lun=/dev/vg1/asu103,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_4,lun=/dev/vg1/asu104,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_5,lun=/dev/vg1/asu105,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_6,lun=/dev/vg1/asu106,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_7,lun=/dev/vg1/asu107,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_8,lun=/dev/vg1/asu108,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_9,lun=/dev/vg1/asu109,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_10,lun=/dev/vg1/asu110,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_11,lun=/dev/vg1/asu111,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_12,lun=/dev/vg1/asu112,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_13,lun=/dev/vg1/asu113,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_14,lun=/dev/vg1/asu114,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_15,lun=/dev/vg1/asu115,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_16,lun=/dev/vg1/asu116,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_17,lun=/dev/vg1/asu117,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_18,lun=/dev/vg1/asu118,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_19,lun=/dev/vg1/asu119,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_20,lun=/dev/vg1/asu120,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_21,lun=/dev/vg1/asu121,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_22,lun=/dev/vg1/asu122,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_23,lun=/dev/vg1/asu123,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_24,lun=/dev/vg1/asu124,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_25,lun=/dev/vg1/asu125,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_26,lun=/dev/vg1/asu126,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_27,lun=/dev/vg1/asu127,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_28,lun=/dev/vg1/asu128,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
```

```
echo "sd=asu1_29,lun=/dev/vg1/asu129,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_30,lun=/dev/vg1/asu130,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_31,lun=/dev/vg1/asu131,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_32,lun=/dev/vg1/asu132,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_33,lun=/dev/vg1/asu133,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_34,lun=/dev/vg1/asu134,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_35,lun=/dev/vg1/asu135,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_36,lun=/dev/vg1/asu136,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_37,lun=/dev/vg1/asu137,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_38,lun=/dev/vg1/asu138,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_39,lun=/dev/vg1/asu139,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_40,lun=/dev/vg1/asu140,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_41,lun=/dev/vg1/asu141,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_42,lun=/dev/vg1/asu142,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_43,lun=/dev/vg1/asu143,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_44,lun=/dev/vg1/asu144,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_45,lun=/dev/vg1/asu145,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_46,lun=/dev/vg1/asu146,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_47,lun=/dev/vg1/asu147,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_48,lun=/dev/vg1/asu148,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_49,lun=/dev/vg1/asu149,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_50,lun=/dev/vg1/asu150,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_51,lun=/dev/vg1/asu151,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_52,lun=/dev/vg1/asu152,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_53,lun=/dev/vg1/asu153,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_54,lun=/dev/vg1/asu154,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_55,lun=/dev/vg1/asu155,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_56,lun=/dev/vg1/asu156,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_57,lun=/dev/vg1/asu157,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_58,lun=/dev/vg1/asu158,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_59,lun=/dev/vg1/asu159,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
```

```
echo "sd=asu1_60,lun=/dev/vg1/asu160,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_61,lun=/dev/vg1/asu161,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_62,lun=/dev/vg1/asu162,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_63,lun=/dev/vg1/asu163,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg

echo "sd=asu2_0,lun=/dev/vg2/asu200,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_1,lun=/dev/vg2/asu201,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_2,lun=/dev/vg2/asu202,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_3,lun=/dev/vg2/asu203,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_4,lun=/dev/vg2/asu204,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_5,lun=/dev/vg2/asu205,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_6,lun=/dev/vg2/asu206,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_7,lun=/dev/vg2/asu207,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_8,lun=/dev/vg2/asu208,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_9,lun=/dev/vg2/asu209,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_10,lun=/dev/vg2/asu210,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_11,lun=/dev/vg2/asu211,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_12,lun=/dev/vg2/asu212,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_13,lun=/dev/vg2/asu213,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_14,lun=/dev/vg2/asu214,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_15,lun=/dev/vg2/asu215,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_16,lun=/dev/vg2/asu216,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_17,lun=/dev/vg2/asu217,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_18,lun=/dev/vg2/asu218,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_19,lun=/dev/vg2/asu219,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_20,lun=/dev/vg2/asu220,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_21,lun=/dev/vg2/asu221,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_22,lun=/dev/vg2/asu222,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_23,lun=/dev/vg2/asu223,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_24,lun=/dev/vg2/asu224,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_25,lun=/dev/vg2/asu225,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_26,lun=/dev/vg2/asu226,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
```

```
echo "sd=asu2_27,lun=/dev/vg2/asu227,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_28,lun=/dev/vg2/asu228,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_29,lun=/dev/vg2/asu229,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_30,lun=/dev/vg2/asu230,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_31,lun=/dev/vg2/asu231,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_32,lun=/dev/vg2/asu232,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_33,lun=/dev/vg2/asu233,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_34,lun=/dev/vg2/asu234,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_35,lun=/dev/vg2/asu235,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_36,lun=/dev/vg2/asu236,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_37,lun=/dev/vg2/asu237,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_38,lun=/dev/vg2/asu238,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_39,lun=/dev/vg2/asu239,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_40,lun=/dev/vg2/asu240,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_41,lun=/dev/vg2/asu241,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_42,lun=/dev/vg2/asu242,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_43,lun=/dev/vg2/asu243,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_44,lun=/dev/vg2/asu244,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_45,lun=/dev/vg2/asu245,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_46,lun=/dev/vg2/asu246,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_47,lun=/dev/vg2/asu247,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_48,lun=/dev/vg2/asu248,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_49,lun=/dev/vg2/asu249,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_50,lun=/dev/vg2/asu250,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_51,lun=/dev/vg2/asu251,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_52,lun=/dev/vg2/asu252,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_53,lun=/dev/vg2/asu253,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_54,lun=/dev/vg2/asu254,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_55,lun=/dev/vg2/asu255,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_56,lun=/dev/vg2/asu256,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_57,lun=/dev/vg2/asu257,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
```

```
echo "sd=asu2_58,lun=/dev/vg2/asu258,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_59,lun=/dev/vg2/asu259,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_60,lun=/dev/vg2/asu260,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_61,lun=/dev/vg2/asu261,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_62,lun=/dev/vg2/asu262,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_63,lun=/dev/vg2/asu263,size=1691143372800" >>
$EXEDIR/config/slave$N.cfg

echo "sd=asu3_0,lun=/dev/vg3/asu300,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_1,lun=/dev/vg3/asu301,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_2,lun=/dev/vg3/asu302,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_3,lun=/dev/vg3/asu303,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_4,lun=/dev/vg3/asu304,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_5,lun=/dev/vg3/asu305,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_6,lun=/dev/vg3/asu306,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_7,lun=/dev/vg3/asu307,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_8,lun=/dev/vg3/asu308,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_9,lun=/dev/vg3/asu309,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_10,lun=/dev/vg3/asu310,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_11,lun=/dev/vg3/asu311,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_12,lun=/dev/vg3/asu312,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_13,lun=/dev/vg3/asu313,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_14,lun=/dev/vg3/asu314,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_15,lun=/dev/vg3/asu315,size=1503238553600" >>
$EXEDIR/config/slave$N.cfg


scp $EXEDIR/config/slave$N.cfg $host:$EXEDIR/config/slave$N.cfg
    ssh $host "$JAVA -cp $EXEDIR/../spc1 spc1 -f $EXEDIR/config/slave$N.cfg -o
$EXEDIR/output/slave$N" > /dev/null &
    N=$[N+1]
  done
done


rm -rf spc1.cfg
cp range.cfg spc1.cfg

$JAVA -cp ../spc1 range -b 3255 -t 14400
mv rangetest rangetest1
$JAVA -cp ../spc1  range -b 6510 -t 14400
mv rangetest rangetest2
$JAVA -cp ../spc1 range -b 9765 -t 10800
```

```
mv rangetest rangetest3


cp metrics.cfg spc1.cfg

$JAVA -cp ../spc1 metrics -b 13020 -t 28800
$JAVA -cp ../spc1 repeat1 -b 13020
$JAVA -cp ../spc1 repeat2 -b 13020

for host in host1 host2 host3 host4 host5 host6
do
  ssh $host killall java
done

rm -rf spc1.cfg
cp persist.cfg spc1.cfg

$JAVA -cp ../spc1 persist1 -b 1302

for host in host2 host3
do
    ssh $host $JAVA -cp $EXEDIR/../spc2 RemoteStart &
done

../spc2/spc2 -f persist1.cfg -o persist1 -init
../spc2/spc2 -f persist1.cfg -o persist1

echo "Power cycle TSC, then Enter to continue"
read

expect shstorage.tcl > profile2_storage.log
date > profile2_volume.log
lvdisplay >> profile2_volume.log
date >> profile2_volume.log

../spc2/spc2 -f persist2.cfg -o persist2 -init
../spc2/spc2 -f persist2.cfg -o persist2
```

SmartTier_config.sh

```
#!/bin/bash


stor=100.124.7.101
stor_user=admin
stor_pswd=Admin@storage1

export LANG=C

echo "configurate SmartTier ..."

expect <<__END_CREATE_LUN
     spawn ssh $stor_user@$stor
     set timeout 60
     expect {
            -re "assword" { send "$stor_pswd\n" }
            -re "yes/no" { send "yes\n"; exp_continue }
     }
     expect ">"
     set start_monitor_time $1
     set monitor_duration $2
     set start_relocation_time $3
     set relocation_duration $4
     set day_of_week_monitor $5
     set day_of_week_relocation $6

     send "change storage_pool relocation_speed relocation_speed=High\r"
     expect ">"

     for { set lunid 0 } { \$lunid <= 31 } { incr lunid } {
            send "change lun lun_id=\$lunid relocation_policy=automatic\r"
            expect ">"
            send "change lun lun_id=[expr \$lunid + 72]
relocation_policy=automatic\r"
            expect ">"
     }

     for { set pool_id 0 } { \$pool_id <= 7 } { incr pool_id } {
            send "change storage_pool relocation_schedule pool_id=\$pool_id
relocation_type=automatic\r"
            expect ">"
            send "create schedule monitor name=m\${pool_id} pool_id=\$pool_id
period=weekly day_of_week=\$day_of_week_monitor start_time=\$start_monitor_time
duration=\$monitor_duration\r"
            expect ">"
            send "create schedule relocate name=r\${pool_id} pool_id=\$pool_id
period=weekly day_of_week=\$day_of_week_relocation
start_time=\$start_relocation_time duration=\$relocation_duration\r"
            expect ">"
          }

     for { set pool_id 24 } { \$pool_id <= 31 } { incr pool_id } {
            send "change storage_pool relocation_schedule pool_id=\$pool_id
relocation_type=automatic\r"
            expect ">"
            send "create schedule monitor name=m\${pool_id} pool_id=\$pool_id
period=weekly day_of_week=\$day_of_week_monitor start_time=\$start_monitor_time
duration=\$monitor_duration\r"
            expect ">"
            send "create schedule relocate name=r\${pool_id} pool_id=\$pool_id
period=weekly day_of_week=\$day_of_week_relocation
start_time=\$start_relocation_time duration=\$relocation_duration\r"
            expect ">"
          }

     send "exit\r"
     expect "(y/n):"
     send "y\r"
        sleep 5
     send "exit\r"
     expect EOF
__END_CREATE_LUN
```

APPENDIX F: THIRD-PARTY QUOTATION

Priced Storage Configuration

netfast® Optimize IT.

Netfast Technology Solutions, Inc.
989, Avenues of America, Fl 12
New York, NY 10018, USA
Voice: (212) 792-5200 , Fax: (212) 213-1152

10/25/2014, Quote Valid:90 Days

No.	Model	Description	Qty	Unit Price($)	Total Price($)
1	Phase				
1.1	Location				
1.1.1	6800 V3 Storage System				
1.1.1.1	Control Module				
	6800V3-512G-AC	6800 V3(6U,Dual Ctrl,AC,512GB,HW Storage System Software,SPE72C0600)	1	41091.00	41,091.00
	6800V3-256G-CTL	Controller Module(2*Intel 6 Cores,256GB Cache)	2	8903.00	17,806.00
1.1.1.2	Disk Enclosure				
	DAE22525U2-1-AC	Disk Enclosure(2U,2.5",AC,SAS Expanding Module)	56	2054.50	115,052.00
1.1.1.3	Hard Disk Drives				
	SAS600-10K-2-V3	600GB 10K RPM SAS Disk Unit(2.5")	1184	457.75	541,976.00
	SLC200-2-02	200GB HSSD SLC SAS Disk Unit(2.5")	160	2888.00	462,080.00
1.1.1.4	IO Interface				
	LPU4FC8V3	4*8Gbps Fibre Channel I/O modules(Total 4 ports)	12	1983.75	23,805.00
	LPU4S12V3	4*12Gbps SAS I/O module(Total 4 ports,MiniSAS HD)	12	741.25	8,895.00
1.1.1.5	Accessory				
	SS-OP-D-LC-M-3	Patchcord,DLC/PC-DLC/PC,Multimode,2mm Parallel,3m	24	11.00	264.00
	HS-SAS-3-01	High Speed Cable,Mini SAS HD Cable,3m,(SFF 8644	1	67.00	67.00
	HS-SAS-5-01	High Speed Cable,Mini SAS HD Cable,5m,(SFF 8644 Plug),(26AWG*4P*2B(S)),(SFF 8644 Plug),Indoor use	18	91.00	1,638.00
	RACK-46U-AC	N610E-22 46U Common AC Storage Rack(include 2 AC power distribution panels)	3	1712.00	5,136.00

Priced Storage Configuration *(continued)*

No.	Model	Description	Qty	Unit Price($)	Total Price($)
1.1.1.6	**HBA**				
	N8GHBA000	QLOGIC QLE2562 HBA Card,PCIE,8Gbps DualPort ,Fiber Channel Multimode LC Optic Interface,English Manual, No Drive CD	12	1000.00	12,000.00
1.1.1.7	**Storage Software**				
	LIC-6800V3-BS	Basic Software License,Include Device Management	1	3281.50	3,281.50
	LIC-6800V3-TIER	SmartTier License	1	11944.50	11,944.50
	LIC-6800V3-PATH	OceanStor HW UltraPath Software License	1	716.25	716.25
Total of Product					**1,245,752.25**
1.1.1.8	**Maintenance Support Service**				
		6800V3 Control Enclosure Implementation Service-Installation Service	1	2484.95	2,484.95
		DAE(6800V3) Implementation Service-Installation Service	56	321.75	18,018.00
		6800V3-Control Enclosure-Warranty Upgrade To Hi-Care Onsite Premier 24x7x4H Engineer Onsite Service	1	9824.10	9,824.10
		DAE（6800V3）-Warranty Upgrade To Hi-Care Onsite Premier 24x7x4H Engineer Onsite Service	56	3784.95	211,957.20
Total of Service (3 years)					**242,284.25**
Total Price					**1,488,036.50**

Notes:Hi-Care Premier On-Site Service include: 7*24 Technical Assistance Center Access. Access to all new software updates and Online Support. 24*7*4 Hours Onsite Hardware Replacement.