SPC BENCHMARK 1™
FULL DISCLOSURE REPORT

HUAWEI TECHNOLOGIES CO., LTD.
HUAWEI OCEANSTOR™ S6800T

SPC-1 V1.12

Submitted for Review: October 21, 2012
Submission Identifier: A00121

First Edition – October 2012

Trademarks

Table of Contents

AUDIT CERTIFICATION

Eric He
Huawei Technologies Co., Ltd.
Tianchen Road 88#
Chengdu, Sichuan, P.R. China 611711

October 19, 2012

The SPC Benchmark 1™ Reported Data listed below for the Huawei OceanStor™ S6800T was produced in compliance with the SPC Benchmark 1™ v1.12 Remote Audit requirements.

SPC Benchmark 1™ v1.12 Reported Data	
Tested Storage Product (TSP) Name: Huawei OceanStor™ S6800T	
Metric	**Reported Result**
SPC-1 IOPS™	250,388.92
SPC-1 Price-Performance	$3.32/SPC-1 IOPS™
Total ASU Capacity	86,73.088 GB
Data Protection Level	Protected *(Mirroring)*
Total TSC Price (including three-year maintenance)	$830,462.06

The following SPC Benchmark 1™ Remote Audit requirements were reviewed and found compliant with 1.12 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.
- An appropriate diagram of the Benchmark Configuration (BC)/Tested Storage Configuration (TSC).

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

AUDIT CERTIFICATION *(CONT.)*

- 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 each Host System 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 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 and 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:

There were no audit notes or exceptions.

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

Date: October 8, 2012

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 S6800T

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 V1.12 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: 2012.10.10.

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/ Eric He – eric.heji@huawei.com Tianchen Road 88# Chengdu, Sichuan, P.R. China 611711 Phone: 86 28 62905595 FAX: 86 28 62905793
Test Sponsor Alternate Contact	Huawei Technologies Co., Ltd. – http://www.huawei.com/en/ Xu Zhong – xuzhong@huawei.com Tianchen Road 88# Chengdu, Sichuan, P.R. China 611711 Phone: 86 28 62905591 FAX: 86 28 62905793
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	V1.12
SPC-1 Workload Generator revision number	V2.3.0
Date Results were first used publicly	October 21, 2012
Date the FDR was submitted to the SPC	October 21, 2012
Date the Priced Storage Configuration is available for shipment to customers	currently available
Date the TSC completed audit certification	October 19, 2012

Tested Storage Product (TSP) Description

Huawei OceanStor S6800T is a new generation of high-end enterprise-class storage system. Based on its industry-leading hardware specifications, the S6800T integrates the advanced technologies of high-density disk design, TurboModule (high-density and hot-swappable I/O modules), and TurboBoost (three-level performance boost), and multi-level data protection, satisfying the application requirements of large OLTP/OLAP databases, high-performance computing, digital media, Internet operation, centralized storage, backup, disaster recovery, and data migration.

Summary of Results

SPC-1 Reported Data	
Tested Storage Product (TSP) Name: Huawei OceanStor™ S6800T	
Metric	**Reported Result**
SPC-1 IOPS™	250,388.92
SPC-1 Price-Performance™	$3.32/SPC-1 IOPS™
Total ASU Capacity	86,973.088 GB
Data Protection Level	Protected *(Mirroring)*
Total TSC Price (including three-year maintenance)	$830,462.06

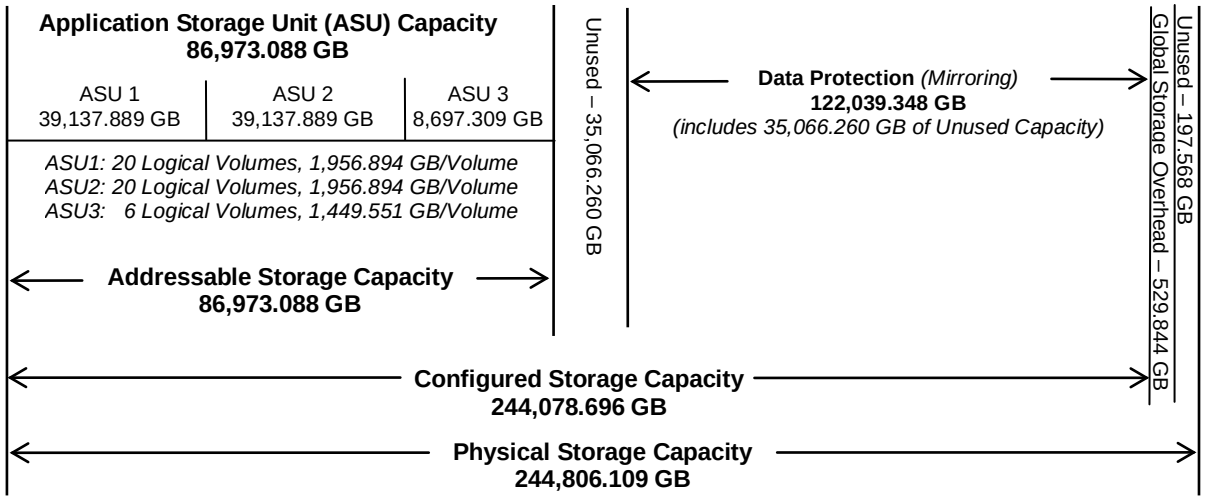
SPC-1 IOPS™ represents the maximum I/O Request Throughput at the 100% load point.

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

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

Storage Capacities, Relationships, and Utilization

The following diagram 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.



SPC-1 Storage Capacity Utilization	
Application Utilization	35.53%
Protected Application Utilization	71.05%
Unused Storage Ratio	28.73%

Application Utilization: Total ASU Capacity *(86,973.088 GB)* divided by Physical Storage Capacity *(244,806.109 GB)*

Protected Application Utilization: Total ASU Capacity *(86,973.088 GB)* plus total Data Protection Capacity *(122,039.348 GB)* minus unused Data Protection Capacity *(35,066.260 GB)* divided by Physical Storage Capacity *(244,806.109 GB)*

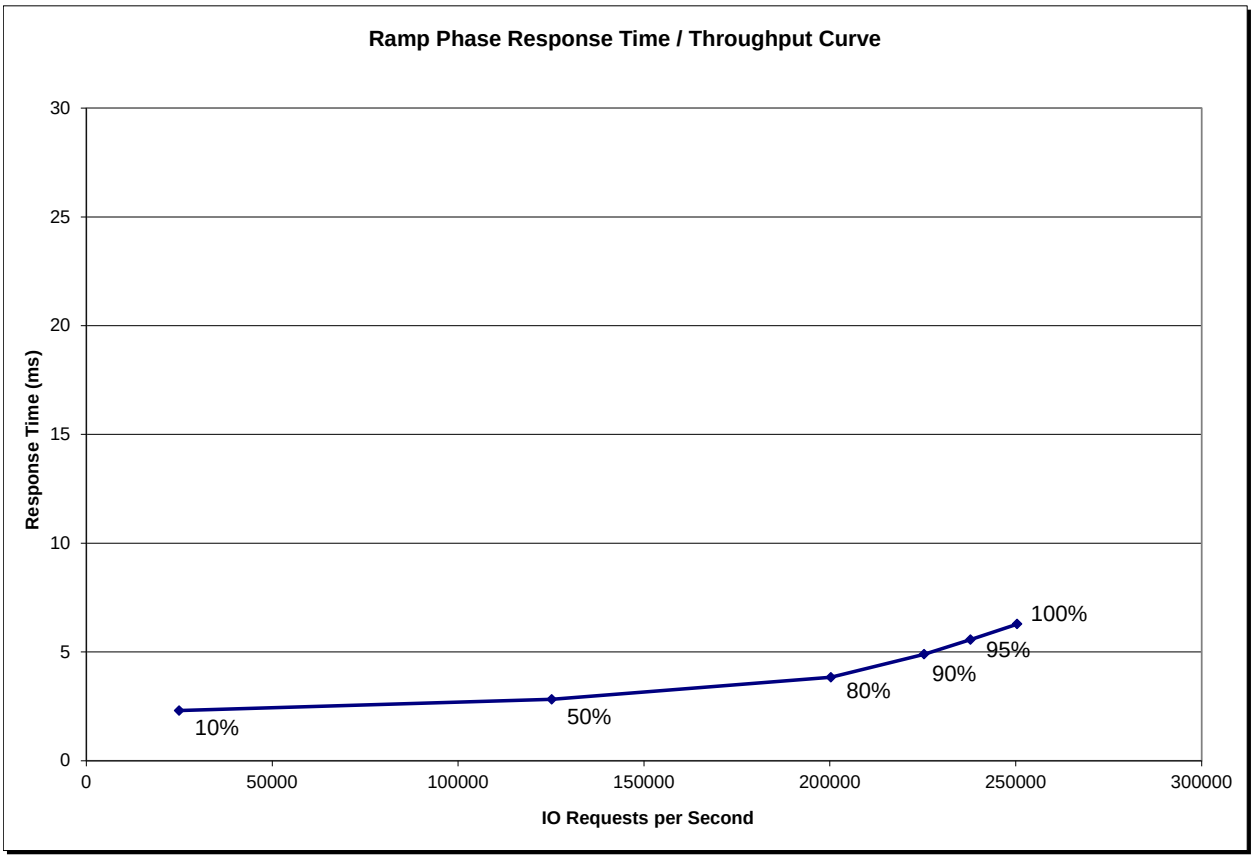
Unused Storage Ratio: Total Unused Capacity *(70,330.089 GB)* divided by Physical Storage Capacity *(244,806.109 GB)* and may not exceed 45%.

Detailed information for the various storage capacities and utilizations is available on pages 21-22.

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	25,003.10	125,210.54	200,298.32	225,358.02	237,849.52	250,388.92
Average Response Time (ms):						
All ASUs	2.30	2.82	3.83	4.89	5.57	6.28
ASU-1	3.05	3.59	4.56	5.44	6.24	6.98
ASU-2	3.03	4.00	5.47	6.74	7.98	8.88
ASU-3	0.40	0.68	1.57	2.93	3.08	3.67
Reads	5.25	6.16	7.45	8.81	10.12	11.33
Writes	0.38	0.65	1.48	2.34	2.60	3.00

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 Pricing

Part Number	Description	Quantity	Unit Price($)	Total Price($)
Control module				**75,196.66**
SPE61C0200-S63-2C192G-DC-BASE	SPE61C0200-64 Controller Enclosure(Dual Controller,AC,384GB Cache,with UPS Cache Protected Module,without Front-End & Back-End Port,with HS Storage Array Control System Software)	1	75,196.66	75,196.66
Disk Enclosure				**102,576.32**
DAE12435U4-AC	DAE12435U4-03 Disk Enclosure(4U,3.5",AC,SAS Expansion Module,without Disk Unit,with HS SAS in Band Management Software, includes 2 SAS cables)	32	3,205.51	102,576.32
Hard Disk Drives				**371,032.32**
SAS300-15K	300GB 15K RPM SAS Disk Unit(3.5")	720	465.65	335,268.00
SAS600-15K	600GB 15K RPM SAS Disk Unit(3.5")	48	745.09	35,764.32
IO Interface				**13,149.25**
LPU4F8	4*8Gbps Fibre Channel I/O modules (Total 4 ports, includes four 8Gbps SFPs)	4	1,218.20	4,872.78
LPU2S6	2*24Gbps SAS-wide I/O modules(Total 2 ports)	8	1,034.56	8,276.47
Accessory				**1,153.00**
SS-OP-D-LC-M-3	Patchcord,DLC/PC-DLC/PC,Multimode,2mm Parallel,3m	16	11.00	176.00
MINI-SAS-3	Purchased Cable,MiniSAS Cable,Key246,3m	1	67.00	67.00
MINI-SAS-6	Outsourcing Cable,External Mini SAS Cable,6m,External Mini SAS 26 Pin Plug,24AWG*8P,External Mini SAS 26 Pin Plug	10	91.00	910.00
Storage management software				**8,555.91**
LIC-S6A-ISM02-V1R5	HS Integrated Storage Manager-Device Management License for OceanStor S6800T(V100R005)	1	6,617.39	6,617.39
LIC-UltraPath02-V1R5	OceanStor HS UltraPath Software License	1	1,938.51	1,938.51
Third Party				**13,584.00**
N8GHBA000	QLOGIC QLE2562 HBA Card,PCIE,8Gbps DualPort ,Fiber Channel Multimode LC Optic Interface,English Manual,Driver CD	8	1,698.00	13,584.00
Total of Product				**585,247.46**
Maintenance Support Service				
Hi-Care Premier On-Site Service (3 years)		1	216,753.00	216,753.00
300GB 15K RPM SAS Disk Unit(3.5") upgrade to On-Site Serice (3 years)		720	35.72	25,718.40
600GB 15K RPM SAS Disk Unit(3.5") upgrade to On-Site Service (3 years)		48	57.15	2,743.20
Total of Service (3 years)				**245,214.60**
Total Price				**830,462.06**
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 with 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 82 *("Appendix F: Third-Party Quotation")* for a copy of the third-party reseller quotation.

Priced Storage Configuration Diagram

Huawei OceanStor™ S6800T

dual controllers - Active-Active

192 GB cache per controller (384 GB total)

2 – FC 4-port I/O modules per controller (8 Gbps, 4 modules total)

4 – 2x24 Gbps SAS-wide I/O modules per controller (8 modules total)

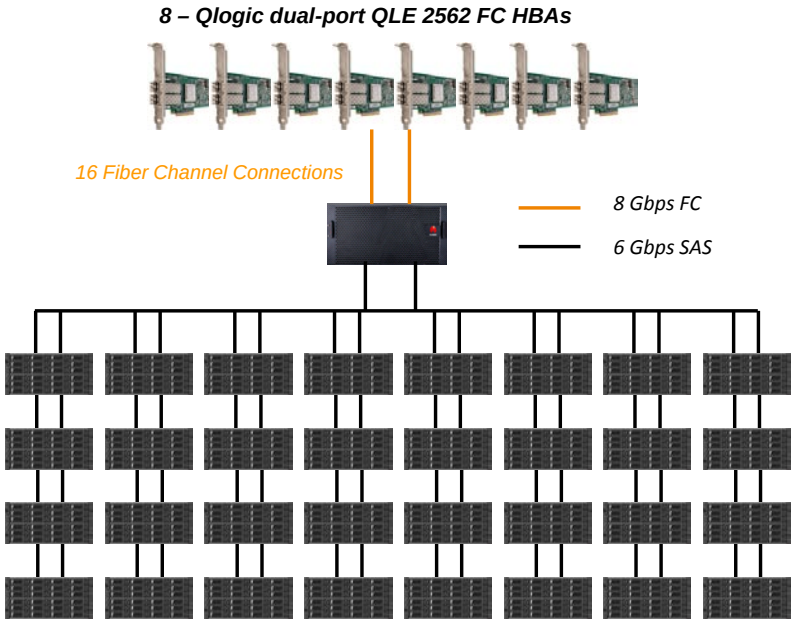
32 - disk enclosures

24 - disk drives per enclosure

768 - 15K RPM SAS disk drives

48 - 600GB 15K RPM SAS disk drives

720 - 300GB 15K RPM SAS disk drives



Priced Storage Configuration Components

Priced Storage Configuration:
8 – Qlogic dual-port QLE2562 FC HBAs
Huawei OceanStor™ S6800T **dual controllers – Active-Active** 192 GB cache per controller *(384 GB total)* 2 – FC 4-port I/O modules per controller *(8 Gbps, 4 modules total)* 8 – 8 Gbps front-end connections per controller *(16 total, 16 used)* 4 – 2x24 Gbps SAS-wide I/O modules per controller *(8 total)* *(2 ports per modules)* 8 –2x24 Gbps SAS backend connections per controller *(16 total, 16 used)*
32 – Disk Enclosures 24 – 3.5” HD slots per enclosure 24 – disk drives in each enclosures
768 – 15K RPM SAS disk drives 720 – 300 GB disk drives 48 – 600 GB disk drives

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 18 *(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) was configured with local storage and, as such, did not employ a storage network.

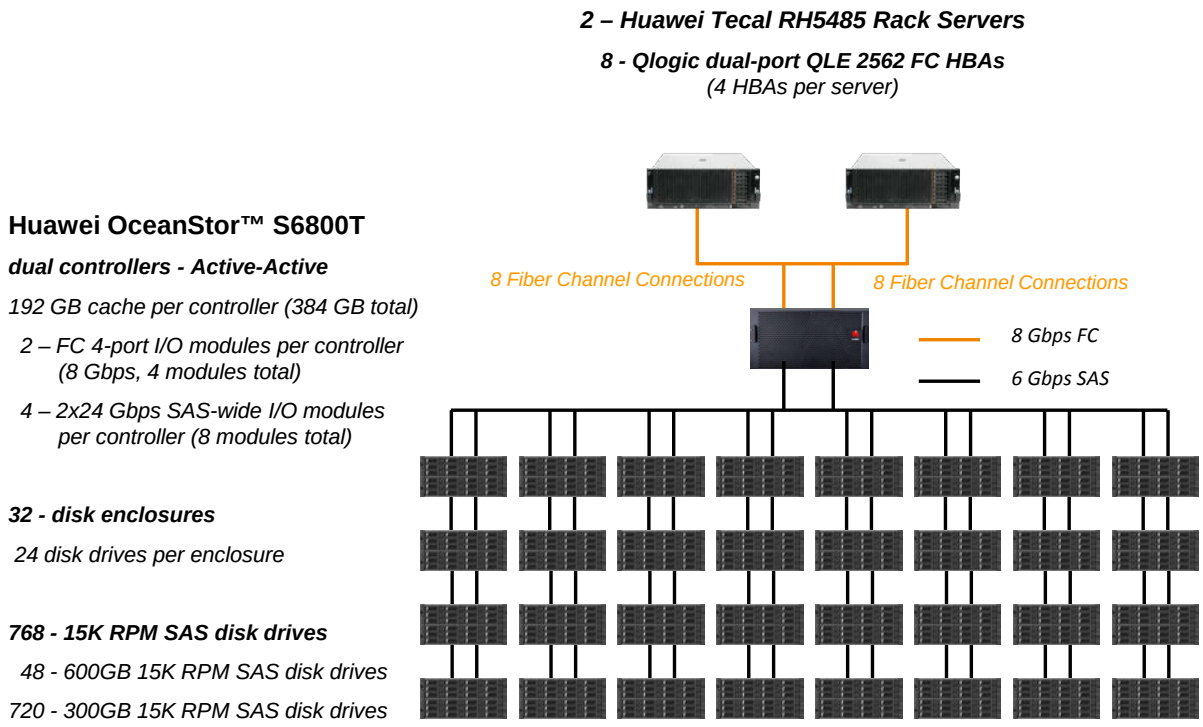
Host System and Tested Storage Configuration (TSC) Table of Components

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). Table 9-10 specifies the content, format, and appearance of the table.

The Host System and TSC table of components may be found on page 18 *(Host Systems and Tested Storage Configuration Components).*

Benchmark Configuration/Tested Storage Configuration Diagram



Host Systems and Tested Storage Configuration Components

Host Systems:	Tested Storage Configuration (TSC):
2 – Huawei Tecal RH5485 Rack Servers each server with: 2 – Intel Xeon E7530 1.87 GHz CPUs 128 GB main memory Red Hat Enterprise Linux 5.5 x86_64 LVM version 2.02.56(1) PCIe	8 – QLogic dual-ported QLE2562 FC HBAs
	Huawei OceanStor™ S6800T **dual controllers – Active-Active** 192 GB cache per controller *(384 GB total)* 2 – FC 4-port I/O modules per controller *(8 Gbps, 4 modules total)* 8 – 8Gbps front-end connections per controller *(16 total, 16 used)* 4 – 2x24 Gbps SAS-wide I/O modules per controller *(8 modules total, 2 ports per modules)* 8 – 2x24 Gbps SAS-wide backend connections per controller *(16 total and used)*
	32 –Disk Enclosures 24 – 3.5” HD slots per enclosure 24 – disk drives per enclosure
	768 – 15K RPM SAS disk drives 720 – 300 GB disk drives 48 – 600 GB disk drives

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 60 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 61 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 71.

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 56 contains definitions of terms specific to the SPC-1 Data Repository.

Storage Capacities and Relationships

Clause 9.4.3.6.1

Two tables and an illustration documenting the storage capacities and relationships of the SPC-1 Storage Hierarchy (Clause 2.1) shall be included in the FDR.

SPC-1 Storage Capacities

SPC-1 Storage Capacities		
Storage Hierarchy Component	**Units**	**Capacity**
Total ASU Capacity	Gigabytes (GB)	86,973.088
Addressable Storage Capacity	Gigabytes (GB)	86,973.088
Configured Storage Capacity	Gigabytes (GB)	244,078.696
Physical Storage Capacity	Gigabytes (GB)	244,806.109
Data Protection *(Mirroring)*	Gigabytes (GB)	122,039.348
Required Storage *(overhead)*	Gigabytes (GB)	0.000
Global Storage Overhead	Gigabytes (GB)	529.844
Total Unused Storage	Gigabytes (GB)	70,330.089

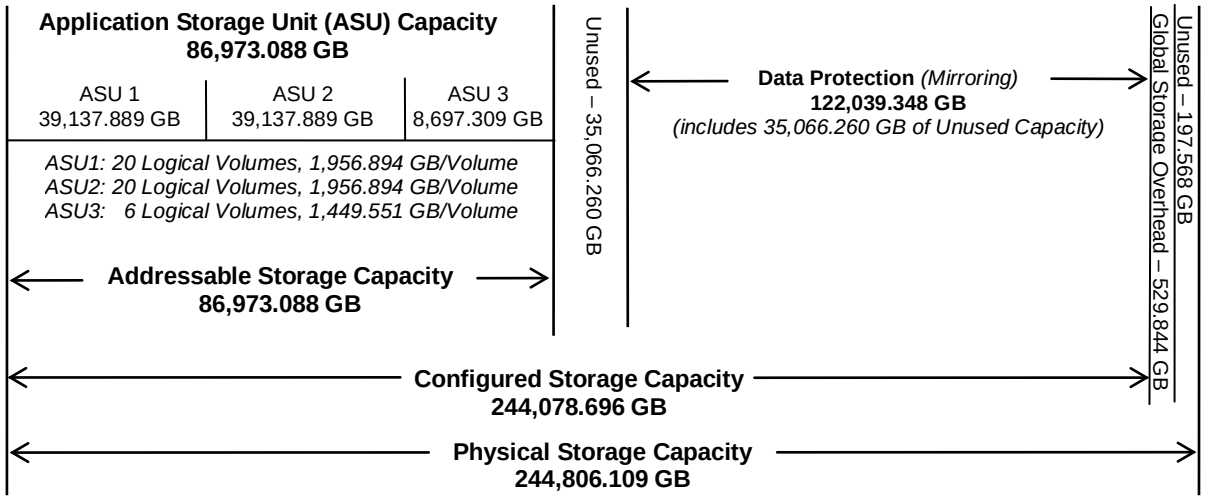
SPC-1 Storage Hierarchy Ratios

	Addressable Storage Capacity	**Configured Storage Capacity**	**Physical Storage Capacity**
Total ASU Capacity	100.00%	35.63%	35.53%
Required for Data Protection *(Mirroring)*		50.00%	49.85%
Addressable Storage Capacity		35.63%	35.53%
Required Storage		0.00%	0.00%
Configured Storage Capacity			99.70%
Global Storage Overhead			0.22%
Unused Storage:			
Addressable	0.00%		
Configured		28.73%	
Physical			0.08%

The Physical Storage Capacity consisted of 244,806.109 GB distributed over 720 disk drives, each with a formatted capacity of 300.000 GB and 48 disk drives, each with a formatted capacity of 600.127 GB. There was 197.568 GB (0.08%) of Unused Storage within the Physical Storage Capacity. Global Storage Overhead consisted of 529.844 GB (0.22%) of the Physical Storage Capacity. There was 70,132.521 GB (28.73%) 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 122,039.348 GB of which 86,973.088 GB was utilized. The total Unused Storage capacity was 70,330.089 GB.

SPC-1 Storage Capacities and Relationships Illustration

The various storage capacities configured in the benchmark result are illustrated below *(not to scale)*.



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 (39,137.889 GB)	**ASU-2 (39,137.889 GB)**	**ASU-3 (8,697.309 GB)**
20 Logical Volumes 1,956.894 GB per Logical Volume (1,956.894 GB used per Logical Volume)	20 Logical Volumes 1,956.894 GB per Logical Volume (1,956.894 GB used per Logical Volume)	6 Logical Volumes 1,449.551 GB per Logical Volume (1,449.551 GB used per Logical Volume)

The Data Protection Level used for all Logical Volumes was ***Mirrored*** as described on page 11. See "ASU Configuration" in the **IOPS Test Results File** for more detailed configuration information.

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	35.53%
Protected Application Utilization	71.05%
Unused Storage Ratio	28.73%

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. "SPC-1 Test Execution Definitions" on page 57 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.

Primary Metrics Test – Sustainability Test Phase

Clause 5.4.4.1.1

The Sustainability Test Phase has exactly one Test Run and shall demonstrate the maximum sustainable I/O Request Throughput within at least a continuous three (3) 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™).

Clause 5.4.4.1.2

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

Clause 5.4.4.1.4

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.

Clause 9.4.3.7.1

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 76.

Sustainability Test Results File

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

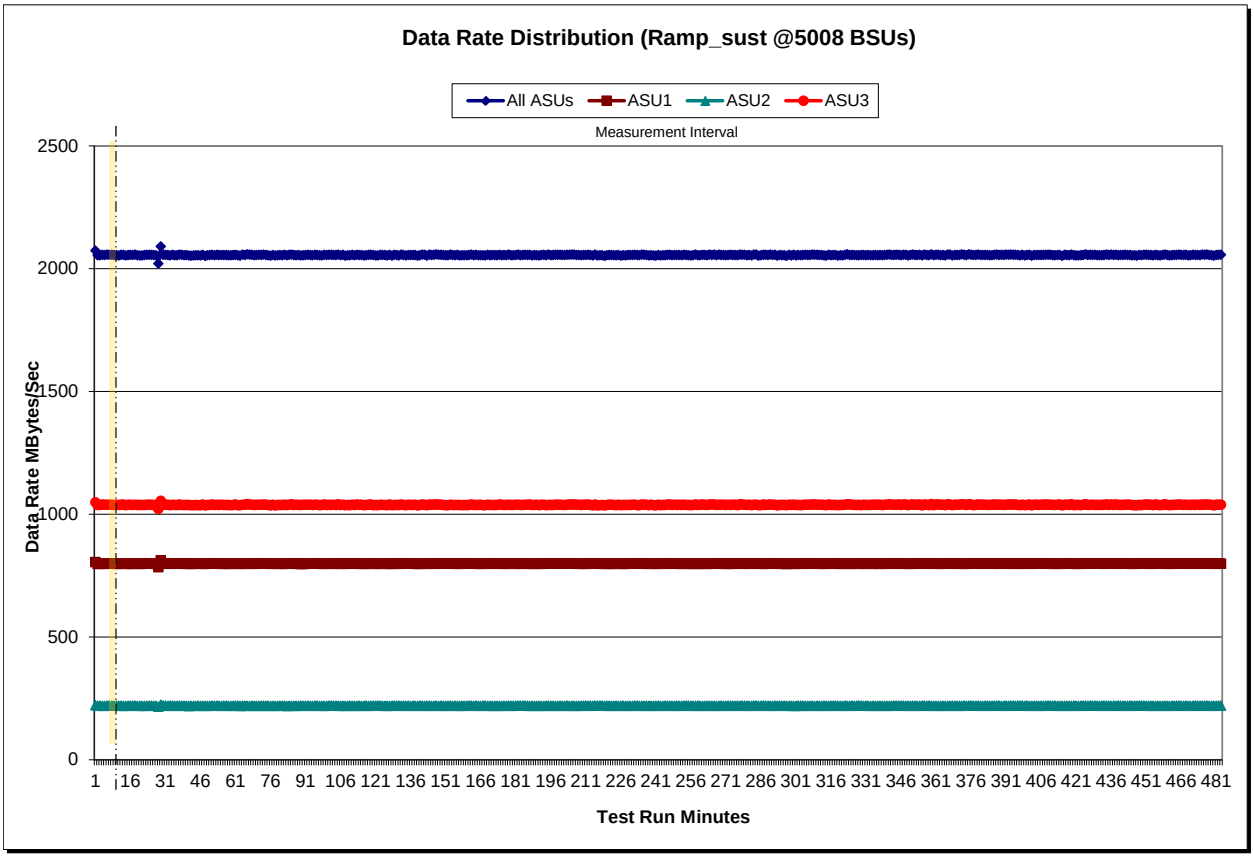
Sustainability Test Results File

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

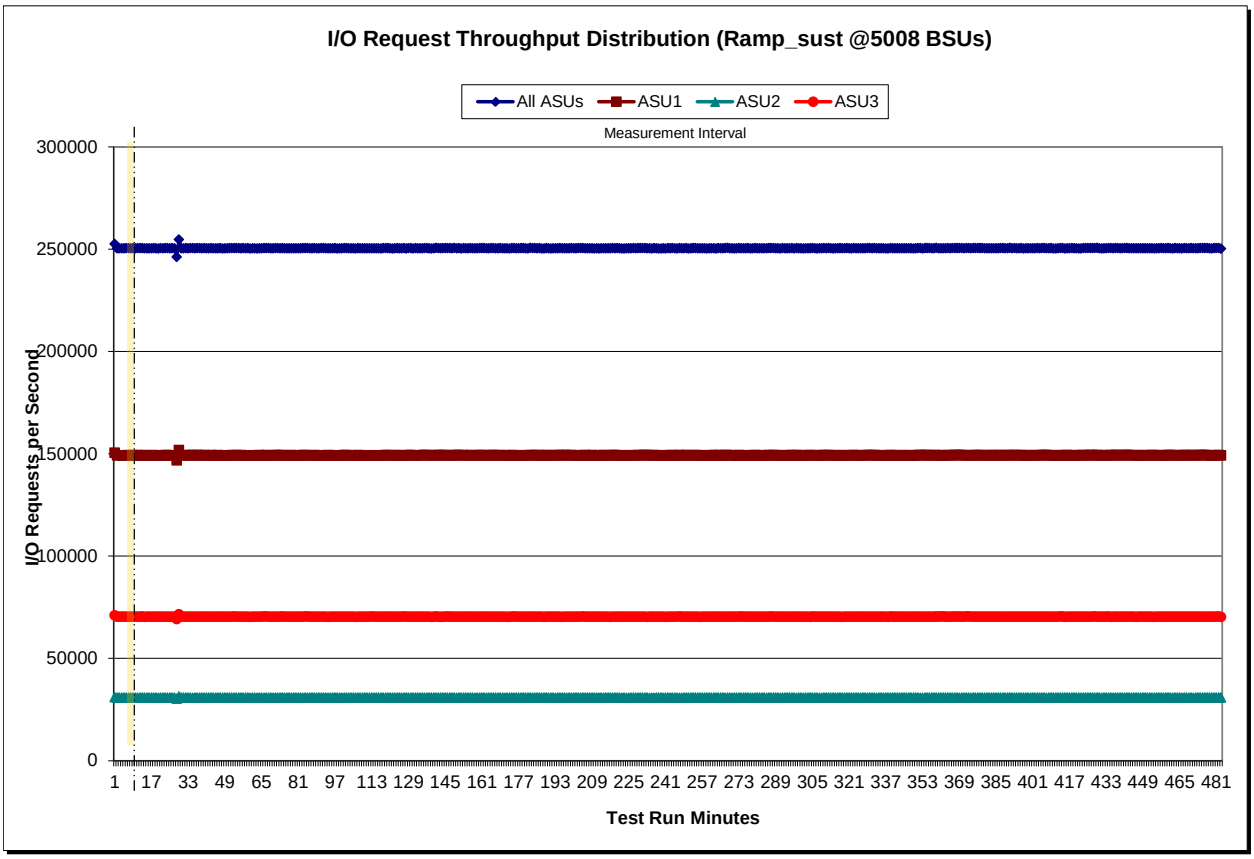


Sustainability – I/O Request Throughput Distribution Data

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

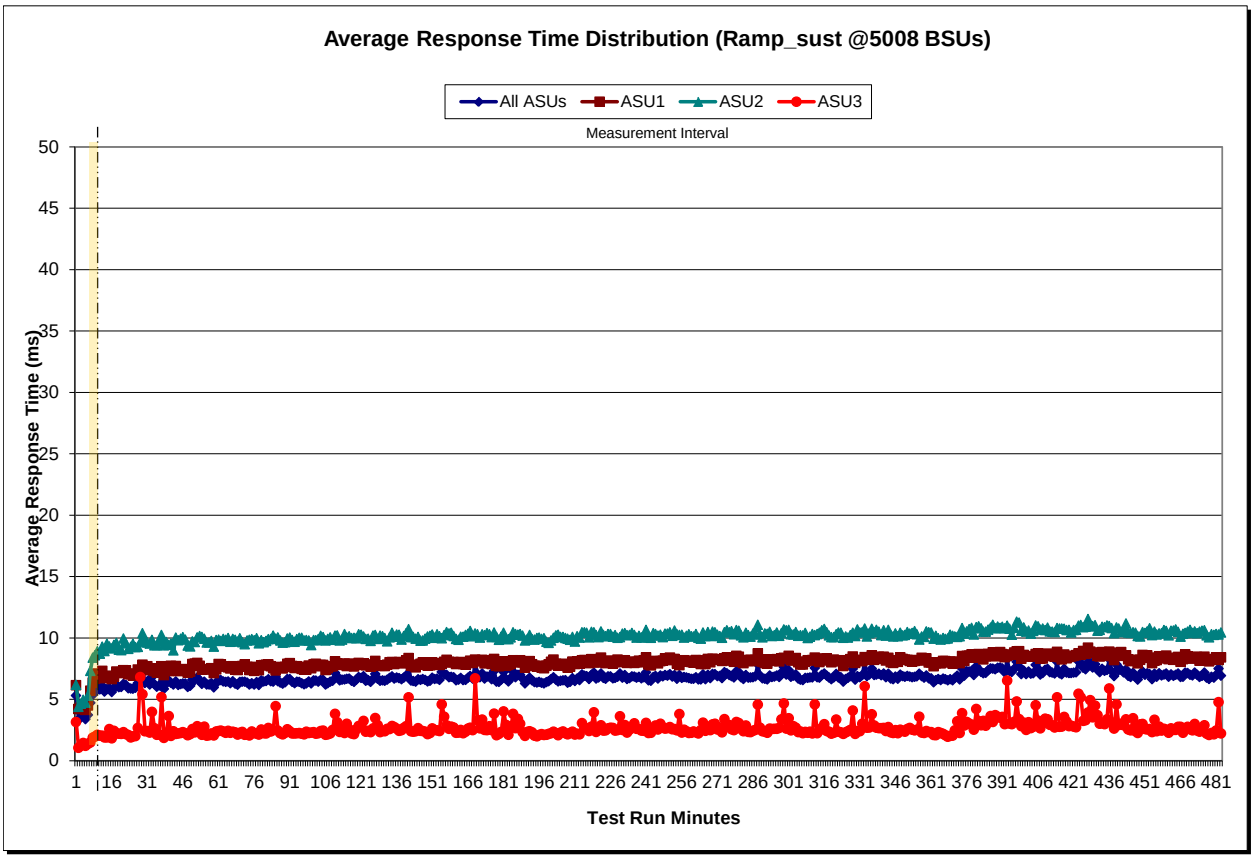


Sustainability – Average Response Time (ms) Distribution Data

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

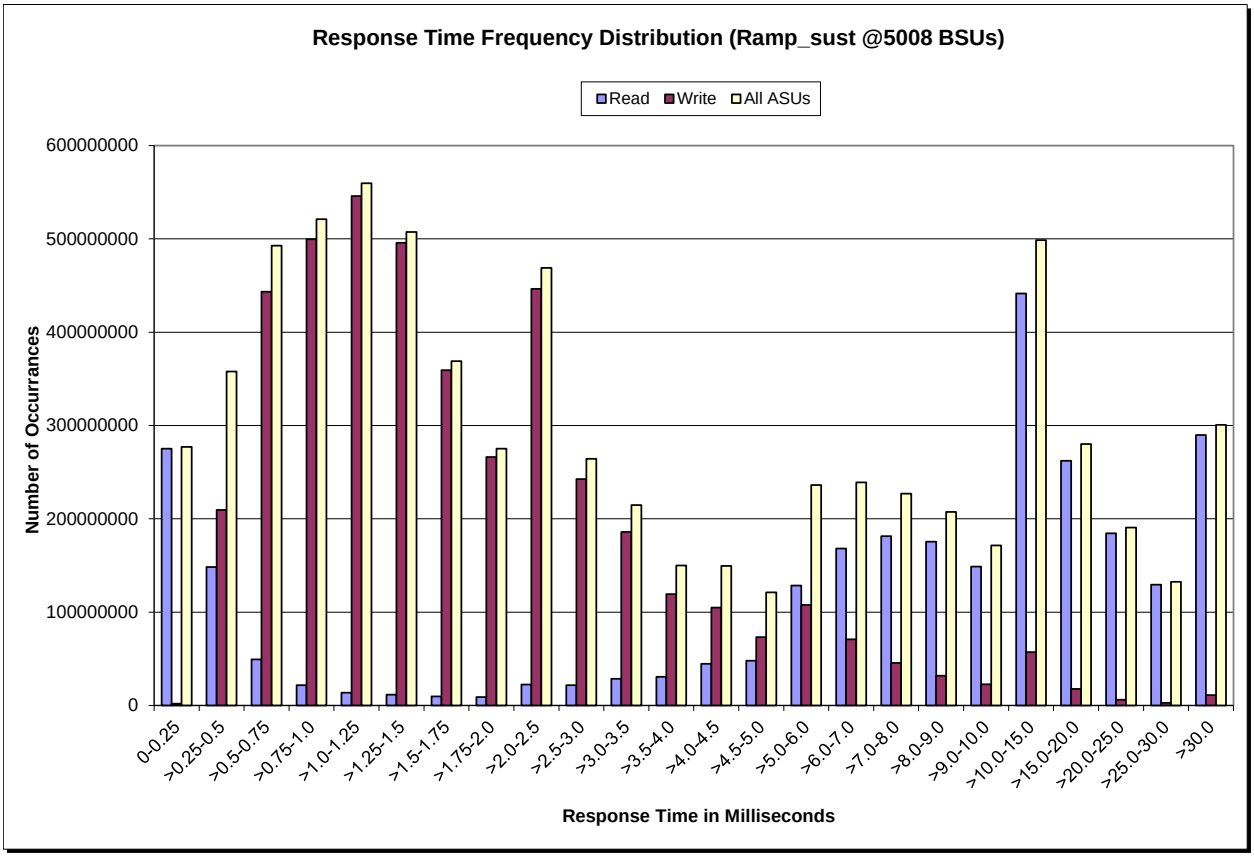
Sustainability – Average Response Time (ms) Distribution Graph



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	275,230,240	148,258,443	49,397,847	21,639,020	13,721,768	11,545,276	9,621,981	8,942,230
Write	1,801,407	209,545,092	443,235,319	499,504,436	545,754,192	495,859,432	359,261,615	266,300,804
All ASUs	277,031,647	357,803,535	492,633,166	521,143,456	559,475,960	507,404,708	368,883,596	275,243,034
ASU1	255,711,194	233,713,546	242,496,884	238,251,272	249,485,717	224,637,514	163,050,633	121,898,783
ASU2	20,607,906	33,186,482	49,461,783	53,027,009	57,400,272	52,217,170	38,014,267	28,470,397
ASU3	712,547	90,903,507	200,674,499	229,865,175	252,589,971	230,550,024	167,818,696	124,873,854
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	22,447,784	21,620,629	28,698,628	30,672,685	44,633,675	47,998,447	128,543,184	168,170,652
Write	446,426,451	242,661,800	185,924,681	119,234,549	104,800,045	73,115,530	107,674,861	70,788,323
All ASUs	468,874,235	264,282,429	214,623,309	149,907,234	149,433,720	121,113,977	236,218,045	238,958,975
ASU1	210,713,819	122,737,121	105,027,097	78,621,359	85,281,428	74,784,455	161,141,310	177,398,362
ASU2	48,940,701	27,740,741	22,584,134	15,535,354	15,215,297	12,152,912	24,667,755	28,365,046
ASU3	209,219,715	113,804,567	87,012,078	55,750,521	48,936,995	34,176,610	50,408,980	33,195,567
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	181,401,178	175,441,210	148,793,219	441,491,598	262,217,556	184,391,103	129,540,905	289,790,556
Write	45,561,507	31,978,712	22,724,990	57,149,782	17,804,156	6,229,531	2,889,171	10,994,793
All ASUs	226,962,685	207,419,922	171,518,209	498,641,380	280,021,712	190,620,634	132,430,076	300,785,349
ASU1	174,375,601	159,294,814	130,986,825	386,020,257	225,106,155	154,222,046	106,097,756	217,020,429
ASU2	31,241,386	33,156,965	29,895,201	85,683,138	46,159,348	32,963,206	24,517,200	75,804,323
ASU3	21,345,698	14,968,143	10,636,183	26,937,985	8,756,209	3,435,382	1,815,120	7,960,597

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 5.3.13.2

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 5.3.13.3

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	0.001	0.002	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 9.4.3.7.2

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 76.

IOPS Test Results File

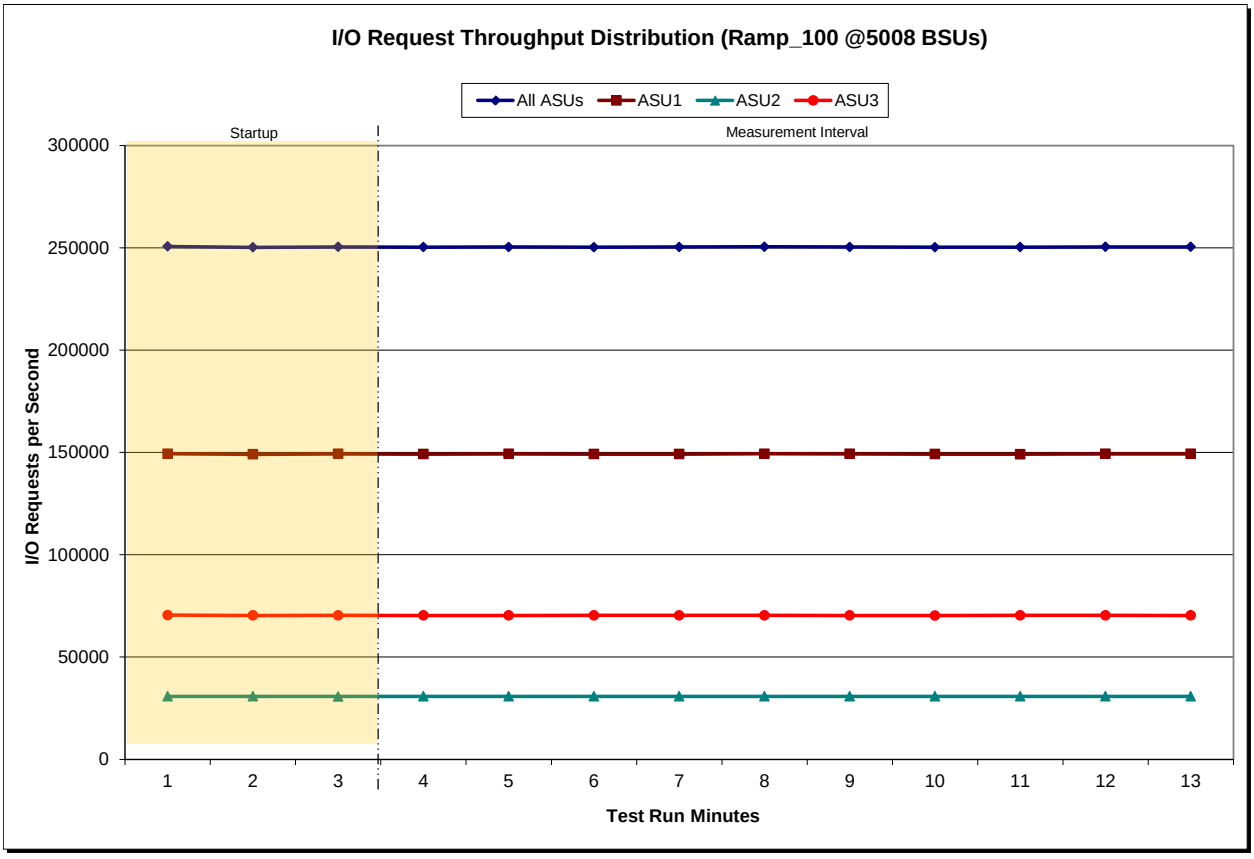
A link to the test results file generated from the IOPS Test Run is listed below.

IOPS Test Results File

IOPS Test Run – I/O Request Throughput Distribution Data

5,008 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	0:39:36	0:42:37	0-2	0:03:01
Measurement Interval	0:42:37	0:52:37	3-12	0:10:00
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	250,660.25	149,365.12	30,804.88	70,490.25
1	250,239.63	149,103.20	30,810.43	70,326.00
2	250,439.63	149,268.13	30,767.70	70,403.80
3	250,346.55	149,202.05	30,800.00	70,344.50
4	250,412.90	149,269.57	30,788.25	70,355.08
5	250,313.35	149,173.88	30,778.00	70,361.47
6	250,384.68	149,187.90	30,817.27	70,379.52
7	250,502.90	149,323.45	30,785.82	70,393.63
8	250,416.87	149,289.57	30,784.95	70,342.35
9	250,277.25	149,163.57	30,811.33	70,302.35
10	250,354.60	149,159.37	30,799.72	70,395.52
11	250,438.23	149,279.45	30,781.05	70,377.73
12	250,441.83	149,294.05	30,788.57	70,359.22
Average	***250,388.92***	***149,234.29***	***30,793.50***	***70,361.14***

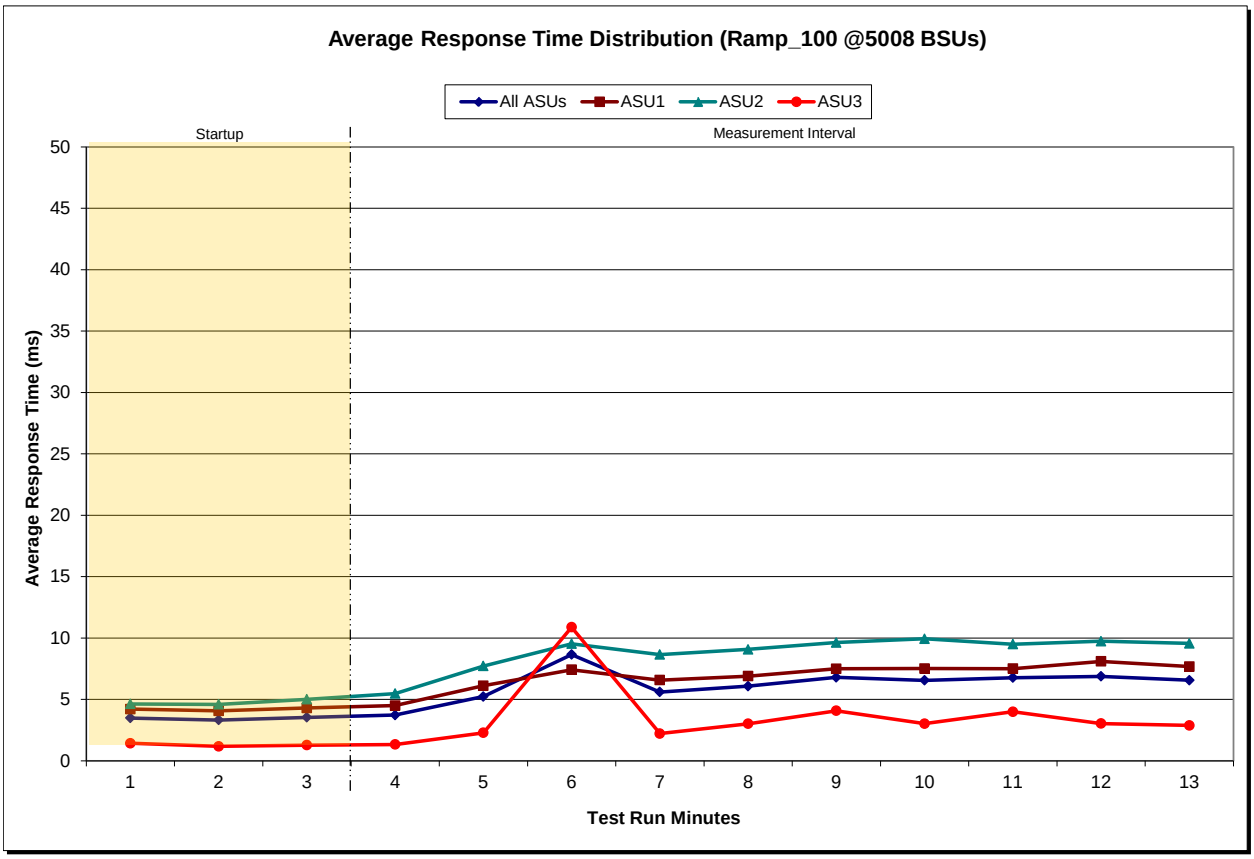
IOPS Test Run – I/O Request Throughput Distribution Graph



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

5,008 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	0:39:36	0:42:37	0-2	0:03:01
Measurement Interval	0:42:37	0:52:37	3-12	0:10:00
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	3.48	4.21	4.63	1.43
1	3.32	4.07	4.59	1.17
2	3.53	4.29	5.00	1.27
3	3.73	4.50	5.48	1.33
4	5.23	6.11	7.72	2.28
5	8.65	7.41	9.54	10.88
6	5.60	6.57	8.65	2.22
7	6.07	6.90	9.07	3.01
8	6.80	7.50	9.63	4.07
9	6.55	7.51	9.93	3.02
10	6.76	7.51	9.49	3.99
11	6.87	8.09	9.73	3.04
12	6.56	7.67	9.56	2.88
Average	***6.28***	***6.98***	***8.88***	***3.67***

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



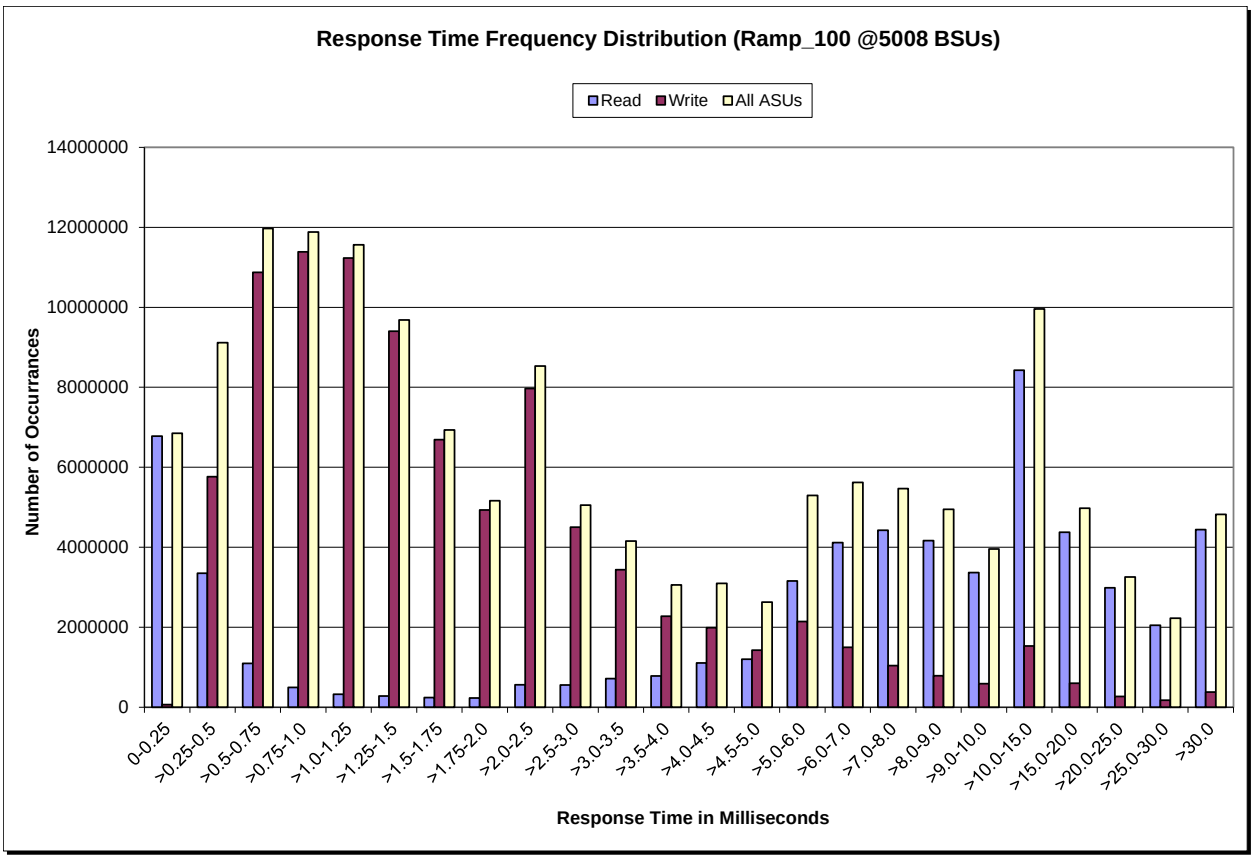
IOPS Test Run – Response Time Frequency Distribution Data

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	6,779,620	3,349,763	1,098,998	493,652	326,995	281,530	244,467	232,832
Write	68,520	5,765,564	10,872,890	11,386,414	11,234,394	9,404,380	6,688,578	4,931,445
All ASUs	6,848,140	9,115,327	11,971,888	11,880,066	11,561,389	9,685,910	6,933,045	5,164,277
ASU1	6,393,712	5,785,177	5,842,236	5,418,852	5,156,278	4,300,289	3,081,682	2,311,714
ASU2	427,100	817,465	1,191,470	1,200,156	1,183,947	997,563	716,828	537,291
ASU3	27,328	2,512,685	4,938,182	5,261,058	5,221,164	4,388,058	3,134,535	2,315,272

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	564,650	556,087	717,449	785,474	1,107,635	1,200,327	3,156,512	4,118,517
Write	7,967,857	4,500,342	3,439,523	2,275,351	1,991,670	1,428,614	2,141,462	1,501,611
All ASUs	8,532,507	5,056,429	4,156,972	3,060,825	3,099,305	2,628,941	5,297,974	5,620,128
ASU1	3,896,333	2,406,627	2,102,750	1,674,933	1,848,119	1,691,488	3,720,124	4,207,003
ASU2	899,220	538,316	445,274	323,529	322,926	270,133	579,199	713,973
ASU3	3,736,954	2,111,486	1,608,948	1,062,363	928,260	667,320	998,651	699,152

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	4,427,139	4,164,241	3,367,653	8,424,097	4,373,524	2,988,485	2,052,777	4,443,572
Write	1,041,896	786,213	589,523	1,532,624	602,957	269,070	174,430	380,201
All ASUs	5,469,035	4,950,454	3,957,176	9,956,721	4,976,481	3,257,555	2,227,207	4,823,773
ASU1	4,171,909	3,727,125	2,931,688	7,401,157	3,841,182	2,544,963	1,719,144	3,364,929
ASU2	813,855	861,647	755,671	1,854,969	854,118	581,222	418,834	1,171,082
ASU3	483,271	361,682	269,817	700,595	281,181	131,370	89,229	287,762

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
150,231,525	145,407,752	4,823,773

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 5.3.13.2

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 5.3.13.3

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	0.2800
COV	0.002	0.000	0.001	0.001	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 13.

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 9.4.3.7.3

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 76.

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 tables and graphs for completeness.

100% Load Level - 5,008 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	0:39:36	0:42:37	0-2	0:03:01
Measurement Interval	0:42:37	0:52:37	3-12	0:10:00

(60 second intervals)	All ASUs	ASU-1	ASU-2	ASU-3
0	250,660.25	149,365.12	30,804.88	70,490.25
1	250,239.63	149,103.20	30,810.43	70,326.00
2	250,439.63	149,268.13	30,767.70	70,403.80
3	250,346.55	149,202.05	30,800.00	70,344.50
4	250,412.90	149,269.57	30,788.25	70,355.08
5	250,313.35	149,173.88	30,778.00	70,361.47
6	250,384.68	149,187.90	30,817.27	70,379.52
7	250,502.90	149,323.45	30,785.82	70,393.63
8	250,416.87	149,289.57	30,784.95	70,342.35
9	250,277.25	149,163.57	30,811.33	70,302.35
10	250,354.60	149,159.37	30,799.72	70,395.52
11	250,438.23	149,279.45	30,781.05	70,377.73
12	250,441.83	149,294.05	30,788.57	70,359.22
Average	***250,388.92***	***149,234.29***	***30,793.50***	***70,361.14***

95% Load Level - 4,757 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	1:04:18	1:07:19	0-2	0:03:01
Measurement Interval	1:07:19	1:17:19	3-12	0:10:00

(60 second intervals)	All ASUs	ASU-1	ASU-2	ASU-3
0	238,081.63	141,941.20	29,283.93	66,856.50
1	237,769.83	141,718.23	29,228.20	66,823.40
2	237,902.08	141,808.12	29,275.18	66,818.78
3	237,863.00	141,792.73	29,281.37	66,788.90
4	237,939.42	141,827.45	29,259.45	66,852.52
5	237,756.83	141,669.25	29,257.08	66,830.50
6	237,882.92	141,819.98	29,254.82	66,808.12
7	237,946.32	141,851.68	29,221.27	66,873.37
8	237,765.67	141,697.65	29,249.52	66,818.50
9	237,762.17	141,712.07	29,227.45	66,822.65
10	237,863.15	141,804.15	29,277.03	66,781.97
11	237,827.33	141,773.72	29,302.33	66,751.28
12	237,888.35	141,758.47	29,287.72	66,842.17
Average	***237,849.52***	***141,770.72***	***29,261.80***	***66,817.00***

90% Load Level - 4,507 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	1:34:23	1:37:24	0-2	0:03:01
Measurement Interval	1:37:24	1:47:25	3-12	0:10:01

(60 second intervals)	All ASUs	ASU-1	ASU-2	ASU-3
0	225,563.80	134,455.08	27,744.00	63,364.72
1	225,448.35	134,360.38	27,734.50	63,353.47
2	225,422.38	134,325.97	27,782.60	63,313.82
3	225,356.83	134,351.28	27,714.28	63,291.27
4	225,361.70	134,337.00	27,723.92	63,300.78
5	225,393.32	134,387.62	27,700.17	63,305.53
6	225,439.05	134,336.83	27,722.92	63,379.30
7	225,339.63	134,334.42	27,688.88	63,316.33
8	225,330.62	134,318.78	27,706.02	63,305.82
9	225,385.28	134,310.22	27,724.28	63,350.78
10	225,289.48	134,304.25	27,706.13	63,279.10
11	225,363.93	134,349.98	27,748.02	63,265.93
12	225,320.32	134,279.23	27,690.83	63,350.25
Average	***225,358.02***	***134,330.96***	***27,712.55***	***63,314.51***

80% Load Level - 4,006 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	2:11:30	2:14:31	0-2	0:03:01
Measurement Interval	2:14:31	2:24:31	3-12	0:10:00

(60 second intervals)	All ASUs	ASU-1	ASU-2	ASU-3
0	200,637.78	119,583.02	24,679.53	56,375.23
1	200,232.95	119,342.40	24,652.37	56,238.18
2	200,447.77	119,470.60	24,686.02	56,291.15
3	200,285.60	119,326.52	24,658.00	56,301.08
4	200,311.95	119,354.13	24,641.88	56,315.93
5	200,251.63	119,362.53	24,650.27	56,238.83
6	200,352.90	119,400.73	24,643.75	56,308.42
7	200,317.25	119,410.25	24,609.97	56,297.03
8	200,292.77	119,404.92	24,626.33	56,261.52
9	200,305.83	119,388.58	24,618.98	56,298.27
10	200,304.55	119,356.28	24,631.98	56,316.28
11	200,251.32	119,355.45	24,648.67	56,247.20
12	200,309.38	119,371.05	24,619.88	56,318.45
Average	***200,298.32***	***119,373.05***	***24,634.97***	***56,290.30***

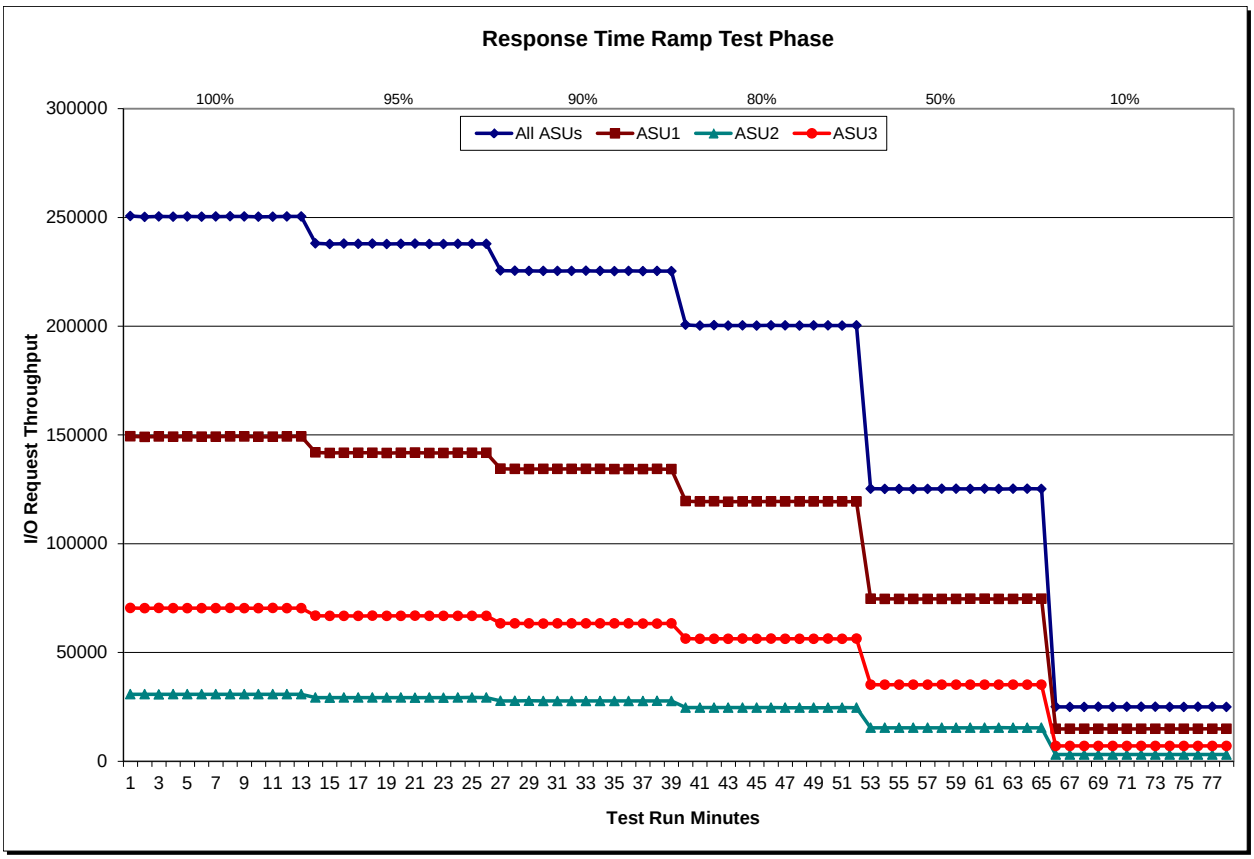
50% Load Level - 2,504 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	2:42:58	2:42:58	0-2	0:00:00
Measurement Interval	2:42:58	2:42:58	3-12	0:00:00

(60 second intervals)	All ASUs	ASU-1	ASU-2	ASU-3
0	125,290.37	74,685.08	15,399.83	35,205.45
1	125,165.87	74,616.20	15,388.75	35,160.92
2	125,202.22	74,622.00	15,402.48	35,177.73
3	125,110.63	74,594.93	15,386.98	35,128.72
4	125,208.02	74,616.05	15,404.78	35,187.18
5	125,225.95	74,641.65	15,396.40	35,187.90
6	125,242.17	74,641.15	15,413.95	35,187.07
7	125,190.50	74,648.85	15,368.37	35,173.28
8	125,270.62	74,682.08	15,400.88	35,187.65
9	125,152.67	74,590.78	15,421.40	35,140.48
10	125,217.93	74,640.40	15,394.47	35,183.07
11	125,282.08	74,704.60	15,383.78	35,193.70
12	125,204.80	74,649.57	15,388.08	35,167.15
Average	***125,210.54***	***74,641.01***	***15,395.91***	***35,173.62***

10% Load Level - 500 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	3:08:08	3:11:09	0-2	0:03:01
Measurement Interval	3:11:09	3:21:09	3-12	0:10:00

(60 second intervals)	All ASUs	ASU-1	ASU-2	ASU-3
0	25,001.87	14,899.07	3,076.82	7,025.98
1	24,980.35	14,901.97	3,065.80	7,012.58
2	24,969.58	14,870.55	3,069.95	7,029.08
3	25,009.33	14,916.15	3,072.27	7,020.92
4	24,986.67	14,888.57	3,072.83	7,025.27
5	25,015.23	14,906.50	3,073.22	7,035.52
6	25,025.43	14,907.02	3,079.13	7,039.28
7	25,040.30	14,911.28	3,077.95	7,051.07
8	24,990.77	14,874.27	3,088.42	7,028.08
9	24,958.10	14,891.92	3,073.90	6,992.28
10	25,013.92	14,898.10	3,082.03	7,033.78
11	24,998.42	14,917.80	3,070.15	7,010.47
12	24,992.78	14,899.98	3,074.07	7,018.73
Average	***25,003.10***	***14,901.16***	***3,076.40***	***7,025.54***

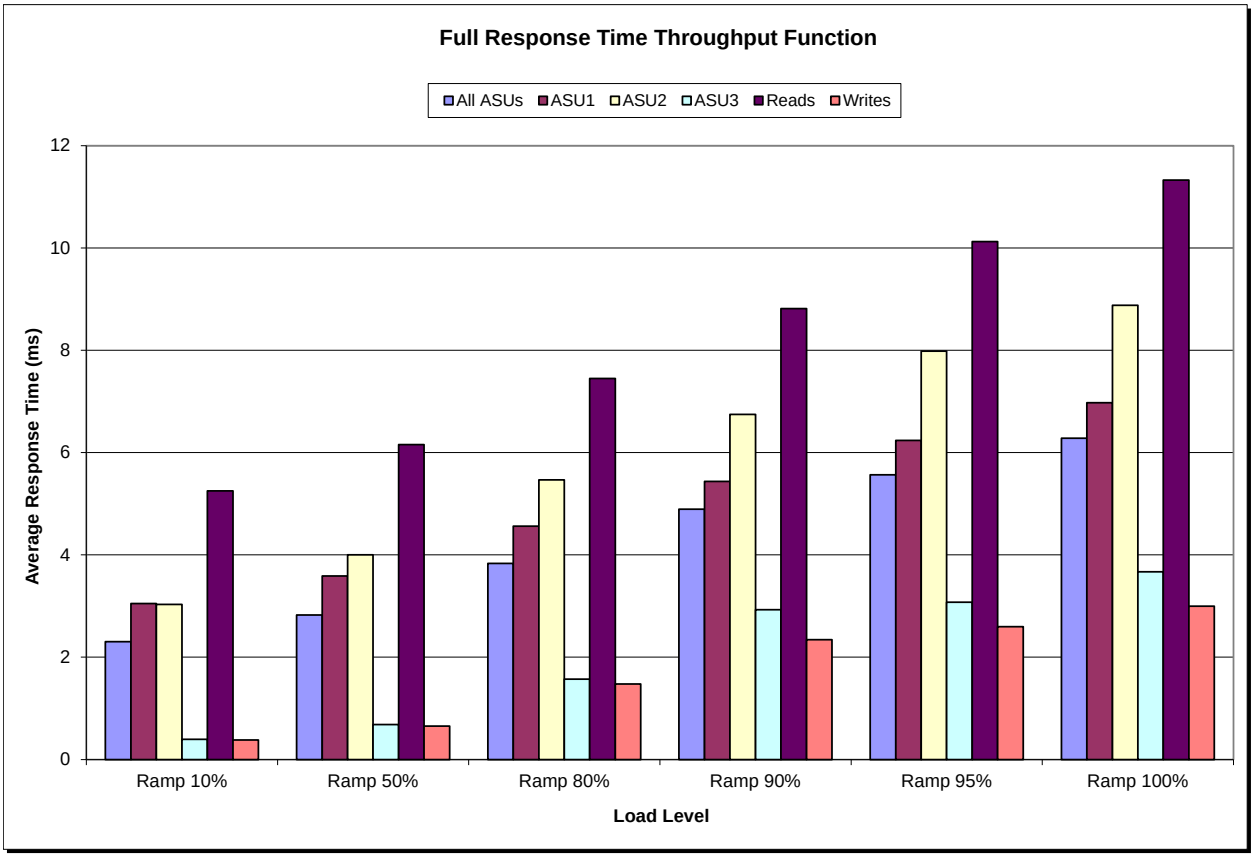
Response Time Ramp Distribution (IOPS) Graph



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

500 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	3:08:08	3:11:09	0-2	0:03:01
Measurement Interval	3:11:09	3:21:09	3-12	0:10:00
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	2.31	3.08	3.04	0.35
1	2.27	3.03	2.97	0.35
2	2.26	3.01	2.98	0.35
3	2.22	2.95	2.95	0.35
4	2.30	3.03	3.04	0.41
5	2.44	3.18	3.17	0.55
6	2.26	3.00	3.00	0.38
7	2.27	3.01	3.00	0.39
8	2.31	3.07	3.03	0.39
9	2.31	3.06	3.03	0.38
10	2.31	3.07	3.04	0.37
11	2.30	3.06	3.04	0.37
12	2.31	3.07	3.03	0.37
Average	***2.30***	***3.05***	***3.03***	***0.40***

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 5.3.13.2

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 5.3.13.3

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	0.0349	0.2810	0.0701	0.2099	0.0180	0.0700	0.0350	0.2810
COV	0.004	0.002	0.003	0.001	0.005	0.002	0.005	0.002

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 9.4.3.7.4

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 76.

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	***250,388.92***
Repeatability Test Phase 1	250,367.22
Repeatability Test Phase 2	250,396.18

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	***2.30 ms***
Repeatability Test Phase 1	2.37 ms
Repeatability Test Phase 2	2.25 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 minus one (1) millisecond (ms)...

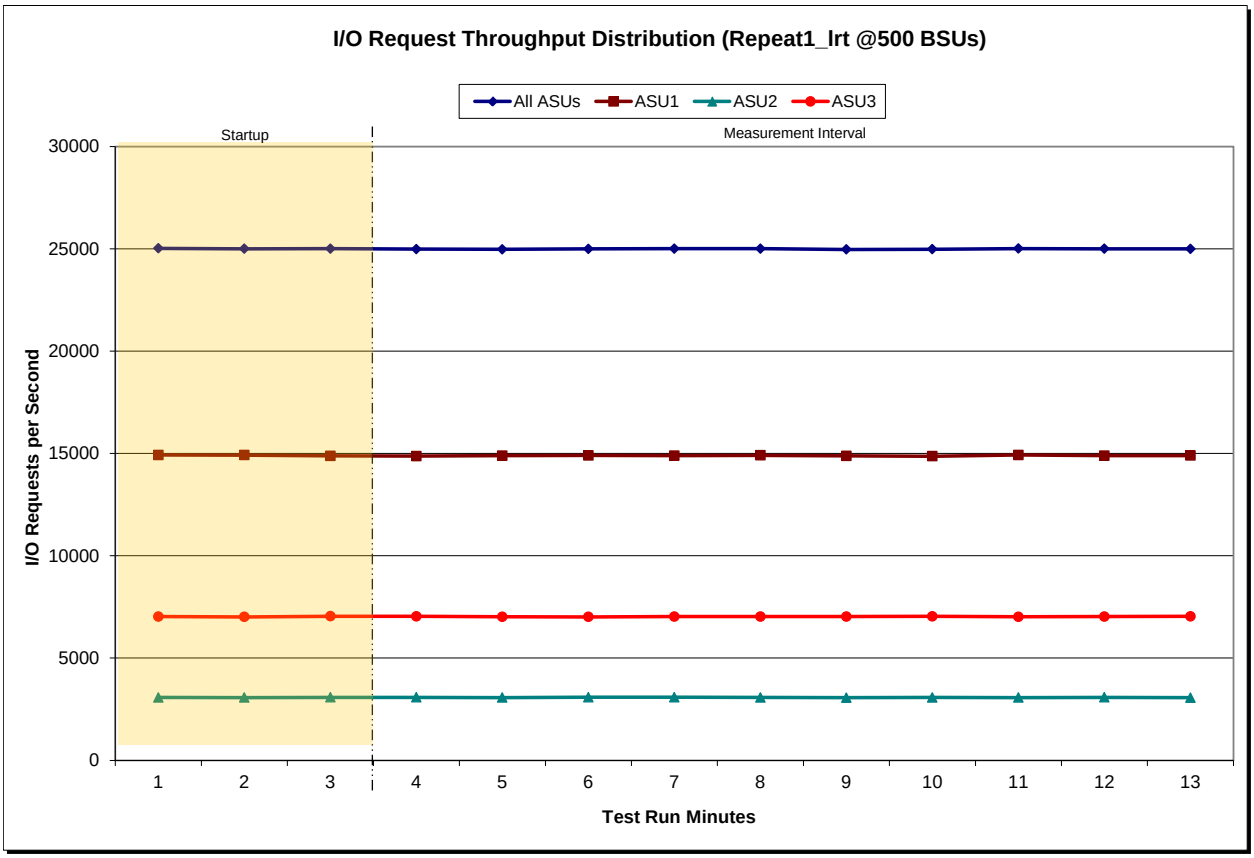
A link to the test result file generated from each Repeatability Test Run is listed below.

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

500 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	3:32:34	3:35:34	0-2	0:03:00
Measurement Interval	3:35:34	3:45:34	3-12	0:10:00
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	25,024.53	14,922.72	3,076.53	7,025.28
1	25,000.25	14,921.25	3,069.02	7,009.98
2	25,005.52	14,880.75	3,079.23	7,045.53
3	24,988.13	14,868.80	3,078.12	7,041.22
4	24,976.93	14,891.38	3,067.83	7,017.72
5	24,996.22	14,901.17	3,085.43	7,009.62
6	25,005.00	14,893.63	3,084.65	7,026.72
7	25,004.87	14,905.67	3,073.97	7,025.23
8	24,971.92	14,880.83	3,063.97	7,027.12
9	24,979.28	14,867.38	3,072.95	7,038.95
10	25,010.98	14,924.43	3,067.37	7,019.18
11	25,002.30	14,893.62	3,081.40	7,027.28
12	24,997.42	14,894.78	3,064.97	7,037.67
Average	***24,993.31***	***14,892.17***	***3,074.07***	***7,027.07***

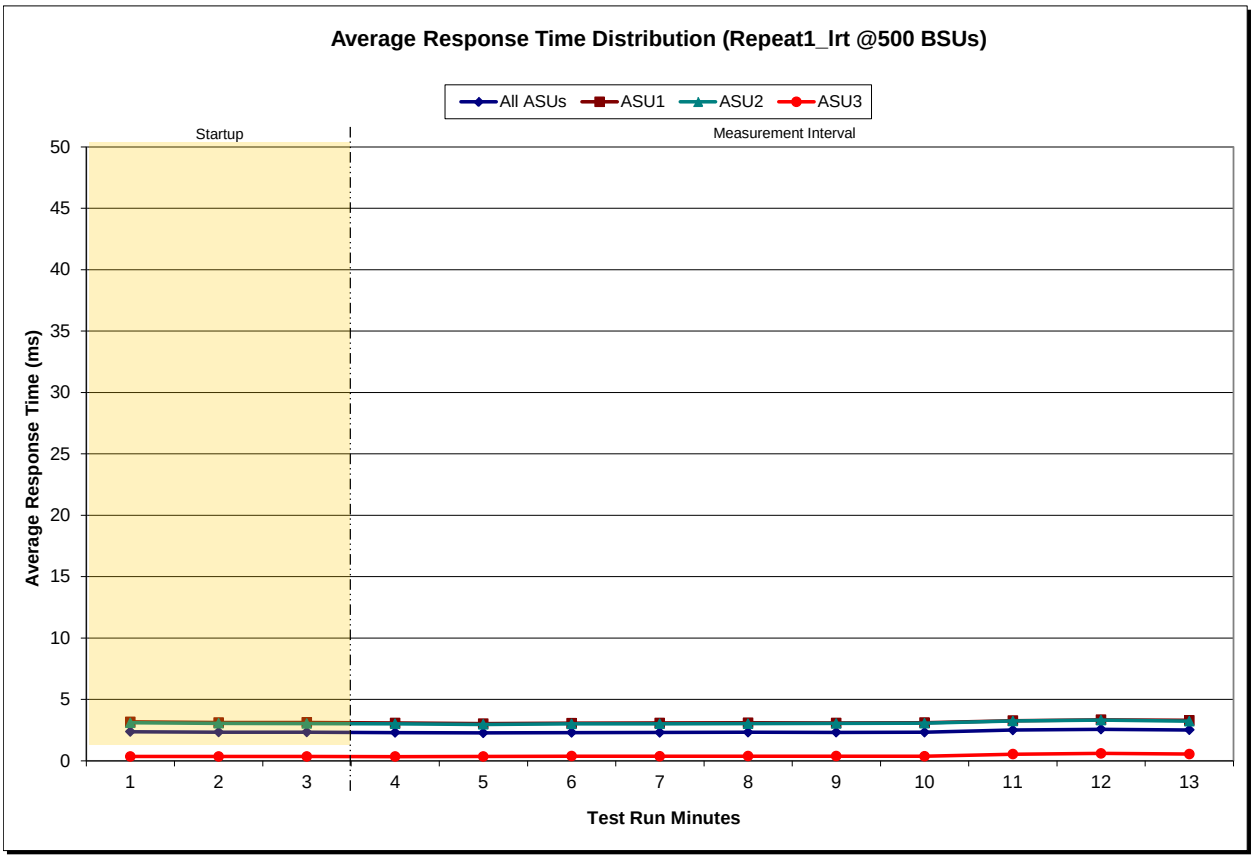
Repeatability 1 LRT – I/O Request Throughput Distribution Graph



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

500 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	3:32:34	3:35:34	0-2	0:03:00
Measurement Interval	3:35:34	3:45:34	3-12	0:10:00
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	2.37	3.17	3.09	0.35
1	2.33	3.12	3.05	0.35
2	2.33	3.12	3.04	0.35
3	2.30	3.08	3.01	0.34
4	2.27	3.03	2.96	0.35
5	2.30	3.06	3.00	0.37
6	2.31	3.08	3.01	0.37
7	2.32	3.09	3.02	0.38
8	2.31	3.08	3.04	0.38
9	2.33	3.10	3.06	0.37
10	2.50	3.27	3.24	0.53
11	2.57	3.34	3.31	0.60
12	2.51	3.29	3.22	0.54
Average	***2.37***	***3.14***	***3.09***	***0.42***

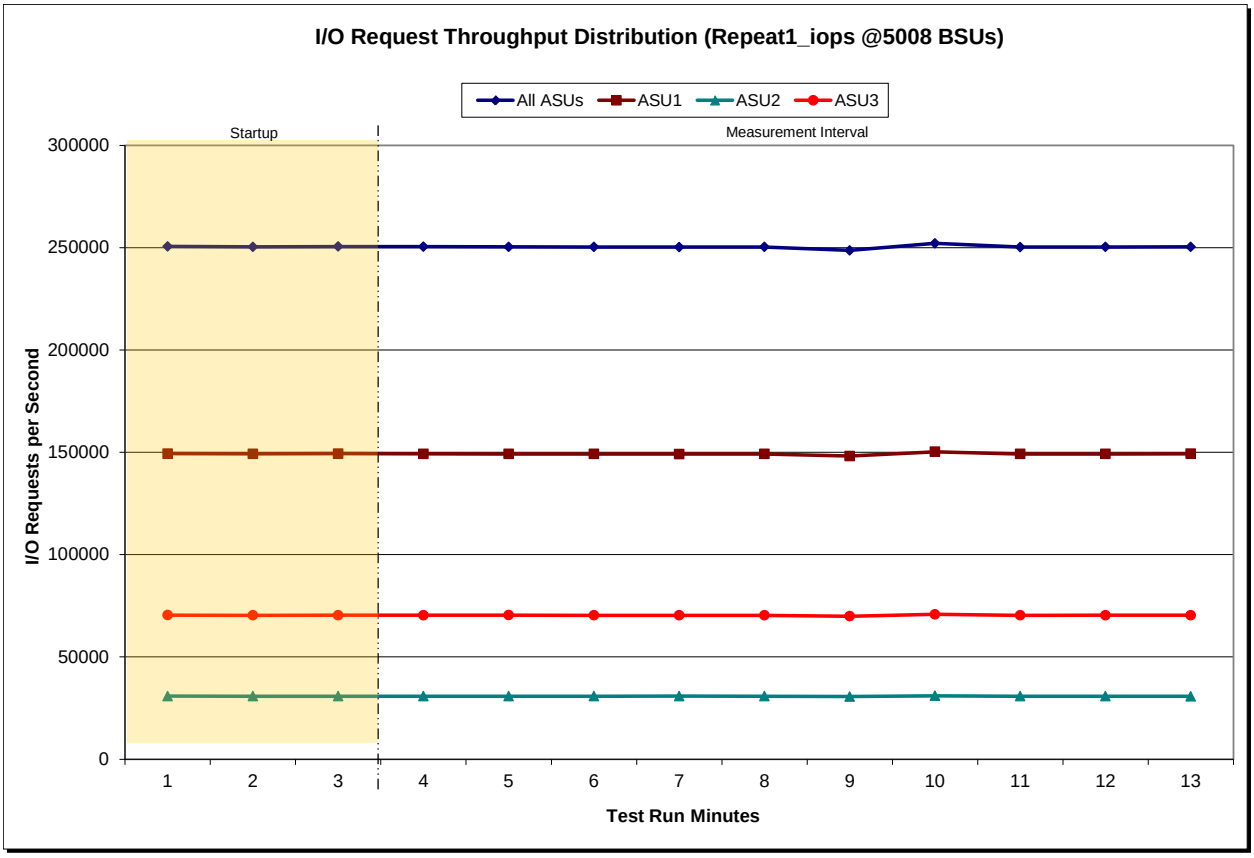
Repeatability 1 LRT –Average Response Time (ms) Distribution Graph



Repeatability 1 IOPS – I/O Request Throughput Distribution Data

5,008 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	3:57:18	4:00:19	0-2	0:03:01
Measurement Interval	4:00:19	4:10:19	3-12	0:10:00
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	250,591.10	149,341.50	30,829.42	70,420.18
1	250,372.68	149,248.58	30,797.57	70,326.53
2	250,533.18	149,347.72	30,817.73	70,367.73
3	250,480.30	149,265.40	30,814.52	70,400.38
4	250,380.35	149,189.62	30,774.57	70,416.17
5	250,345.37	149,188.30	30,804.30	70,352.77
6	250,312.48	149,153.70	30,845.40	70,313.38
7	250,334.42	149,182.93	30,804.57	70,346.92
8	248,694.05	148,176.65	30,611.40	69,906.00
9	252,113.12	150,272.17	31,016.05	70,824.90
10	250,269.12	149,164.32	30,780.72	70,324.08
11	250,337.97	149,179.50	30,777.28	70,381.18
12	250,405.02	149,275.65	30,767.03	70,362.33
Average	***250,367.22***	***149,204.82***	***30,799.58***	***70,362.81***

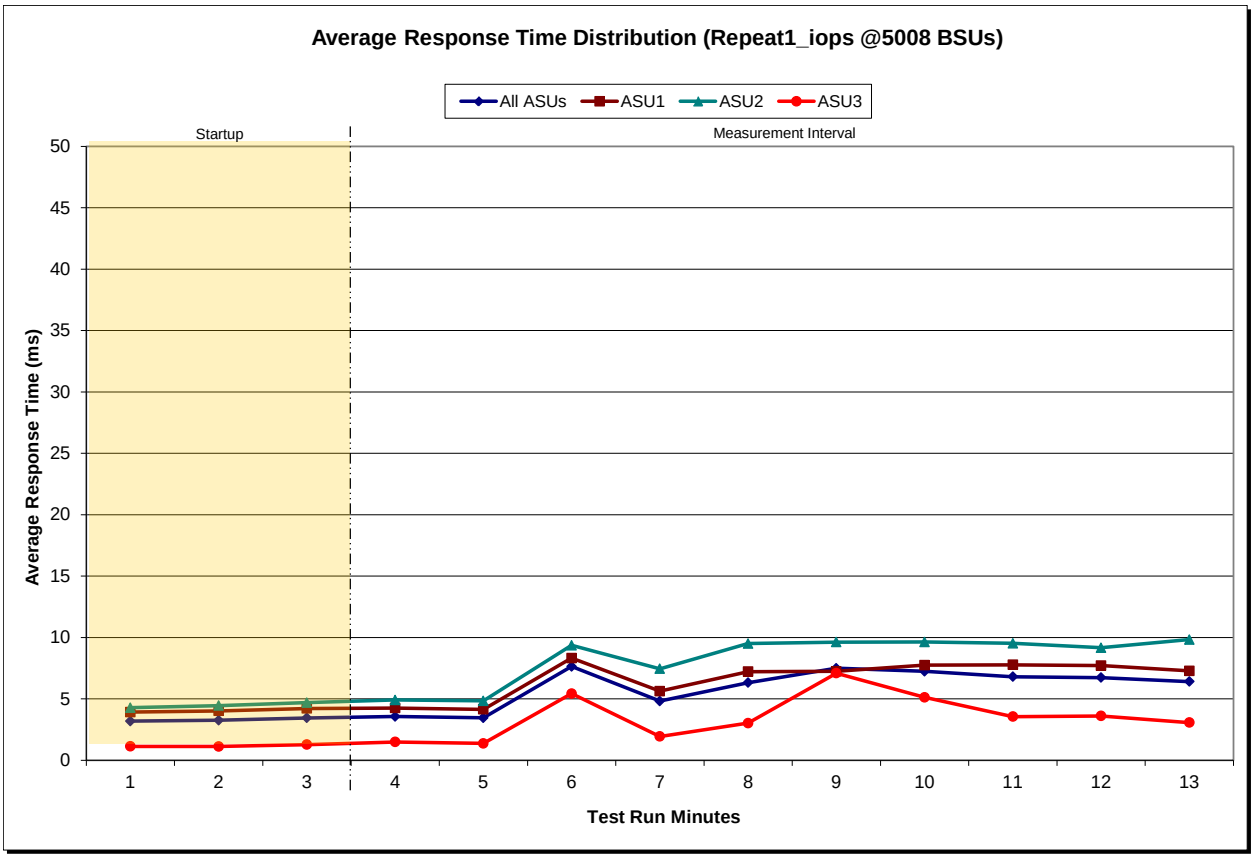
Repeatability 1 IOPS – I/O Request Throughput Distribution Graph



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

5,008 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	3:57:18	4:00:19	0-2	0:03:01
Measurement Interval	4:00:19	4:10:19	3-12	0:10:00
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	3.18	3.92	4.28	1.13
1	3.26	4.02	4.45	1.12
2	3.45	4.21	4.69	1.28
3	3.56	4.26	4.91	1.49
4	3.45	4.15	4.84	1.37
5	7.64	8.32	9.36	5.43
6	4.82	5.63	7.45	1.94
7	6.32	7.22	9.51	3.02
8	7.49	7.25	9.63	7.09
9	7.24	7.74	9.63	5.13
10	6.80	7.77	9.52	3.55
11	6.73	7.71	9.17	3.61
12	6.41	7.28	9.83	3.07
Average	***6.05***	***6.73***	***8.38***	***3.57***

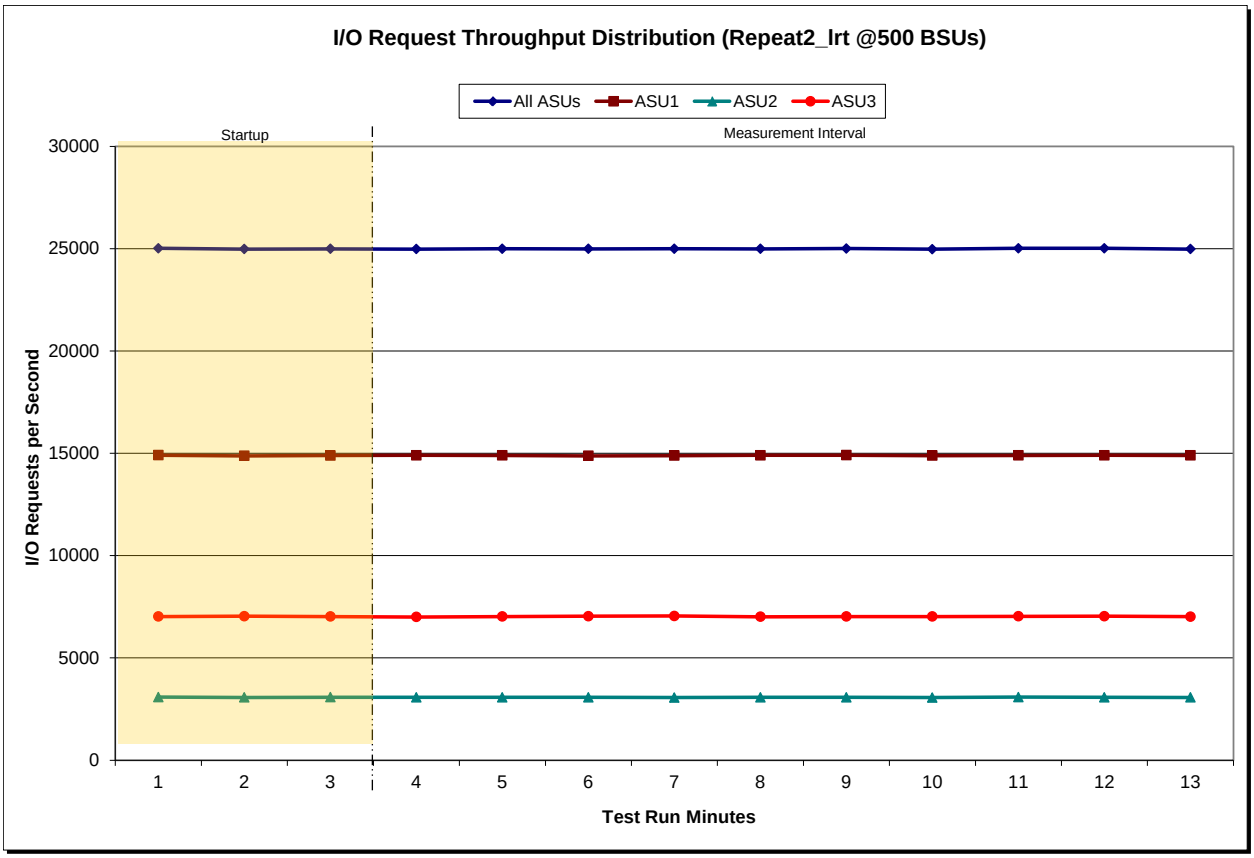
Repeatability 1 IOPS –Average Response Time (ms) Distribution Graph



Repeatability 2 LRT – I/O Request Throughput Distribution Data

500 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	4:21:53	4:24:53	0-2	0:03:00
Measurement Interval	4:24:53	4:34:53	3-12	0:10:00
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	25,017.37	14,908.75	3,086.62	7,022.00
1	24,980.07	14,875.57	3,067.18	7,037.32
2	24,992.63	14,890.38	3,078.77	7,023.48
3	24,980.00	14,903.13	3,073.38	7,003.48
4	24,995.67	14,899.62	3,072.38	7,023.67
5	24,989.03	14,876.73	3,074.80	7,037.50
6	24,996.40	14,885.97	3,063.65	7,046.78
7	24,990.08	14,903.05	3,073.97	7,013.07
8	25,005.65	14,905.73	3,075.80	7,024.12
9	24,975.33	14,888.78	3,065.47	7,021.08
10	25,015.45	14,897.53	3,085.08	7,032.83
11	25,017.90	14,904.25	3,076.27	7,037.38
12	24,980.48	14,894.80	3,069.55	7,016.13
Average	***24,994.60***	***14,895.96***	***3,073.04***	***7,025.61***

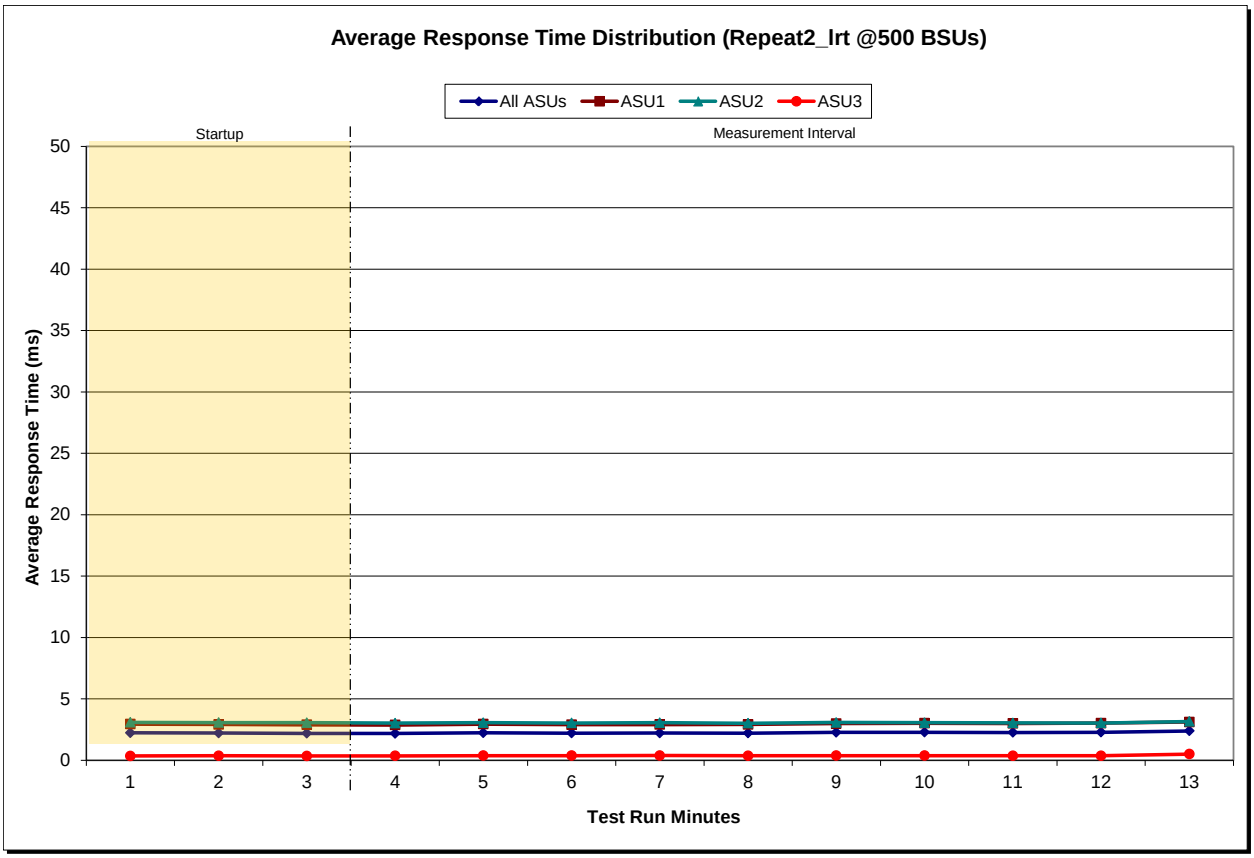
Repeatability 2 LRT – I/O Request Throughput Distribution Graph



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

500 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	4:21:53	4:24:53	0-2	0:03:00
Measurement Interval	4:24:53	4:34:53	3-12	0:10:00
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	2.23	2.94	3.08	0.35
1	2.22	2.92	3.07	0.36
2	2.19	2.88	3.05	0.34
3	2.18	2.87	3.02	0.35
4	2.23	2.94	3.06	0.38
5	2.21	2.90	3.04	0.38
6	2.21	2.91	3.06	0.38
7	2.21	2.91	3.01	0.37
8	2.27	2.99	3.08	0.38
9	2.28	3.01	3.06	0.38
10	2.26	3.00	3.04	0.37
11	2.28	3.03	3.04	0.37
12	2.38	3.11	3.15	0.50
Average	***2.25***	***2.97***	***3.06***	***0.39***

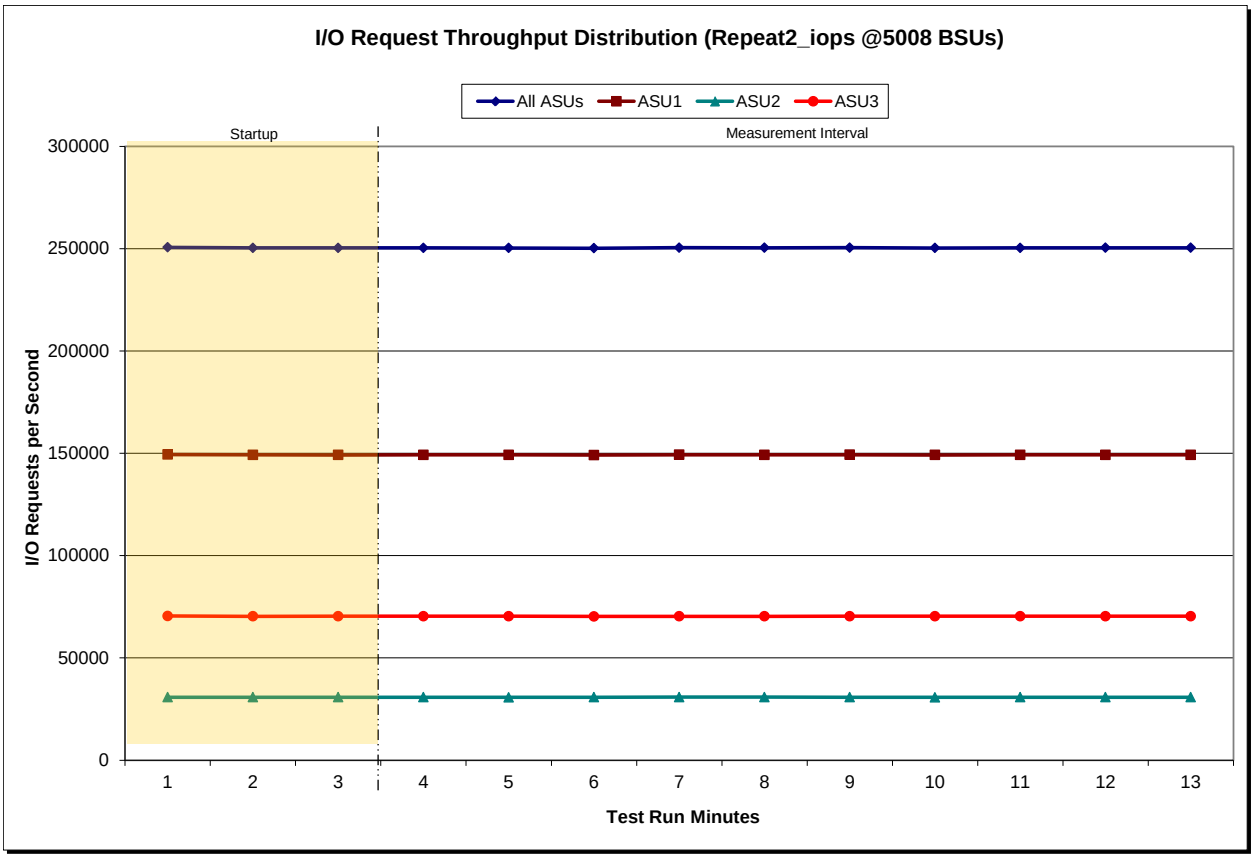
Repeatability 2 LRT –Average Response Time (ms) Distribution Graph



Repeatability 2 IOPS – I/O Request Throughput Distribution Data

5,008 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	4:47:32	4:50:33	0-2	0:03:01
Measurement Interval	4:50:33	5:00:33	3-12	0:10:00
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	250,661.77	149,387.37	30,796.90	70,477.50
1	250,375.47	149,252.30	30,790.93	70,332.23
2	250,406.47	149,213.85	30,824.00	70,368.62
3	250,407.15	149,220.63	30,812.92	70,373.60
4	250,332.12	149,222.57	30,745.70	70,363.85
5	250,226.83	149,157.48	30,775.03	70,294.32
6	250,481.15	149,287.75	30,843.85	70,349.55
7	250,436.32	149,256.33	30,835.73	70,344.25
8	250,490.52	149,313.32	30,798.50	70,378.70
9	250,315.15	149,185.83	30,768.90	70,360.42
10	250,400.85	149,263.00	30,776.17	70,361.68
11	250,423.17	149,260.93	30,799.32	70,362.92
12	250,448.57	149,251.92	30,812.15	70,384.50
Average	***250,396.18***	***149,241.98***	***30,796.83***	***70,357.38***

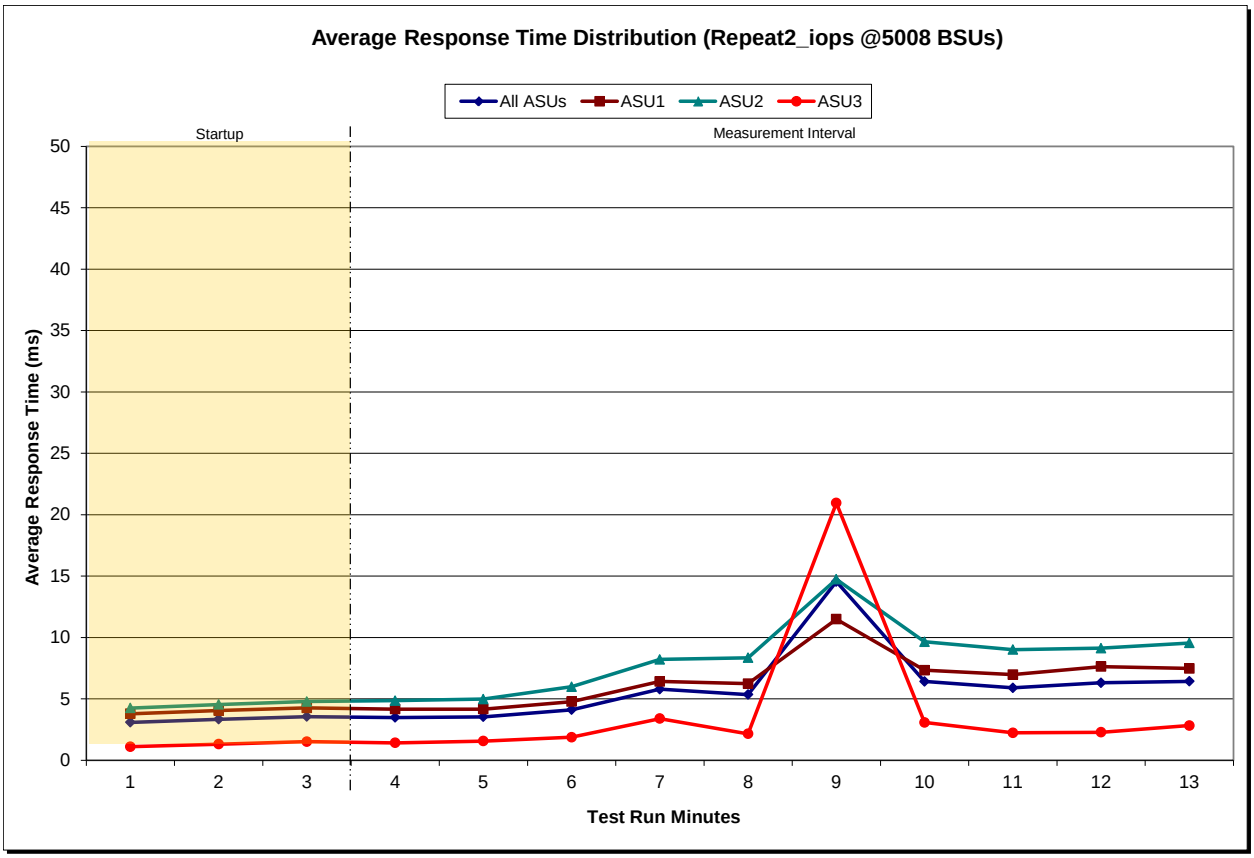
Repeatability 2 IOPS – I/O Request Throughput Distribution Graph



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

5,008 BSUs	Start	Stop	Interval	Duration
Start-Up/Ramp-Up	4:47:32	4:50:33	0-2	0:03:01
Measurement Interval	4:50:33	5:00:33	3-12	0:10:00
60 second intervals	**All ASUs**	**ASU1**	**ASU2**	**ASU3**
0	3.09	3.78	4.25	1.10
1	3.34	4.05	4.54	1.31
2	3.55	4.26	4.78	1.51
3	3.48	4.17	4.86	1.42
4	3.53	4.16	4.98	1.56
5	4.11	4.78	5.99	1.87
6	5.79	6.42	8.22	3.39
7	5.35	6.24	8.35	2.15
8	14.55	11.49	14.74	20.95
9	6.42	7.33	9.65	3.08
10	5.89	6.97	9.01	2.23
11	6.31	7.63	9.12	2.28
12	6.42	7.47	9.55	2.82
Average	***6.18***	***6.66***	***8.44***	***4.18***

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.*

Clauses 5.1.10 and 5.3.13.2

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 5.3.13.3

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	0.0701	0.0349	0.2812
COV	0.006	0.001	0.003	0.002	0.006	0.003	0.003	0.002

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	0.2809	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
COV	0.002	0.000	0.001	0.001	0.002	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	0.2810	0.0700	0.2099	0.0180	0.0699	0.0350	0.2811
COV	0.003	0.002	0.003	0.002	0.005	0.002	0.004	0.002

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	0.002	0.001	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.*

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 76.

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	825,781
Total Number of Logical Blocks Verified	821,467
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 9.2.4.9

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™ S6800T 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 14.

TESTED STORAGE CONFIGURATION (TSC) AND PRICED STORAGE CONFIGURATION DIFFERENCES

Clause 9.4.3.3.7

The Executive Summary shall contain a pricing 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 14.

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.

There were no anomalies or irregularities encountered during the SPC-1 Remote Audit of the Huawei OceanStor™ S6800T.

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: This level will ensure data protection in the event of a single point of failure of any configured storage device. A brief description of the data protection utilized is included in the Executive Summary.

Unprotected: No claim of data protection is asserted in the event of a single point of failure.

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.

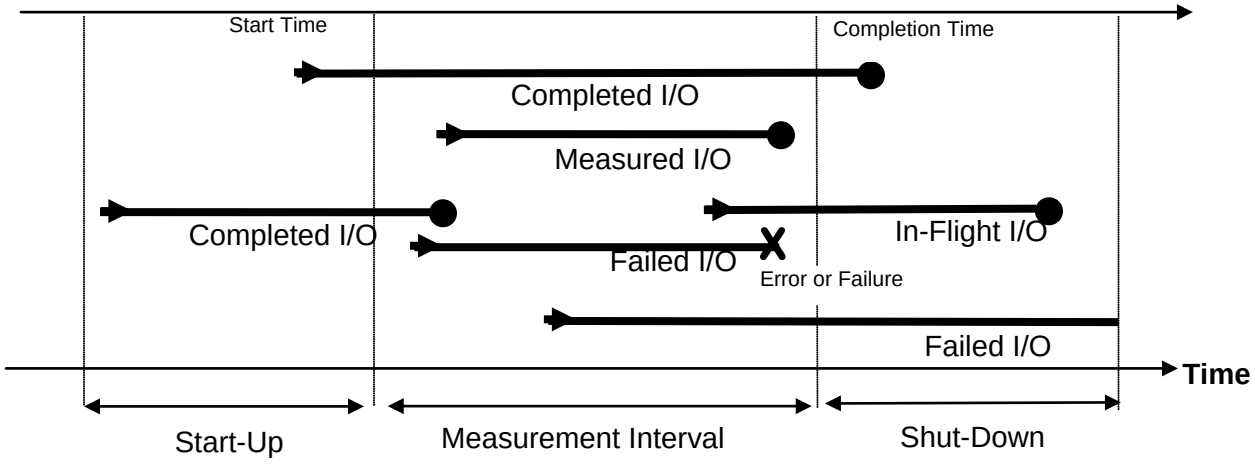
Test: A collection of Test Phases and or Test Runs sharing a common objective.

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

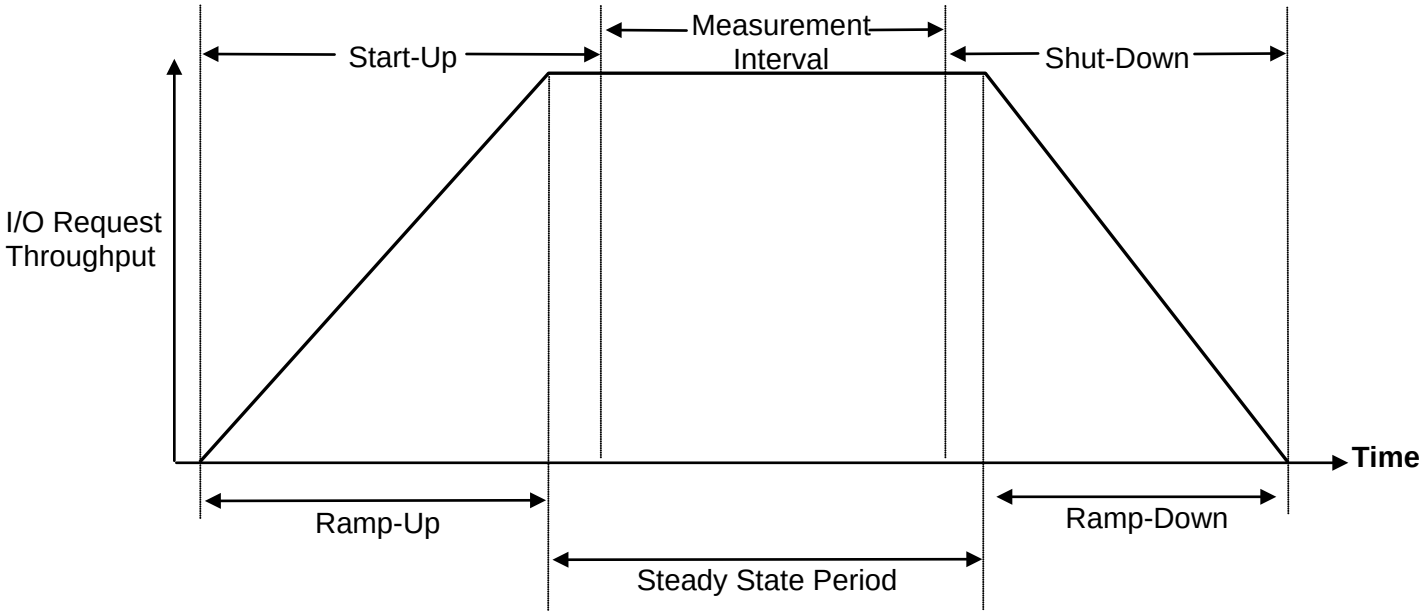
period, Shut-Down period, and Ramp-Down period as illustrated in the "SPC-1 Test Run Components" below. All SPC-1 Test Runs shall have a Steady State period and a Measurement Interval.

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

APPENDIX C: TESTED STORAGE CONFIGURATION (TSC) CREATION

Step 1 Create Host Group and Host

Execute the following commands in OceanStor S6800T's CLI to create one host group *HostGroup001*, and add *Host001* and *Host002* to the host group, then add eight host FC ports WWNs to *Host001* and add the other eight FC ports WWNs to *Host002*.

The ***-t*** parameter is used in the command ***addhost*** to define the host operating system type, and ***-t 0*** means Linux. The ***-type*** parameter of command ***addhostport*** means port type, and ***type 1*** means FC host port.

```
createhostgroup -n HostGroup001
addhost -group 1 -n Host001 -t 0
addhost -group 1 -n Host002 -t 0
addhostport -host 0 -type 1 -wwn 21000024ff2088e3 -n FCInitiator001
addhostport -host 0 -type 1 -wwn 21000024ff2c9524 -n FCInitiator002
addhostport -host 0 -type 1 -wwn 21000024ff2c9434 -n FCInitiator003
addhostport -host 0 -type 1 -wwn 21000024ff2087b6 -n FCInitiator004
addhostport -host 0 -type 1 -wwn 21000024ff2c9435 -n FCInitiator005
addhostport -host 0 -type 1 -wwn 21000024ff2088e2 -n FCInitiator006
addhostport -host 0 -type 1 -wwn 21000024ff2c9525 -n FCInitiator007
addhostport -host 0 -type 1 -wwn 21000024ff2087b7 -n FCInitiator008
addhostport -host 1 -type 1 -wwn 21000024ff2088ce -n FCInitiator009
addhostport -host 1 -type 1 -wwn 21000024ff2088cf -n FCInitiator010
addhostport -host 1 -type 1 -wwn 21000024ff28ea72 -n FCInitiator011
addhostport -host 1 -type 1 -wwn 21000024ff28ea73 -n FCInitiator012
addhostport -host 1 -type 1 -wwn 21000024ff2c9498 -n FCInitiator013
addhostport -host 1 -type 1 -wwn 21000024ff2c9499 -n FCInitiator014
addhostport -host 1 -type 1 -wwn 21000024ff2c952a -n FCInitiator015
addhostport -host 1 -type 1 -wwn 21000024ff2c952b -n FCInitiator016
```

Step 2 Create RAID Groups and LUNs

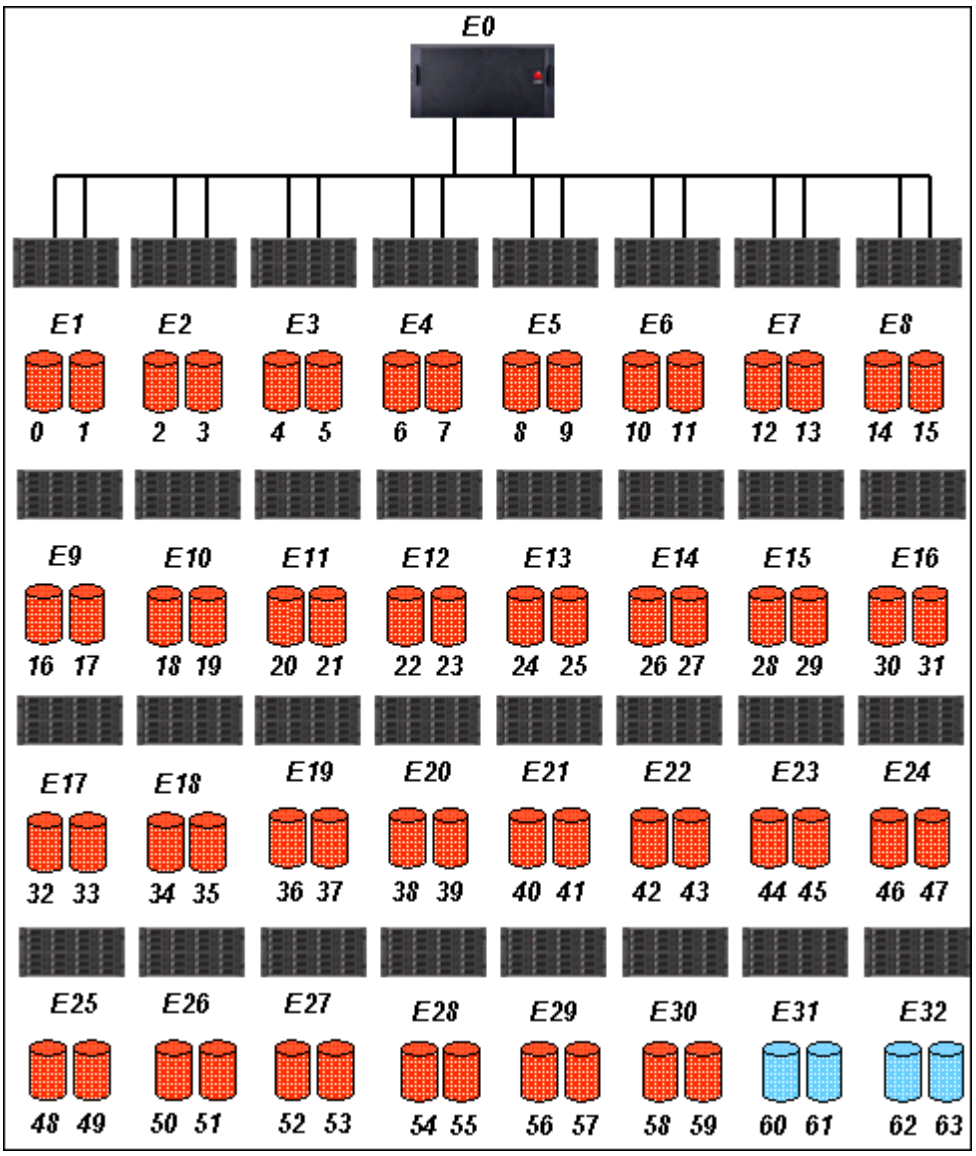
Execute the **mklun.sh** script on one of the Host Systems which has **expect** installed. The script will create 64 RAID Groups, 64 LUNs (one LUN per RAID Group) and map the LUNs to *HostGroup001*.

*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*

The ***createlun*** commands create one LUN per RAID Group and use all of the available capacity for the LUN. The ***addhostmap*** command maps a LUN to a host or a host group.

RAID Groups 0-59 each contain twelve 300 GB disks. Four of the disks in RAID Group 0 are vault disks, which have 23 GiB reserved on each disk to conserve dirty data in case of power failure. RAID Groups 60-63 each contain twelve 600 GB disks.

The RAID-Disk mapping is illustrated below.



Step 3 Change the low and high water level of storage

Execute the following command, using the OceanStor S6800T's CLI to change the low water level to 70 and the high water level to 90.

chgcachewaterlevel -high 90 -low 70

Step 4 Create Volumes on the Master Host System

Execute the **mkvolume.sh** script on the Master Host System to create 46 logical volumes.

1. Create Physical Volumes

Invoke ***pvcreate*** on each block device, creating 64 physical volumes

2. Create Volume Groups

Create **vg0** using: /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

Create **vg1** using: /dev/sdbl, /dev/sdbk, /dev/sdbm, /dev/sdbn

3. Create Logical Volumes

- Create 20 logical volumes on vg0 for ASU-1, each with a capacity of 1822.5 GiB.
- Create 20 logical volumes on vg0 for ASU-2, each with a capacity of 1822.5 GiB.
- Create 6 logical volumes on vg1 for ASU-3, each with a capacity of 1350 GiB.

4. Scan Logical Volumes

Invoke **ssh** to scan physical volumes, volume groups and logical volumes on the second Host System.

Step 5 Change the schedule on each block device

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

Referenced Scripts

mklun.sh

```
#!/bin/bash

stor=129.27.228.89
stor_user=admin
stor_pswd=Admin@storage

export LANG=C

echo "creating LUN ..."

expect <<__END_CREATE_LUN
     spawn ssh $stor_user@$stor
     expect {
            "assword" {
                   send "$stor_pswd\r"
             }
            "yes/no" {
                   send "yes\r"
                   expect "assword"
                      send "$stor_pswd\r"
            }
     }

        expect "#"
        send "login admin\r"
        expect ">"

     set timeout 60

     set lunid 0
     set rgid 0


     foreach enclosure {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: 27: 28: 29: 30: } {
            foreach diskset {0,1,2,3,4,5,6,7,8,9,10,11
12,13,14,15,16,17,18,19,20,21,22,23} {
                   set disk_list ""
                   foreach disk [split \$diskset ,] {
                          set list_unit [string map [list : ,\$disk:] \$enclosure]
                          append disk_list \$list_unit
                   }
                   send "createrg -n ASU-\$rgid -l 10 -num 2 -list \$disk_list\r"
                   expect {
                          "(y/n)" {
                                 send "y\r"
                          }
                          ">" {
                                 send "\r"
                          }
                   }
                   expect ">"
                   send "showrg -rg \$rgid\r"
                   expect ">"
                   if [ expr \$lunid%2 ] {
                          set ctrl b
                   } else {
```

```
                        set ctrl a
                }
                      if { \$lunid >= 12 && \$lunid <= 41 } {
                      send "createlun -rg \$rgid -n ASU-\$lunid -susize 512 -
pretype 1 -value 4 -c \$ctrl\r"
                      } else {
                send "createlun -rg \$rgid -n ASU-\$lunid -susize 512 -pretype 0
-c \$ctrl\r"
                      }
                expect ">"
                sleep 1
                send "showlun -lun \$lunid\r"
                expect ">"
                send "addhostmap -group 1 -devlun \$lunid\r"
                expect ">"
                send "showhostmap -map [expr \$lunid + 1048576]\r"
                expect ">"
                incr lunid
                incr rgid
          }
     }
     foreach enclosure {31: 32:} {
           foreach diskset {0,1,2,3,4,5,6,7,8,9,10,11
12,13,14,15,16,17,18,19,20,21,22,23} {
                set disk_list ""
                foreach disk [split \$diskset ,] {
                      set list_unit [string map [list : ,\$disk:] \$enclosure]
                      append disk_list \$list_unit
                }
                send "createrg -n ASU-\$rgid -l 10 -num 2 -list \$disk_list\r"
                expect {
                      "(y/n)" {
                            send "y\r"
                      }
                      ">" {
                            send "\r"
                      }
                }
                expect ">"
                send "showrg -rg \$rgid\r"
                expect ">"
                if [ expr \$lunid%2 ] {
                      set ctrl b
                } else {
                      set ctrl a
                }

                send "createlun -rg \$rgid -n ASU-\$lunid -susize 512 -pretype 0 -
c \$ctrl\r"
                expect ">"
                sleep 1
                send "showlun -lun \$lunid\r"
                expect ">"
                send "addhostmap -group 1 -devlun \$lunid\r"
                expect ">"
                incr lunid
                incr rgid
          }
     }
     send "exit\r"
     expect "(y/n):"
     send "y\r"
     expect EOF
```

```
__END_CREATE_LUN
```

mkvolume.sh

```
pvcreate /dev/sdc
pvcreate /dev/sdd
pvcreate /dev/sde
pvcreate /dev/sdf
pvcreate /dev/sdg
pvcreate /dev/sdh
pvcreate /dev/sdi
pvcreate /dev/sdj
pvcreate /dev/sdk
pvcreate /dev/sdl
pvcreate /dev/sdm
pvcreate /dev/sdn
pvcreate /dev/sdo
pvcreate /dev/sdp
pvcreate /dev/sdq
pvcreate /dev/sdr
pvcreate /dev/sds
pvcreate /dev/sdt
pvcreate /dev/sdu
pvcreate /dev/sdv
pvcreate /dev/sdw
pvcreate /dev/sdx
pvcreate /dev/sdy
pvcreate /dev/sdz
pvcreate /dev/sdaa
pvcreate /dev/sdab
pvcreate /dev/sdac
pvcreate /dev/sdad
pvcreate /dev/sdae
pvcreate /dev/sdaf
pvcreate /dev/sdag
pvcreate /dev/sdah
pvcreate /dev/sdai
pvcreate /dev/sdaj
pvcreate /dev/sdak
pvcreate /dev/sdal
pvcreate /dev/sdam
pvcreate /dev/sdan
pvcreate /dev/sdao
pvcreate /dev/sdap
pvcreate /dev/sdaq
pvcreate /dev/sdar
pvcreate /dev/sdas
pvcreate /dev/sdat
pvcreate /dev/sdau
pvcreate /dev/sdav
pvcreate /dev/sdaw
pvcreate /dev/sdax
pvcreate /dev/sday
pvcreate /dev/sdaz
pvcreate /dev/sdba
pvcreate /dev/sdbb
pvcreate /dev/sdbc
pvcreate /dev/sdbd
pvcreate /dev/sdbe
pvcreate /dev/sdbf
pvcreate /dev/sdbg
pvcreate /dev/sdbh
```

```
pvcreate /dev/sdbi
pvcreate /dev/sdbj
pvcreate /dev/sdbk
pvcreate /dev/sdbl
pvcreate /dev/sdbm
pvcreate /dev/sdbn



vgcreate vg0 /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
vgcreate vg1 /dev/sdbk /dev/sdbl /dev/sdbm /dev/sdbn

lvcreate -n asu11  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu12  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu13  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu14  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu15  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu16  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu17  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu18  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu19  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu110  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu111  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu112  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu113  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu114  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu115  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu116  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu117  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu118  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu119  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu120  -i 60 -I 512 -L 1822.5g vg0

lvcreate -n asu21  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu22  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu23  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu24  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu25  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu26  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu27  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu28  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu29  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu210  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu211  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu212  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu213  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu214  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu215  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu216  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu217  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu218  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu219  -i 60 -I 512 -L 1822.5g vg0
lvcreate -n asu220  -i 60 -I 512 -L 1822.5g vg0

lvcreate -n asu31  -i 4 -I 512 -L 1350g vg1
lvcreate -n asu32  -i 4 -I 512 -L 1350g vg1
```

```
lvcreate -n asu33  -i 4 -I 512 -L 1350g vg1
lvcreate -n asu34  -i 4 -I 512 -L 1350g vg1
lvcreate -n asu35  -i 4 -I 512 -L 1350g vg1
lvcreate -n asu36  -i 4 -I 512 -L 1350g vg1

for host in host2
do
ssh $host pvscan
ssh $host vgscan
ssh $host lvscan
ssh $host lvchange -ay /dev/vg0/asu11
ssh $host lvchange -ay /dev/vg0/asu12
ssh $host lvchange -ay /dev/vg0/asu13
ssh $host lvchange -ay /dev/vg0/asu14
ssh $host lvchange -ay /dev/vg0/asu15
ssh $host lvchange -ay /dev/vg0/asu16
ssh $host lvchange -ay /dev/vg0/asu17
ssh $host lvchange -ay /dev/vg0/asu18
ssh $host lvchange -ay /dev/vg0/asu19
ssh $host lvchange -ay /dev/vg0/asu110
ssh $host lvchange -ay /dev/vg0/asu111
ssh $host lvchange -ay /dev/vg0/asu112
ssh $host lvchange -ay /dev/vg0/asu113
ssh $host lvchange -ay /dev/vg0/asu114
ssh $host lvchange -ay /dev/vg0/asu115
ssh $host lvchange -ay /dev/vg0/asu116
ssh $host lvchange -ay /dev/vg0/asu117
ssh $host lvchange -ay /dev/vg0/asu118
ssh $host lvchange -ay /dev/vg0/asu119
ssh $host lvchange -ay /dev/vg0/asu120

ssh $host lvchange -ay /dev/vg0/asu21
ssh $host lvchange -ay /dev/vg0/asu22
ssh $host lvchange -ay /dev/vg0/asu23
ssh $host lvchange -ay /dev/vg0/asu24
ssh $host lvchange -ay /dev/vg0/asu25
ssh $host lvchange -ay /dev/vg0/asu26
ssh $host lvchange -ay /dev/vg0/asu27
ssh $host lvchange -ay /dev/vg0/asu28
ssh $host lvchange -ay /dev/vg0/asu29
ssh $host lvchange -ay /dev/vg0/asu210
ssh $host lvchange -ay /dev/vg0/asu211
ssh $host lvchange -ay /dev/vg0/asu212
ssh $host lvchange -ay /dev/vg0/asu213
ssh $host lvchange -ay /dev/vg0/asu214
ssh $host lvchange -ay /dev/vg0/asu215
ssh $host lvchange -ay /dev/vg0/asu216
ssh $host lvchange -ay /dev/vg0/asu217
ssh $host lvchange -ay /dev/vg0/asu218
ssh $host lvchange -ay /dev/vg0/asu219
ssh $host lvchange -ay /dev/vg0/asu220

ssh $host lvchange -ay /dev/vg1/asu31
ssh $host lvchange -ay /dev/vg1/asu32
ssh $host lvchange -ay /dev/vg1/asu33
ssh $host lvchange -ay /dev/vg1/asu34
ssh $host lvchange -ay /dev/vg1/asu35
ssh $host lvchange -ay /dev/vg1/asu36
done
```

scheduler.sh

```
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/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/sdau/queue/scheduler
 echo noop > /sys/block/sdav/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
```

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.

```
pattern=/root/S6800T/random
hd=default,vdbench=/root/vdbench,user=root,shell=ssh
hd=hd1,system=host1
hd=hd2,system=host2
sd=default,threads=1
sd=sd1,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu11,size=1956894474240
sd=sd2,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu12,size=1956894474240
sd=sd3,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu13,size=1956894474240
sd=sd4,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu14,size=1956894474240
sd=sd5,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu15,size=1956894474240
sd=sd6,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu16,size=1956894474240
sd=sd7,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu17,size=1956894474240
sd=sd8,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu18,size=1956894474240
sd=sd9,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu19,size=1956894474240
sd=sd10,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu110,size=1956894474240
sd=sd11,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu111,size=1956894474240
sd=sd12,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu112,size=1956894474240
sd=sd13,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu113,size=1956894474240
sd=sd14,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu114,size=1956894474240
sd=sd15,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu115,size=1956894474240
sd=sd16,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu116,size=1956894474240
sd=sd17,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu117,size=1956894474240
sd=sd18,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu118,size=1956894474240
sd=sd19,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu119,size=1956894474240
sd=sd20,hd=hd1,openflags=o_direct,lun=/dev/vg0/asu120,size=1956894474240
sd=sd21,hd=hd1,openflags=o_direct,lun=/dev/vg1/asu31,size=1449551462400
sd=sd22,hd=hd1,openflags=o_direct,lun=/dev/vg1/asu32,size=1449551462400
sd=sd23,hd=hd1,openflags=o_direct,lun=/dev/vg1/asu33,size=1449551462400

sd=sd24,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu21,size=1956894474240
sd=sd25,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu22,size=1956894474240
sd=sd26,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu23,size=1956894474240
sd=sd27,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu24,size=1956894474240
sd=sd28,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu25,size=1956894474240
sd=sd29,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu26,size=1956894474240
sd=sd30,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu27,size=1956894474240
sd=sd31,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu28,size=1956894474240
sd=sd32,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu29,size=1956894474240
sd=sd33,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu210,size=1956894474240
sd=sd34,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu211,size=1956894474240
sd=sd35,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu212,size=1956894474240
sd=sd36,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu213,size=1956894474240
sd=sd37,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu214,size=1956894474240
sd=sd38,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu215,size=1956894474240
sd=sd39,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu216,size=1956894474240
sd=sd40,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu217,size=1956894474240
sd=sd41,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu218,size=1956894474240
sd=sd42,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu219,size=1956894474240
sd=sd43,hd=hd2,openflags=o_direct,lun=/dev/vg0/asu220,size=1956894474240
sd=sd44,hd=hd2,openflags=o_direct,lun=/dev/vg1/asu34,size=1449551462400
sd=sd45,hd=hd2,openflags=o_direct,lun=/dev/vg1/asu35,size=1449551462400
sd=sd46,hd=hd2,openflags=o_direct,lun=/dev/vg1/asu36,size=1449551462400
```

```
wd=wd1,sd=sd*,rdpct=0,seekpct=-1,xfersize=512K
rd=PREPSSD,wd=wd1,iorate=max,elapsed=360000,interval=10
```

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 and Repeatability 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)


sd=asu1_1,lun=/dev/vg0/asu11 ,size=1956894474240
sd=asu1_2,lun=/dev/vg0/asu12 ,size=1956894474240
sd=asu1_3,lun=/dev/vg0/asu13 ,size=1956894474240
sd=asu1_4,lun=/dev/vg0/asu14 ,size=1956894474240
sd=asu1_5,lun=/dev/vg0/asu15 ,size=1956894474240
sd=asu1_6,lun=/dev/vg0/asu16 ,size=1956894474240
sd=asu1_7,lun=/dev/vg0/asu17 ,size=1956894474240
sd=asu1_8,lun=/dev/vg0/asu18 ,size=1956894474240
sd=asu1_9,lun=/dev/vg0/asu19 ,size=1956894474240
sd=asu1_10,lun=/dev/vg0/asu110,size=1956894474240
sd=asu1_11,lun=/dev/vg0/asu111,size=1956894474240
sd=asu1_12,lun=/dev/vg0/asu112,size=1956894474240
sd=asu1_13,lun=/dev/vg0/asu113,size=1956894474240
sd=asu1_14,lun=/dev/vg0/asu114,size=1956894474240
sd=asu1_15,lun=/dev/vg0/asu115,size=1956894474240
sd=asu1_16,lun=/dev/vg0/asu116,size=1956894474240
sd=asu1_17,lun=/dev/vg0/asu117,size=1956894474240
sd=asu1_18,lun=/dev/vg0/asu118,size=1956894474240
sd=asu1_19,lun=/dev/vg0/asu119,size=1956894474240
sd=asu1_20,lun=/dev/vg0/asu120,size=1956894474240

sd=asu2_1,lun=/dev/vg0/asu21 ,size= 1956894474240
sd=asu2_2,lun=/dev/vg0/asu22 ,size= 1956894474240
sd=asu2_3,lun=/dev/vg0/asu23 ,size= 1956894474240
sd=asu2_4,lun=/dev/vg0/asu24 ,size= 1956894474240
sd=asu2_5,lun=/dev/vg0/asu25  ,size=1956894474240
sd=asu2_6,lun=/dev/vg0/asu26  ,size=1956894474240
sd=asu2_7,lun=/dev/vg0/asu27  ,size=1956894474240
sd=asu2_8,lun=/dev/vg0/asu28  ,size=1956894474240
sd=asu2_9,lun=/dev/vg0/asu29  ,size=1956894474240
sd=asu2_10,lun=/dev/vg0/asu210,size=1956894474240
sd=asu2_11,lun=/dev/vg0/asu211,size=1956894474240
sd=asu2_12,lun=/dev/vg0/asu212,size=1956894474240
sd=asu2_13,lun=/dev/vg0/asu213,size=1956894474240
sd=asu2_14,lun=/dev/vg0/asu214,size=1956894474240
sd=asu2_15,lun=/dev/vg0/asu215,size=1956894474240
sd=asu2_16,lun=/dev/vg0/asu216,size=1956894474240
sd=asu2_17,lun=/dev/vg0/asu217,size=1956894474240
sd=asu2_18,lun=/dev/vg0/asu218,size=1956894474240
sd=asu2_19,lun=/dev/vg0/asu219,size=1956894474240
sd=asu2_20,lun=/dev/vg0/asu220,size=1956894474240
```

```
sd=asu3_1,lun=/dev/vg1/asu31 ,size=1449551462400
sd=asu3_2,lun=/dev/vg1/asu32 ,size=1449551462400
sd=asu3_3,lun=/dev/vg1/asu33 ,size=1449551462400
sd=asu3_4,lun=/dev/vg1/asu34 ,size=1449551462400
sd=asu3_5,lun=/dev/vg1/asu35 ,size=1449551462400
sd=asu3_6,lun=/dev/vg1/asu36 ,size=1449551462400
```

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**, **slave2**, **slave3**… .

```
master=host1
host=slave1
sd=asu1_1,lun=/dev/vg0/asu11 ,size=1956894474240
sd=asu1_2,lun=/dev/vg0/asu12 ,size=1956894474240
sd=asu1_3,lun=/dev/vg0/asu13 ,size=1956894474240
sd=asu1_4,lun=/dev/vg0/asu14 ,size=1956894474240
sd=asu1_5,lun=/dev/vg0/asu15 ,size=1956894474240
sd=asu1_6,lun=/dev/vg0/asu16 ,size=1956894474240
sd=asu1_7,lun=/dev/vg0/asu17 ,size=1956894474240
sd=asu1_8,lun=/dev/vg0/asu18 ,size=1956894474240
sd=asu1_9,lun=/dev/vg0/asu19 ,size=1956894474240
sd=asu1_10,lun=/dev/vg0/asu110,size=1956894474240
sd=asu1_11,lun=/dev/vg0/asu111,size=1956894474240
sd=asu1_12,lun=/dev/vg0/asu112,size=1956894474240
sd=asu1_13,lun=/dev/vg0/asu113,size=1956894474240
sd=asu1_14,lun=/dev/vg0/asu114,size=1956894474240
sd=asu1_15,lun=/dev/vg0/asu115,size=1956894474240
sd=asu1_16,lun=/dev/vg0/asu116,size=1956894474240
sd=asu1_17,lun=/dev/vg0/asu117,size=1956894474240
sd=asu1_18,lun=/dev/vg0/asu118,size=1956894474240
sd=asu1_19,lun=/dev/vg0/asu119,size=1956894474240
sd=asu1_20,lun=/dev/vg0/asu120,size=1956894474240
sd=asu2_1,lun=/dev/vg0/asu21 ,size= 1956894474240
sd=asu2_2,lun=/dev/vg0/asu22 ,size= 1956894474240
sd=asu2_3,lun=/dev/vg0/asu23 ,size= 1956894474240
sd=asu2_4,lun=/dev/vg0/asu24 ,size= 1956894474240
sd=asu2_5,lun=/dev/vg0/asu25  ,size=1956894474240
sd=asu2_6,lun=/dev/vg0/asu26  ,size=1956894474240
sd=asu2_7,lun=/dev/vg0/asu27  ,size=1956894474240
sd=asu2_8,lun=/dev/vg0/asu28  ,size=1956894474240
sd=asu2_9,lun=/dev/vg0/asu29  ,size=1956894474240
sd=asu2_10,lun=/dev/vg0/asu210,size=1956894474240
sd=asu2_11,lun=/dev/vg0/asu211,size=1956894474240
sd=asu2_12,lun=/dev/vg0/asu212,size=1956894474240
sd=asu2_13,lun=/dev/vg0/asu213,size=1956894474240
sd=asu2_14,lun=/dev/vg0/asu214,size=1956894474240
sd=asu2_15,lun=/dev/vg0/asu215,size=1956894474240
sd=asu2_16,lun=/dev/vg0/asu216,size=1956894474240
sd=asu2_17,lun=/dev/vg0/asu217,size=1956894474240
sd=asu2_18,lun=/dev/vg0/asu218,size=1956894474240
sd=asu2_19,lun=/dev/vg0/asu219,size=1956894474240
sd=asu2_20,lun=/dev/vg0/asu220,size=1956894474240
sd=asu3_1,lun=/dev/vg1/asu31 ,size=1449551462400
sd=asu3_2,lun=/dev/vg1/asu32 ,size=1449551462400
sd=asu3_3,lun=/dev/vg1/asu33 ,size=1449551462400
sd=asu3_4,lun=/dev/vg1/asu34 ,size=1449551462400
sd=asu3_5,lun=/dev/vg1/asu35 ,size=1449551462400
sd=asu3_6,lun=/dev/vg1/asu36 ,size=1449551462400
```

SPC-2 Persistence Test

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,jvms=8,maxstreams=200

sd=sd1,lun=/dev/vg0/asu11,size=1956894474240
sd=sd2,lun=/dev/vg0/asu12,size=1956894474240
sd=sd3,lun=/dev/vg0/asu13,size=1956894474240
sd=sd4,lun=/dev/vg0/asu14,size=1956894474240
sd=sd5,lun=/dev/vg0/asu15,size=1956894474240
sd=sd6,lun=/dev/vg0/asu16,size=1956894474240
sd=sd7,lun=/dev/vg0/asu17,size=1956894474240
sd=sd8,lun=/dev/vg0/asu18,size=1956894474240
sd=sd9,lun=/dev/vg0/asu19,size=1956894474240
sd=sd10,lun=/dev/vg0/asu110,size=1956894474240
sd=sd11,lun=/dev/vg0/asu111,size=1956894474240
sd=sd12,lun=/dev/vg0/asu112,size=1956894474240
sd=sd13,lun=/dev/vg0/asu113,size=1956894474240
sd=sd14,lun=/dev/vg0/asu114,size=1956894474240
sd=sd15,lun=/dev/vg0/asu115,size=1956894474240
sd=sd16,lun=/dev/vg0/asu116,size=1956894474240
sd=sd17,lun=/dev/vg0/asu117,size=1956894474240
sd=sd18,lun=/dev/vg0/asu118,size=1956894474240
sd=sd19,lun=/dev/vg0/asu119,size=1956894474240
sd=sd20,lun=/dev/vg0/asu120,size=1956894474240
sd=sd21,lun=/dev/vg0/asu21,size=1956894474240
sd=sd22,lun=/dev/vg0/asu22,size=1956894474240
sd=sd23,lun=/dev/vg0/asu23,size=1956894474240
sd=sd24,lun=/dev/vg0/asu24,size=1956894474240
sd=sd25,lun=/dev/vg0/asu25,size=1956894474240
sd=sd26,lun=/dev/vg0/asu26,size=1956894474240
sd=sd27,lun=/dev/vg0/asu27,size=1956894474240
sd=sd28,lun=/dev/vg0/asu28,size=1956894474240
sd=sd29,lun=/dev/vg0/asu29,size=1956894474240
sd=sd30,lun=/dev/vg0/asu210,size=1956894474240
sd=sd31,lun=/dev/vg0/asu211,size=1956894474240
sd=sd32,lun=/dev/vg0/asu212,size=1956894474240
sd=sd33,lun=/dev/vg0/asu213,size=1956894474240
sd=sd34,lun=/dev/vg0/asu214,size=1956894474240
sd=sd35,lun=/dev/vg0/asu215,size=1956894474240
sd=sd36,lun=/dev/vg0/asu216,size=1956894474240
sd=sd37,lun=/dev/vg0/asu217,size=1956894474240
sd=sd38,lun=/dev/vg0/asu218,size=1956894474240
sd=sd39,lun=/dev/vg0/asu219,size=1956894474240
sd=sd40,lun=/dev/vg0/asu220,size=1956894474240
sd=sd41,lun=/dev/vg1/asu31,size=1449551462400
sd=sd42,lun=/dev/vg1/asu32,size=1449551462400
sd=sd43,lun=/dev/vg1/asu33,size=1449551462400
sd=sd44,lun=/dev/vg1/asu34,size=1449551462400
sd=sd45,lun=/dev/vg1/asu35,size=1449551462400
sd=sd46,lun=/dev/vg1/asu36,size=1449551462400
```

SPC-2 Persistence Test Run 1 (write phase)

common command lines

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

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=166
```

SPC-2 Persistence Test Run 2 (read phase)

common command lines

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

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

ASU Pre-Fill, Primary Metrics Test, Repeatability Test, Persistence Test Run 1, TSC power off/power on and Persistence Test Run 2

The following script was used to execute the required ASU pre-fill, 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)*, and Persistence Test Run 1 in an uninterrupted sequence. The script pauses until the required TSC power off/power on cycle is completed then executes Persistence Test Run 2.

Due to storage capacity limitations of the SPC-1 Workload Generator, the test results from the SPC-1 Persistence Test were discarded and the SPC-2 Persistence Test used to provide the same test functionality.

```
JAVA="/usr/java/jre1.7.0_06/bin/java -Xms1536m -Xmx1536m -Xss256k"
EXEDIR=/root/S6800T

expect shstorage.tcl > profile1_storage.log
date > profile1_volume.log
pvdisplay >> profile1_volume.log
vgdisplay >> profile1_volume.log
lvdisplay >> profile1_volume.log
date >> profile1_volume.log


N=1
for host in host1 host2
do
  ssh $host rm -rf $EXEDIR/output
  ssh $host rm -rf $EXEDIR/config
  ssh $host mkdir $EXEDIR/output
  ssh $host mkdir $EXEDIR/config
  for((i=1;i<=40;i++))
  do
    echo "start slave$N on $host"
    echo "master=host1" > $EXEDIR/config/slave$N.cfg
    echo "host=slave$N" >> $EXEDIR/config/slave$N.cfg

echo "sd=asu1_1,lun=/dev/vg0/asu11 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_2,lun=/dev/vg0/asu12 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_3,lun=/dev/vg0/asu13 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_4,lun=/dev/vg0/asu14 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_5,lun=/dev/vg0/asu15 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_6,lun=/dev/vg0/asu16 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_7,lun=/dev/vg0/asu17 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_8,lun=/dev/vg0/asu18 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_9,lun=/dev/vg0/asu19 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_10,lun=/dev/vg0/asu110,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
```

```
echo "sd=asu1_11,lun=/dev/vg0/asu111,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_12,lun=/dev/vg0/asu112,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_13,lun=/dev/vg0/asu113,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_14,lun=/dev/vg0/asu114,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_15,lun=/dev/vg0/asu115,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_16,lun=/dev/vg0/asu116,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_17,lun=/dev/vg0/asu117,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_18,lun=/dev/vg0/asu118,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_19,lun=/dev/vg0/asu119,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_20,lun=/dev/vg0/asu120,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg

echo "sd=asu2_1,lun=/dev/vg0/asu21 ,size= 1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_2,lun=/dev/vg0/asu22 ,size= 1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_3,lun=/dev/vg0/asu23 ,size= 1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_4,lun=/dev/vg0/asu24 ,size= 1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_5,lun=/dev/vg0/asu25  ,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_6,lun=/dev/vg0/asu26  ,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_7,lun=/dev/vg0/asu27  ,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_8,lun=/dev/vg0/asu28  ,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_9,lun=/dev/vg0/asu29  ,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_10,lun=/dev/vg0/asu210,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_11,lun=/dev/vg0/asu211,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_12,lun=/dev/vg0/asu212,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_13,lun=/dev/vg0/asu213,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_14,lun=/dev/vg0/asu214,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_15,lun=/dev/vg0/asu215,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_16,lun=/dev/vg0/asu216,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_17,lun=/dev/vg0/asu217,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_18,lun=/dev/vg0/asu218,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_19,lun=/dev/vg0/asu219,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_20,lun=/dev/vg0/asu220,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
```

```
echo "sd=asu3_1,lun=/dev/vg1/asu31 ,size=1449551462400" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_2,lun=/dev/vg1/asu32 ,size=1449551462400" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_3,lun=/dev/vg1/asu33 ,size=1449551462400" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_4,lun=/dev/vg1/asu34 ,size=1449551462400" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_5,lun=/dev/vg1/asu35 ,size=1449551462400" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_6,lun=/dev/vg1/asu36 ,size=1449551462400" >>
$EXEDIR/config/slave$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 metrics.cfg spc1.cfg

$JAVA -cp ../spc1 repeat1 -b 5008
$JAVA -cp ../spc1 repeat2 -b 5008

for host in host1 host2
do
  ssh $host killall java
done

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

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

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

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

Repeatability Test, SPC-2 Persistence Test Run 1, TSC power off/power on and SPC-2 Persistence Test Run 2

The following script was used to execute the Repeatability Test *(Repeatability Test Phase 1 and Repeatability Test Phase 2)*, and SPC-2 Persistence Test Run 1 in an uninterrupted sequence. The script pauses until the required TSC power off/power on cycle is completed then executes SPC-2 Persistence Test Run 2.

The Repeatability Test was re-executed to compare those results with the initial set of measurement for comparison, which were verified to be "run to run" equivalent.

```
JAVA="/usr/java/jre1.7.0_06/bin/java -Xms1536m -Xmx1536m -Xss256k"
EXEDIR=/root/S6800T

expect shstorage.tcl > profile1_storage.log
date > profile1_volume.log
pvdisplay >> profile1_volume.log
vgdisplay >> profile1_volume.log
lvdisplay >> profile1_volume.log
date >> profile1_volume.log


N=1
for host in host1 host2
do
  ssh $host rm -rf $EXEDIR/output
  ssh $host rm -rf $EXEDIR/config
  ssh $host mkdir $EXEDIR/output
  ssh $host mkdir $EXEDIR/config
  for((i=1;i<=40;i++))
  do
    echo "start slave$N on $host"
    echo "master=host1" > $EXEDIR/config/slave$N.cfg
    echo "host=slave$N" >> $EXEDIR/config/slave$N.cfg

echo "sd=asu1_1,lun=/dev/vg0/asu11 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_2,lun=/dev/vg0/asu12 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_3,lun=/dev/vg0/asu13 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_4,lun=/dev/vg0/asu14 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_5,lun=/dev/vg0/asu15 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_6,lun=/dev/vg0/asu16 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_7,lun=/dev/vg0/asu17 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_8,lun=/dev/vg0/asu18 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_9,lun=/dev/vg0/asu19 ,size=1956894474240 " >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_10,lun=/dev/vg0/asu110,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_11,lun=/dev/vg0/asu111,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_12,lun=/dev/vg0/asu112,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
```

```
echo "sd=asu1_13,lun=/dev/vg0/asu113,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_14,lun=/dev/vg0/asu114,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_15,lun=/dev/vg0/asu115,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_16,lun=/dev/vg0/asu116,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_17,lun=/dev/vg0/asu117,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_18,lun=/dev/vg0/asu118,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_19,lun=/dev/vg0/asu119,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu1_20,lun=/dev/vg0/asu120,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg

echo "sd=asu2_1,lun=/dev/vg0/asu21 ,size= 1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_2,lun=/dev/vg0/asu22 ,size= 1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_3,lun=/dev/vg0/asu23 ,size= 1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_4,lun=/dev/vg0/asu24 ,size= 1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_5,lun=/dev/vg0/asu25  ,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_6,lun=/dev/vg0/asu26  ,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_7,lun=/dev/vg0/asu27  ,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_8,lun=/dev/vg0/asu28  ,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_9,lun=/dev/vg0/asu29  ,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_10,lun=/dev/vg0/asu210,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_11,lun=/dev/vg0/asu211,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_12,lun=/dev/vg0/asu212,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_13,lun=/dev/vg0/asu213,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_14,lun=/dev/vg0/asu214,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_15,lun=/dev/vg0/asu215,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_16,lun=/dev/vg0/asu216,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_17,lun=/dev/vg0/asu217,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_18,lun=/dev/vg0/asu218,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_19,lun=/dev/vg0/asu219,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu2_20,lun=/dev/vg0/asu220,size=1956894474240" >>
$EXEDIR/config/slave$N.cfg

echo "sd=asu3_1,lun=/dev/vg1/asu31 ,size=1449551462400" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_2,lun=/dev/vg1/asu32 ,size=1449551462400" >>
$EXEDIR/config/slave$N.cfg
```

```
echo "sd=asu3_3,lun=/dev/vg1/asu33 ,size=1449551462400" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_4,lun=/dev/vg1/asu34 ,size=1449551462400" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_5,lun=/dev/vg1/asu35 ,size=1449551462400" >>
$EXEDIR/config/slave$N.cfg
echo "sd=asu3_6,lun=/dev/vg1/asu36 ,size=1449551462400" >>
$EXEDIR/config/slave$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 metrics.cfg spc1.cfg

$JAVA -cp ../spc1 repeat1 -b 5008
$JAVA -cp ../spc1 repeat2 -b 5008

for host in host1 host2
do
  ssh $host killall java
done

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

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

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

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

APPENDIX F: THIRD-PARTY QUOTATION

Even Enterprises
12439 Magnolia Blvd. Suite# 303
N. Hollywood, CA 91607
Phone: 818-793-4403
Fax: 818-302-3344

Even Enterprises Quotation

Issued To: «Company»
«First_Name» «Last_Name»
«City», «State»
Phone: «Business_Phone»
Email: «Email»

DATE ISSUED: October 9, 2012
PAYMENT TERMS: Credit Card / Net 30 (upon approval)
Salesperson: Esti Even
Prices subject to change without notice
Prices Valid for 90 days Confidential

No.	Model	Description	Add	Unit Price ($)	Total Price ($)
1	Phase				
1.1	Location				
1.1.1	S6800T		1		585,247.46
1.1.1.1	Control module				75,196.66
	S64-2C384G-BASE	SPE61C0200-64 Controller Enclosure(Dual Controller,AC,384GB Cache,with UPS Cache Protected Module,without Front-End & Back-End Port,with HS Storage Array Control System Software)	1	75,196.66	75,196.66
1.1.1.2	Disk Enclosure				102,576.32
	DAE12435U4-AC	DAE12435U4-03 Disk Enclosure(4U,3.5",AC,SAS Expansion Module,without Disk Unit,with HS SAS in Band Management Software)	32	3,205.51	102,576.32
1.1.1.3	Hard Disk Drives				371,032.32
	SAS300-15K	300GB 15K RPM SAS Disk Unit(3.5")	720	465.65	335,268.00
	SAS600-15K	600GB 15K RPM SAS Disk Unit(3.5")	48	745.09	35,764.32
1.1.1.4	IO Interface				13,149.25
	LPU4F8	4*8Gbps Fibre Channel I/O modules(Total 4 ports)	4	1,218.20	4,872.78
	LPU2S6	2*24Gbps SAS-wide I/O modules(Total 2 ports)	8	1,034.56	8,276.47
1.1.1.5	Accessory				1,153.00
	SS-OP-D-LC-M-3	Patchcord,DLC/PC-DLC/PC,Multimode,2mm Parallel,3m	16	11.00	176.00
	MINI-SAS-3	Purchased Cable,MiniSAS Cable,Key246,3m	1	67.00	67.00
	MINI-SAS-6	Outsourcing Cable,External Mini SAS Cable,6.0m,External Mini SAS 26 Pin Plug,24AWG*8P,External Mini SAS 26 Pin Plug	10	91.00	910.00
1.1.1.6	Storage management software				8,555.91
	LIC-S6A-ISM02-V1R5	HS Integrated Storage Manager-Device Management License for OceanStor S6800T(V100R005)	1	6,617.39	6,617.39
	LIC-UltraPath02-V1R5	OceanStor HS UltraPath Software License	1	1,938.51	1,938.51
1.1.1.7	Third Party				13,584.00
	N8GHBA000	QLOGIC QLE2562 HBA Card,PCIE,8Gbps DualPort ,Fiber Channel Multimode LC Optic Interface,English Manual,Driver CD	8	1,698.00	13,584.00
Total of Product					585,247.46
1.1.1.8	Maintenance Support Service				245,214.60
	Hi-Care Premier On-Site Service (3 years)		1	216,753.00	216,753.00
	300GB 15K RPM SAS Disk Unit(3.5") upgrade to On-Site Serice (3 years)		720	35.72	25,718.40
	600GB 15K RPM SAS Disk Unit(3.5") upgrade to On-Site Service (3 years)		48	57.15	2,743.20

Total of Service (3 years)					**245,214.60**
Total Price					**830,462.06**
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.					