SPC BENCHMARK 1™

FULL DISCLOSURE REPORT

TELECOMMUNICATIONS TECHNOLOGY ASSOCIATION
JET-SPEED™ HHS3124F / HHS2112F (10 NODES)

SPC-1 V3.8

SUBMISSION IDENTIFIER: A31019

SUBMITTED FOR REVIEW: DECEMBER 4, 2018

First Edition – December 2018

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

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

Trademarks

SPC Benchmark 1, SPC-1, SPC-1 IOPS, SPC-1 LRT and SPC-1 Price-Performance are trademarks of the Storage Performance Council.

TTA and the TTA logo are trademarks or registered trademarks of the Telecommunications Technology Association in Korea and other countries. Taejin and Jet-speed™ are trademarks or registered trademarks of Taejin Infotech Co., Ltd. in Korea and other countries. All other brands, trademarks, and product names are the property of their respective owners.

Benchmark Specification and Glossary

The official SPC Benchmark 1™ (SPC-1™) specification is available on the website of the Storage Performance Council (SPC) at www.spcresults.org.

The SPC-1™ specification contains a glossary of the SPC-1™ terms used in this publication.

Table of Contents

AUDIT CERTIFICATION

InfoSizing
The Right Metric For Sizing IT

Storage Performance Council
Certified Auditor

Cheol-Soon Park
Telecommunications Technology Association
47, Bundang-ro, Bundang-gu, Seongnam-city,
Gyeonggi-do, 13591
Republic of Korea

December 3, 2018

I verified the SPC Benchmark 1™ (SPC-1™ Revision3.8) test execution and performance results of the following Tested Storage Product:

JET-SPEED™ HHS3124F / HHS2112F (10 NODES)

The results were:

SPC-1 IOPS™	**2,410,271**
SPC-1 Price-Performance™	**$287.01/SPC-1 KIOPS™**
SPC-1 IOPS™ Response Time	0.311 ms
SPC-1 Overall Response Time	0.205 ms
SPC-1 ASU Capacity	46,789 GB
SPC-1 Space Effectiveness Ratio	NA
SPC-1 ASU Price	$14.79/GB
SPC-1 Total System Price	$691,767.07

In my opinion, these performance results were produced in compliance with the SPC requirements for the benchmark.

The testing was executed using the SPC-1 Toolkit Version 0xe28e08v3.0.2. The audit process was conducted in accordance with the SPC Policies and met the requirements for the benchmark.

A Letter of Good Faith was issued by the Test Sponsor, stating the accuracy and completeness of the documentation and testing data provided in support of the audit of this result.

A Full Disclosure Report for this result was prepared by InfoSizing, reviewed and approved by the Test Sponsor, and can be found at **www.spcresults.org** under the Submission Identifier **A31019**.

20 Kreg Lane • Manitou Springs, CO 80829 • 719-473-7555 • www.sizing.com

The independent audit process conducted by InfoSizing included the verifications of the following items:

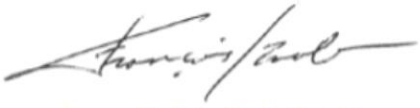
- The physical capacity of the data repository;
- The total capacity of the Application Storage Unit (ASU);
- The accuracy of the Benchmark Configuration diagram;
- The tuning parameters used to configure the Benchmark Configuration;
- The Workload Generator commands used to execute the testing;
- The validity and integrity of the test result files;
- The compliance of the results from each performance test;
- The compliance of the results from the persistence test;
- The compliance of the submitted pricing model; and
- The differences between the tested and the priced configuration, if any.

The Full Disclosure Report for this result was prepared in accordance with the disclosure requirements set forth in the specification for the benchmark.

The following benchmark requirements, if any, were waived according to the SPC Policies:

- None.

Respectfully Yours,



François Raab, Certified SPC Auditor

20 Kreg Lane • Manitou Springs, CO 80829 • 719-473-7555 • www.sizing.com

LETTER OF GOOD FAITH

TTA 한국정보통신기술협회
Telecommunications Technology Association

47, Bundang-ro, Bundang-gu, Seongnam-city,
Gyeonggi-do, 13591, Republic of Korea
TEL : 82-31-724-0114

December 3, 2018

From: Telecommunications Technology Association

To: Mr. Francois Raab, Certified SPC Auditor
InfoSizing, Inc.
20 Kreg Lane
Manitou Springs, CO 80829

Subject: SPC-1 Letter of Good Faith for the Telecommunications Technology Association JetSpeed™ HHS3124F/2112F

Telecommunications Technology Association 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 V3.8 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: Park C. S. Date: Dec. 3, 2018

Cheol-Soon Park
Vice President,
Telecommunications Technology Association

SPC BENCHMARK 1™

EXECUTIVE SUMMARY

TELECOMMUNICATIONS TECHNOLOGY ASSOCIATION JET-SPEED HHS3124F / HHS2112F (10 NODES)

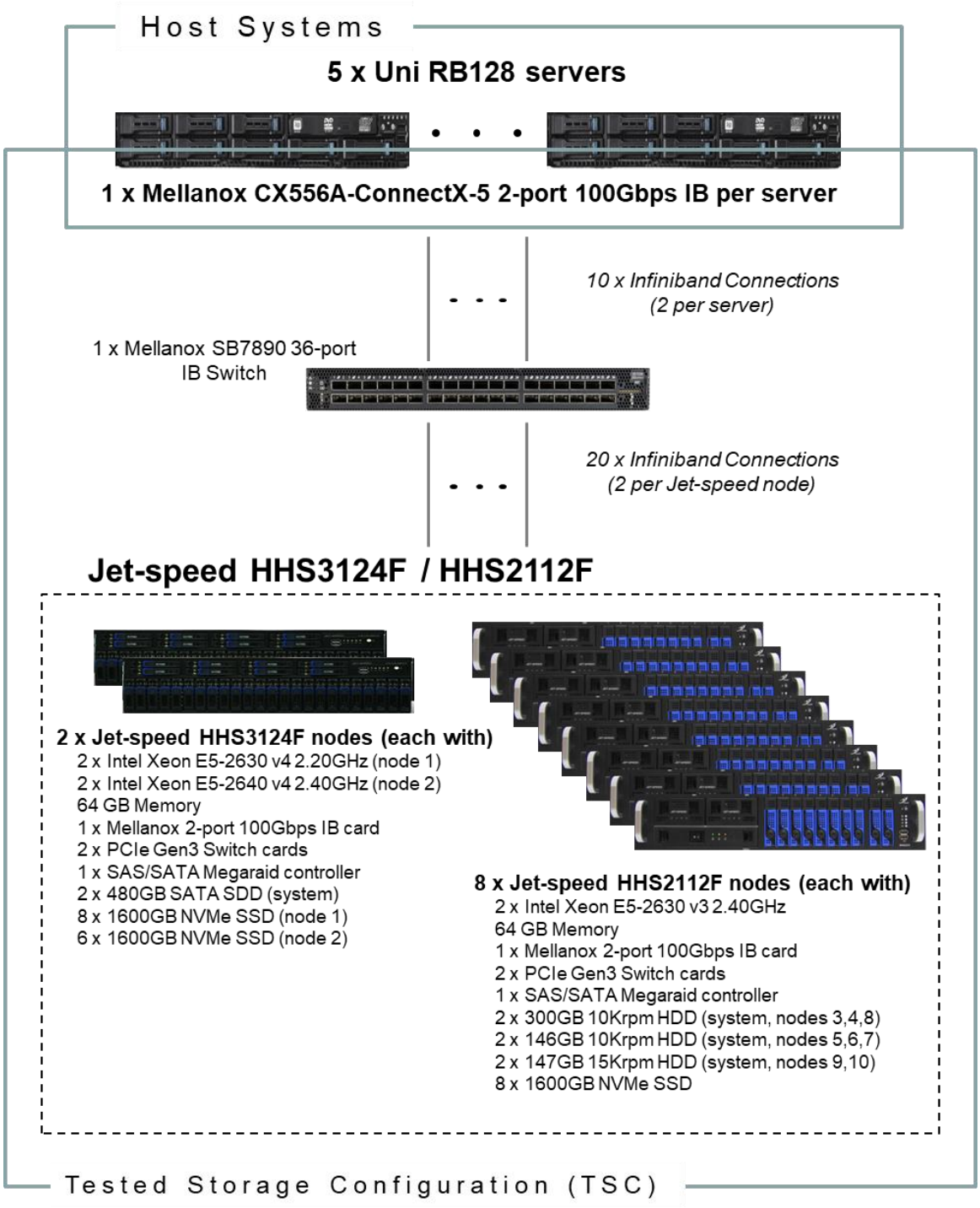
SPC-1 IOPS™	**2,410,271**
SPC-1 Price-Performance™	**$287.01/SPC-1 KIOPS™**
SPC-1 IOPS™ Response Time	0.311 ms
SPC-1 Overall Response Time	0.205 ms
SPC-1 ASU Capacity	46,789 GB
SPC-1 Space Effectiveness Ratio	NA
SPC-1 ASU Price	$14.79/GB
SPC-1 Total System Price	$691,767.07
Data Protection Level	Protected 1 (RAID 1+0)
Physical Storage Capacity	129,976.GB
Pricing Currency / Target Country	U.S. Dollars / Korea

SPC-1 V3.8

SUBMISSION IDENTIFIER: A31019

SUBMITTED FOR REVIEW: DECEMBER 4, 2018

Benchmark Configuration Diagram



Tested Storage Product Description

The intelligent Hyper Hybrid Storage HHS3124F/HHS2112F present an ultra-high speed, all-NVMe RAID storage platform. The systems guarantee stability and performance through its own control system and management system using Intel® Xeon® controller.

Data stability and redundancy is secured by implementing a PCIe RAID architecture. Flash SSD performance degradation and breakdown rates that are caused by frequent write operations are innovatively reduced. Especially, the HHS3124F, which supports concurrent use of NVMe Flash SSDs and Tajin Infotech's NVMe RAM-based SSDs, can activate data cache engine which is configured to improve data I/O performance using the controller's own cache as well as the NVMe RAM-based SSD.

Note: NVMe RAM-based SSDs were not used for this SPC-1 result.

For more details, visit:

http://www.taejin.co.kr/wp/?page_id=11611

Priced Storage Configuration Components

5 x Mellanox CX556A-ConnectX-5 2-port 100Gbps IB cards (1 per host)
2 x Jet-speed HHS3124F storage nodes, each with: 2 x Intel Xeon CPU E5-2630 v4 2.20GHz 10-core (node 1) 2 x Intel Xeon CPU E5-2640 v4 2.40GHz 10-core (node 2) 4 x 16GB DDR4 Memory 1 x Mellanox 2-port 100Gbps IB card 1 x SAS/SATA MegaRAID controller 2 x PCIe Gen3 switch cards 2 x 480GB SATA 6Gbps SSD (System) 8 x 1600GB NVMe SSD 2.5' SFF (node 1) 6 x 1600GB NVMe SSD 2.5' SFF (node 2)
8 x Jet-speed HHS2112F storage nodes, each with: 2 x Intel Xeon CPU E5-2630 v3 2.40GHz 8-core 4 x 16GB DDR4 Memory 2 x PCIe Gen3 switch cards 1 x Mellanox 2-port 100Gbps IB card 1 x SAS/SATA MegaRAID controller 2 x 300GB SAS 10Krpm HDD (System, nodes 3,4,8) 2 x 146GB SAS 10Krpm HDD (System, nodes 5,6,7) 2 x 147GB SAS 15Krpm HDD (System, nodes 9,10) 8 x 1600GB NVMe SSD 2.5' SFF
Mellanox SB7890 100 Gbps 36-port IB Switch

Storage Configuration Pricing

	Description	Qty	Unit Price	Ext. Price	Disc.	Disc. Price
Hardware & Software						
90102-0001-00A	HHS2112F All NVMe Storage (2U / 12 disk bays Dual Intel Xeon E5-2600v3/v4 Family, 16x DIMM Slots, 800W redundant PSU 80Plus platinum)	8	50,379.73	403,037.82	50%	201,518.91
90102-0002-00A	HHS3124F All NVMe Storage (3U / 24 disk bays Dual Intel Xeon E5-2600v3/v4 Family, 16x DIMM Slots, 1000W redundant PSU 80Plus platinum)	2	91,777.78	183,555.56	50%	91,777.78
20204-0001-00A	DDR4 16GB PC4-17000 ECC/REG (16GB PC4-17000 DDR Rdimm Ecc Reg RX8 CL17 1.2V)	28	179.64	5,029.89	0%	5,029.89
20204-0002-00A	DDR4 16GB PC4-19200 ECC/REG (16GB PC4-19200 DDR Rdimm Ecc Reg RX8 CL17 1.2V)	12	179.64	2,155.67	0%	2,155.67
22209-0001-00A	100G IB EDR HBC CARD CX556A - ConnectX-5 (ConnectX-5 Ex VPI Adapter Card EDR IB and 100GbE Dual-port QSFP28 PCIe4.0 x16 Tall Bracket ROHS R6)	15	951.00	14,265.00	0%	14,265.00
22202-0001-00A	Raid Controller SAS Megaraid 9361-8i (up to 8 SATA or SAS drives via direct connection or up to 240 drives with SAS expander	10	931.61	9,316.10	0%	9,316.10
22506-0003-00A	SSD 480GB 2.5" 6Gb SATA3	4	281.64	1,126.56	0%	1,126.56
22505-0001-00A	HDD 300GB 2.5" 10KRPM 6Gb SAS	4	55.00	220.00	0%	220.00
22505-0001-00A	HDD 300GB 2.5" 10.5KRPM 6Gb SAS	2	55.00	110.00	0%	110.00
22505-0002-00A	HDD 146GB 2.5" 10KRPM 6Gb SAS	6	40.07	240.42	0%	240.42
22505-0002-00A	HDD 147GB 2.5" 15KRPM 6Gb SAS	4	42.10	168.40	0%	168.40
22501-0003-00A	NVMe SSD 1.6TB, HGST, SN200, 2.5" SFF	8	1,306.62	10,452.96	0%	10,452.96
22501-0003-00A	SSF	14	1,306.62	18,292.68	0%	18,292.68
22501-0003-00A	NVMe SSD 1.6TB, Intel, DCP4600 2.5" SSF	56	1,306.62	73,170.73	0%	73,170.73
31020-0008-00	Board PCI-e Switching GEN3 x8 4Port for HHS2112F,HHS3124F	20	5,038.95	100,779.00	50%	50,389.50
10401-0001-00A	MSB7890ES2F 100G IB SWITCH By Mellanox (36-port Non-blocking Externally-managed EDR)	1	18,423.08	18,423.08	0%	18,423.08
22809-0001-00A	MCP1600-E002 IB EDR Cable	30	136.76	4,102.79	0%	4,102.79
58090-0001-00A	42U Rack Cabinet	1	776.59	776.59	0%	776.59
60304-0002-00A	Gluesys Anystore Enterprise, AnyManager SW (NAS O/S, RAID 0/1/10/5/6, Protocol NFS/CIFS/AFP, NVMe Option, UI Mgt. etc.)	10	20,000.00	200,000.00	50%	100,000.00
				Hardware & Software Subtotal		**601,537.07**
Support & Maintenance						
A0103-0001-00A	Premium Package 3-Year Support & Maintenance	10	18,046.00	180,460.00	50%	90,230.00
				Support & Maintenance Subtotal		**90,230.00**
	SPC-1 Total System Price					**691,767.07**
	SPC-1 IOPS™					2,410,271
	SPC-1 Price-Performance™ ($/SPC-1 KIOPS™)					**287.01**
	SPC-1 ASU Capacity (GB)					46,789
	SPC-1 ASU Price ($/GB)					**14.79**

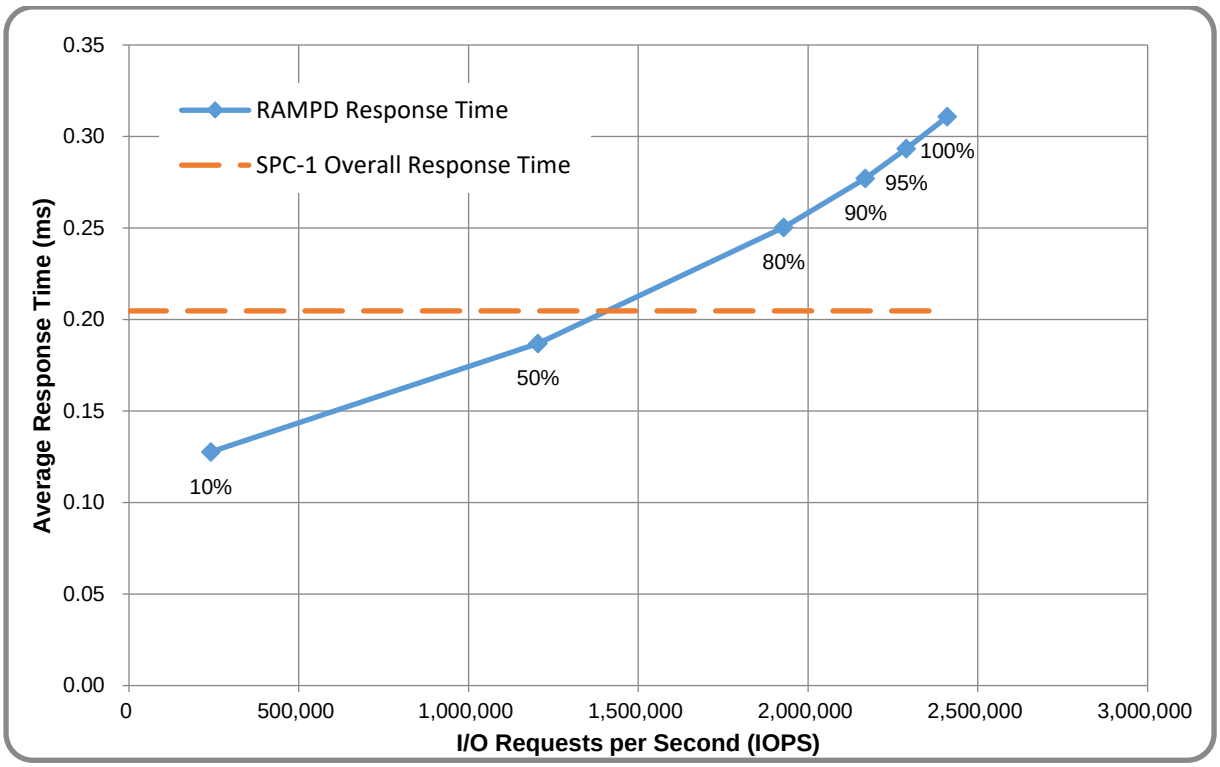
Third-Party Reseller: TTA is the sponsor of tis result but does not directly sell the products and components of the Priced Storage Configuration (PSC). The above reflects the pricing quoted by the vendor and third-party reseller Taejin Infotech Co., Ltd. See Appendix B of the Full Disclosure Report for a copy of the third-party reseller's quotation.

Discount Details: The discounts shown are based on the storage capacity purchased and are generally available.

Warranty: The 3-year maintenance and support included in the above pricing meets or exceeds a 24x7 coverage with a 4-hour response time.

Availability Date: Currently available.

Response Time and Throughput Graph



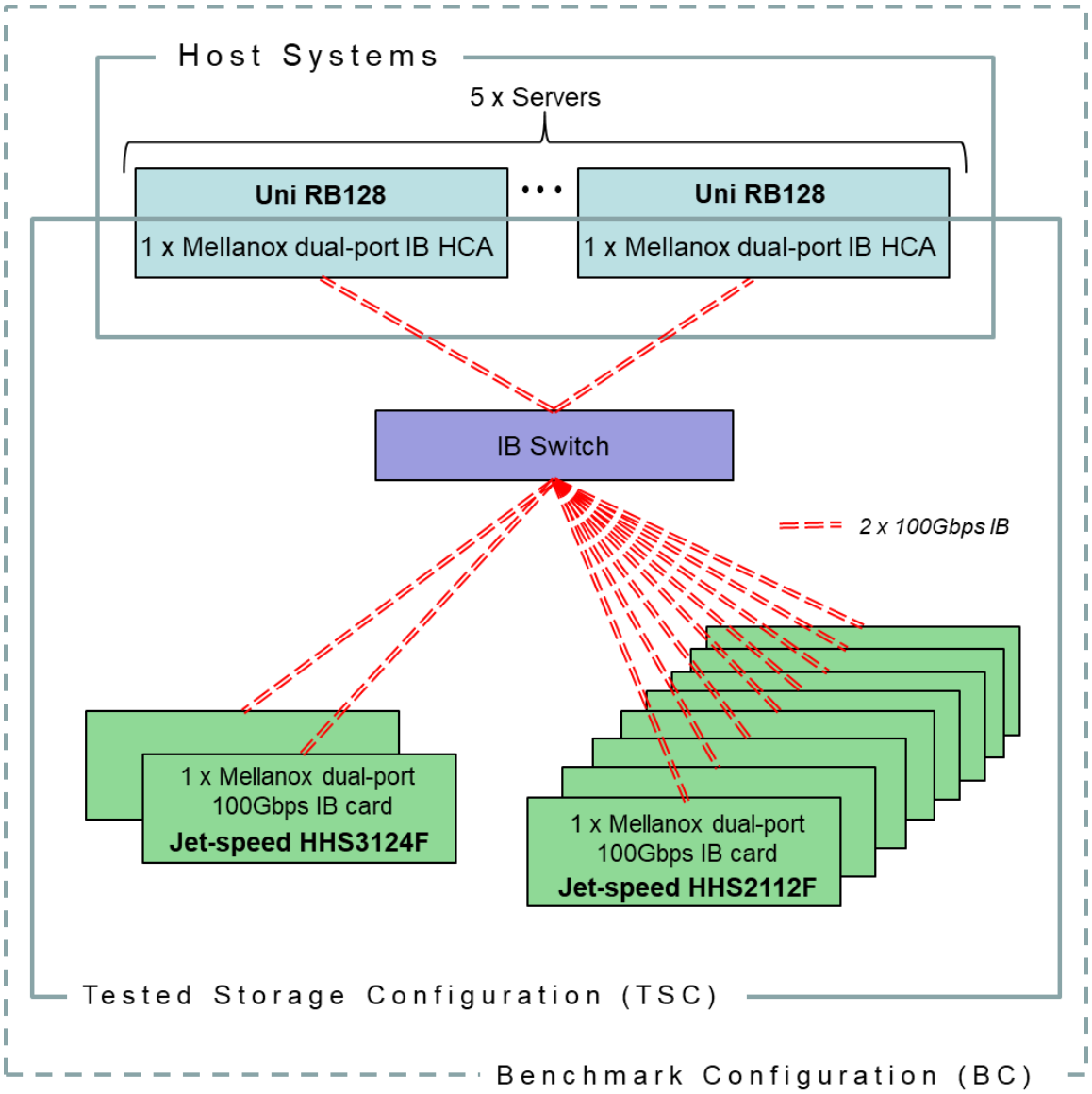
Contact Information	
Test Sponsor Primary Contact	TTA – http://tta.or.kr/eng/index.jsp Hyojin (Kailynne) Kim – hjkim16@tta.or.kr
SPC Auditor	InfoSizing – www.sizing.com Francois Raab – francois@sizing.com

Revision Information	
SPC Benchmark 1™ Revision	V3.8.0
SPC-1 Workload Generator Revision	0xe28e08v3.0.2
Publication Revision History	First Edition

CONFIGURATION INFORMATION

Benchmark Configuration and Tested Storage Configuration

The following diagram illustrates the Benchmark Configuration (BC), including the Tested Storage Configuration (TSC) and the Host System(s).



Storage Network Configuration

The Tested Storage Configuration (TSC) involved ten storage nodes of two types (two HHS3124F nodes and eight HHS2112F nodes), driven by five Uni RB128 host systems. Each host had two connections to a Mellanox SB7890 InfiniBand (IB) Switch. Each of the ten Jet-speed storage nodes had two connections to the Mellanox SB7890 IB Switch. All connections operated at 100Gbps.

Host System and Tested Storage Configuration Components

The following table lists the components of the Host System(s) and the Tested Storage Configuration (TSC).

Host Systems
5 x Uni RB128 x86 Servers 2 x Intel Xeon E5-2699 v4 2.20 GHz 22-core 12 x 32GB DDR4 Memory CentOS 7.4 (64-bit)
Priced Storage Configuration
5 x Mellanox CX556A-ConnectX-5 2-port 100Gbps IB cards (1 per host)
2 x Jet-speed HHS3124F storage nodes, each with: 2 x Intel Xeon CPU E5-2630 v4 2.20GHz 10-core (node 1) 2 x Intel Xeon CPU E5-2640 v4 2.40GHz 10-core (node 2) 4 x 16GB DDR4 Memory 1 x Mellanox 2-port 100Gbps IB card 1 x SAS/SATA MegaRAID controller 2 x PCIe Gen3 switch cards 2 x 480GB SATA 6Gbps SSD (System) 8 x 1600GB NVMe SSD 2.5' SFF (node 1) 6 x 1600GB NVMe SSD 2.5' SFF (node 2) 8 x Jet-speed HHS2112F storage nodes, each with: 2 x Intel Xeon CPU E5-2630 v3 2.40GHz 8-core 4 x 16GB DDR4 Memory 2 x PCIe Gen3 switch cards 1 x Mellanox 2-port 100Gbps IB card 1 x SAS/SATA MegaRAID controller 2 x 300GB SAS 10Krpm HDD (System, nodes 3,4,8) 2 x 146GB SAS 10Krpm HDD (System, nodes 5,6,7) 2 x 147GB SAS 15Krpm HDD (System, nodes 9,10) 8 x 1600GB NVMe SSD 2.5' SFF Mellanox SB7890 100 Gbps 36-port IB Switch

Differences Between Tested and Priced Storage Configurations

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

Component Changes in Revised Full Disclosure Report

The following table outlines component changes that were made in revisions to this Full Disclosure Report.

Original Component	Revised Component	Description of Change
n/a	n/a	Initial submission

Benchmark Configuration Creation Process

Customer Tuning Parameters and Options

All the customer tuning parameters and options that have been altered from their default values for this benchmark are included in Appendix C and in the Supporting Files (see Appendix A).

Tested Storage Configuration Creation

A detailed description of how the logical representation of the TSC was created is included in Appendix D and in the Supporting Files (see Appendix A).

Tested Storage Configuration Inventory

An inventory of the components in the TSC, as seen by the Benchmark Configuration, is included in Appendix E and in the Supporting Files (see Appendix A).

Workload Generator Storage Configuration

The SPC-1 Workload Generator storage configuration commands and parameters used to invoke the execution of the tests are included in Appendix F and in the Supporting Files (see Appendix A).

Logical Volume Capacity and ASU Mapping

The following table details the capacity of each ASU and how they are mapped to logical volumes (LV).

	LV per ASU	LV Capacity	Used per LV	Total per ASU	% ASU Capacity
ASU-1	9	2,339.49	2,339.49	21,055.41	45.00%
ASU-2	9	2,339.49	2,339.49	21,055.41	45.00%
ASU-3	1	4,678.33	4,678.33	4,678.33	10.00%
			SPC-1 ASU Capacity	**46,789.15**	

Physical Storage Capacity and Utilization

The following table details the Physical Capacity of the storage devices and the Physical Capacity Utilization (percentage of Total Physical Capacity used) in support of hosting the ASUs.

Devices	Count	Physical Capacity	Total Capacity
1,600GB NVMe SSD	78	1,600.00	124,800.00
480GB SSD (system)	4	479.60	1,918.40
300GB HDD (system)	6	299.00	1,794.00
146GB HDD (system)	6	146.30	877.80
147GB HDD (system)	4	146.46	585.84
Total Physical Storage Capacity			**129,976.04**
Physical Capacity Utilization			**36.00%**

Data Protection

The data protection level used for all logical volumes was **Protected 1**, which was accomplished on the NVMe storage nodes by combining pairs of nearby NVMe SSDs in RAID 1 volumes and creating a RAID 0 volume over all of the RAID 1 volumes.

BENCHMARK EXECUTION RESULTS

This portion of the Full Disclosure Report documents the results of the various SPC-1 Tests, Test Phases, and Test Runs.

Benchmark Execution Overview

Workload Generator Input Parameters

The SPC-1 Workload Generator commands and input parameters for the Test Phases are presented in the Supporting Files (see Appendix A).

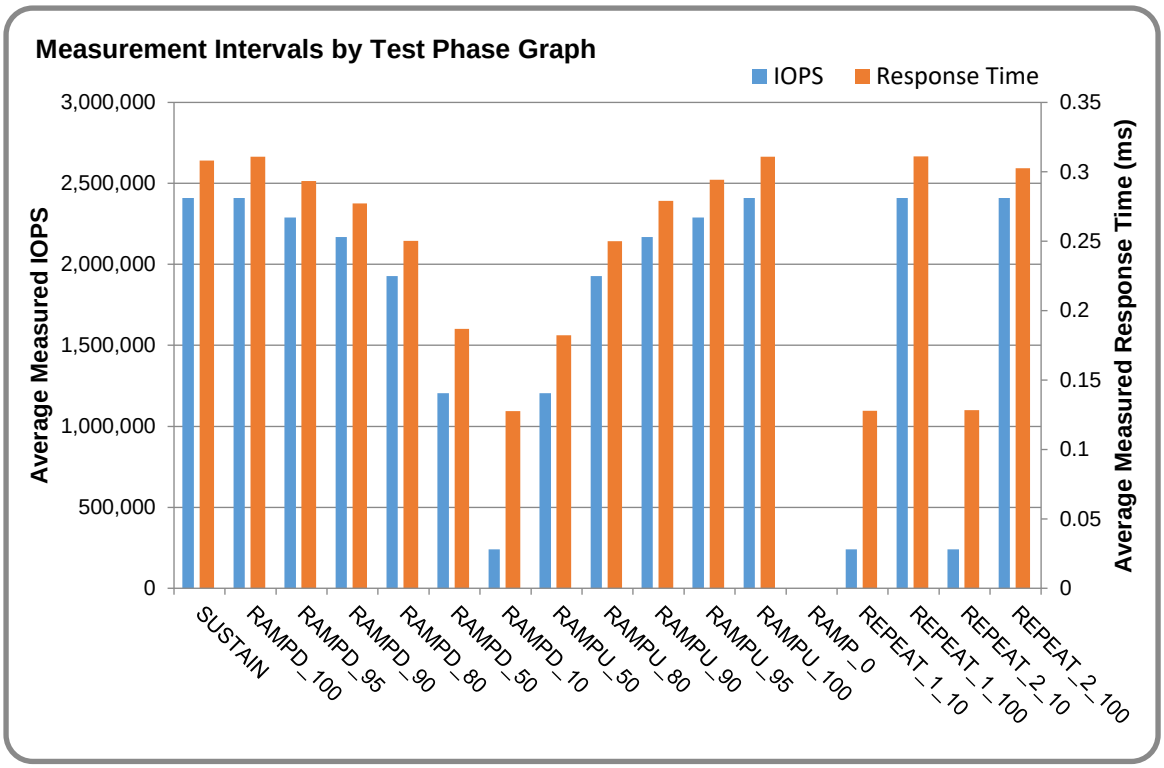
Primary Metrics Test Phases

The benchmark execution consists of the Primary Metrics Test Phases, including the Test Phases SUSTAIN, RAMPD_100 to RAMPD_10, RAMPU_50 to RAMPU_100, RAMP_0, REPEAT_1 and REPEAT_2.

Each Test Phase starts with a transition period followed by a Measurement Interval.

Measurement Intervals by Test Phase Graph

The following graph presents the average IOPS and the average Response Times measured over the Measurement Interval (MI) of each Test Phase.



Exception and Waiver

None.

SUSTAIN Test Phase

SUSTAIN – Results File

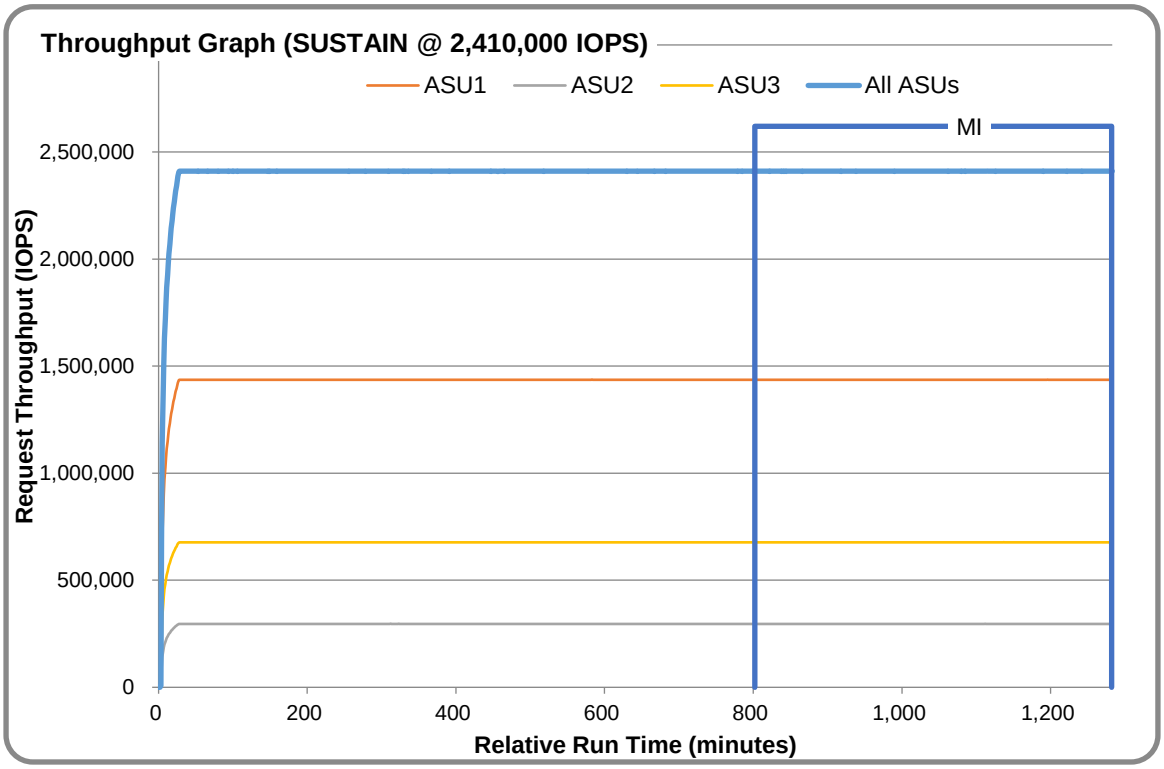
The results file generated during the execution of the SUSTAIN Test Phase is included in the Supporting Files (see Appendix A) as follows:

- **SPC1_METRICS_0_Raw_Results.xlsx**

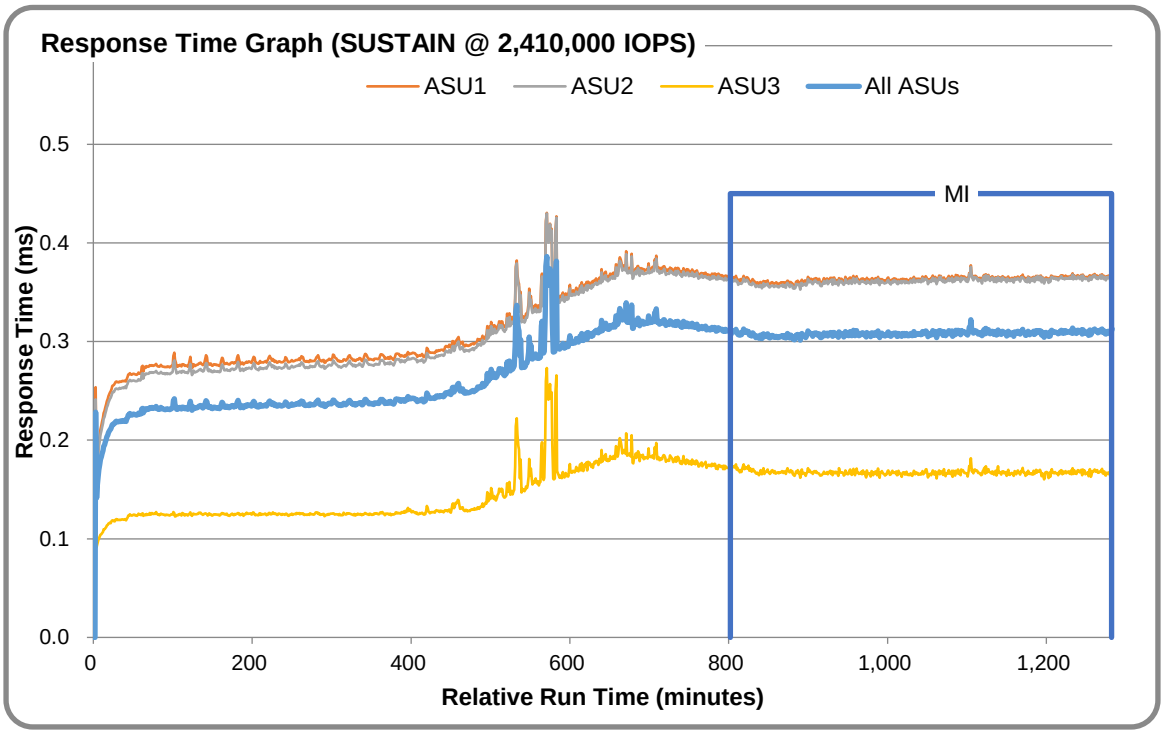
SUSTAIN – Execution Times

Interval	Start Time	End Time	Duration
Transition Period	14-Nov-18 08:15:28	14-Nov-18 21:35:25	13:19:56
Measurement Interval	14-Nov-18 21:35:25	15-Nov-18 05:35:26	8:00:01

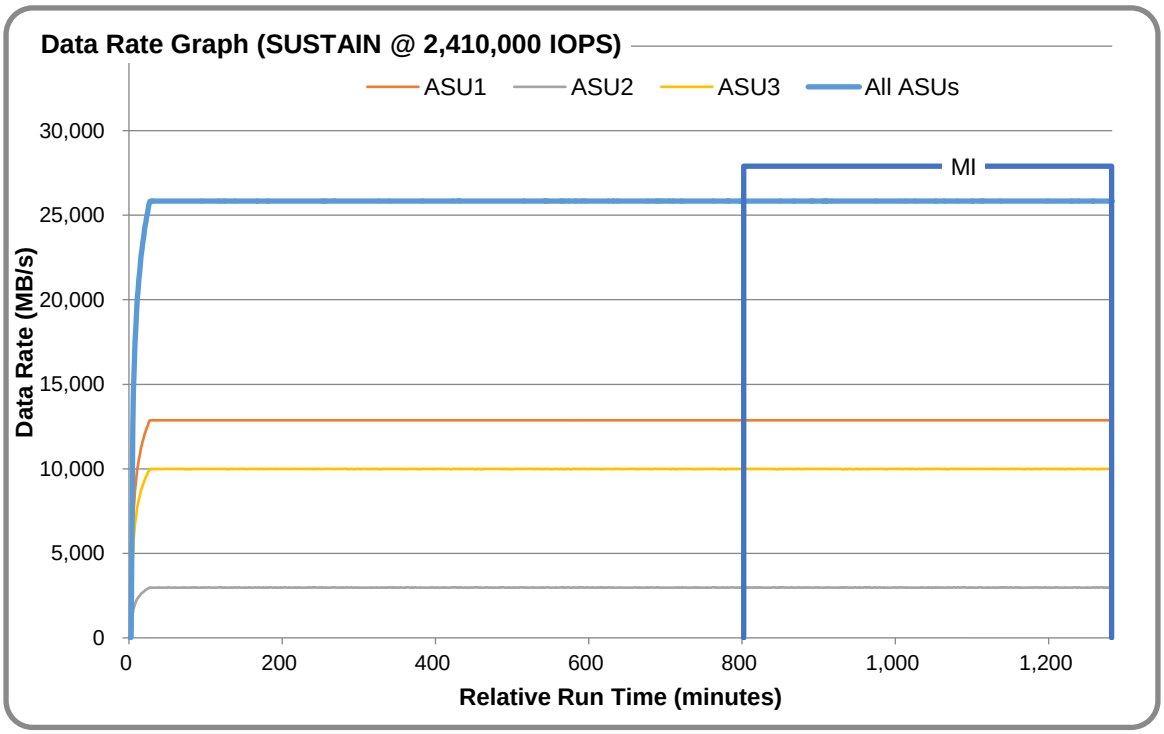
SUSTAIN – Throughput Graph



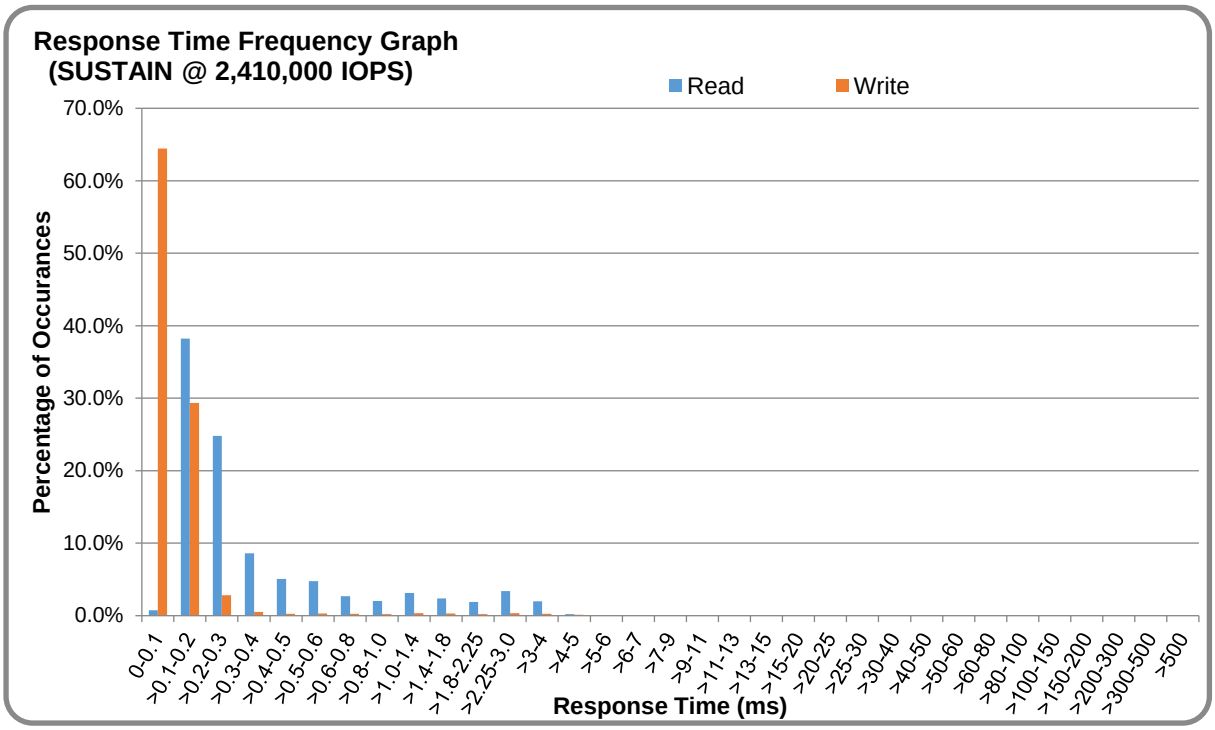
SUSTAIN – Response Time Graph



SUSTAIN – Data Rate Graph



SUSTAIN – Response Time Frequency Graph



SUSTAIN – Intensity Multiplier

The following table lists the targeted intensity multiplier (Defined), the measured intensity multiplier (Measured) for each I/O STREAM, its coefficient of variation (Variation) and the percentage of difference (Difference) between Target and Measured.

	ASU1-1	ASU1-2	ASU1-3	ASU1-4	ASU2-1	ASU2-2	ASU2-3	ASU3-1
Defined	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
Measured	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
Variation	0.0004	0.0001	0.0003	0.0002	0.0006	0.0003	0.0004	0.0001
Difference	0.006%	0.003%	0.004%	0.002%	0.001%	0.004%	0.003%	0.002%

RAMPD_100 Test Phase

RAMPD_100 – Results File

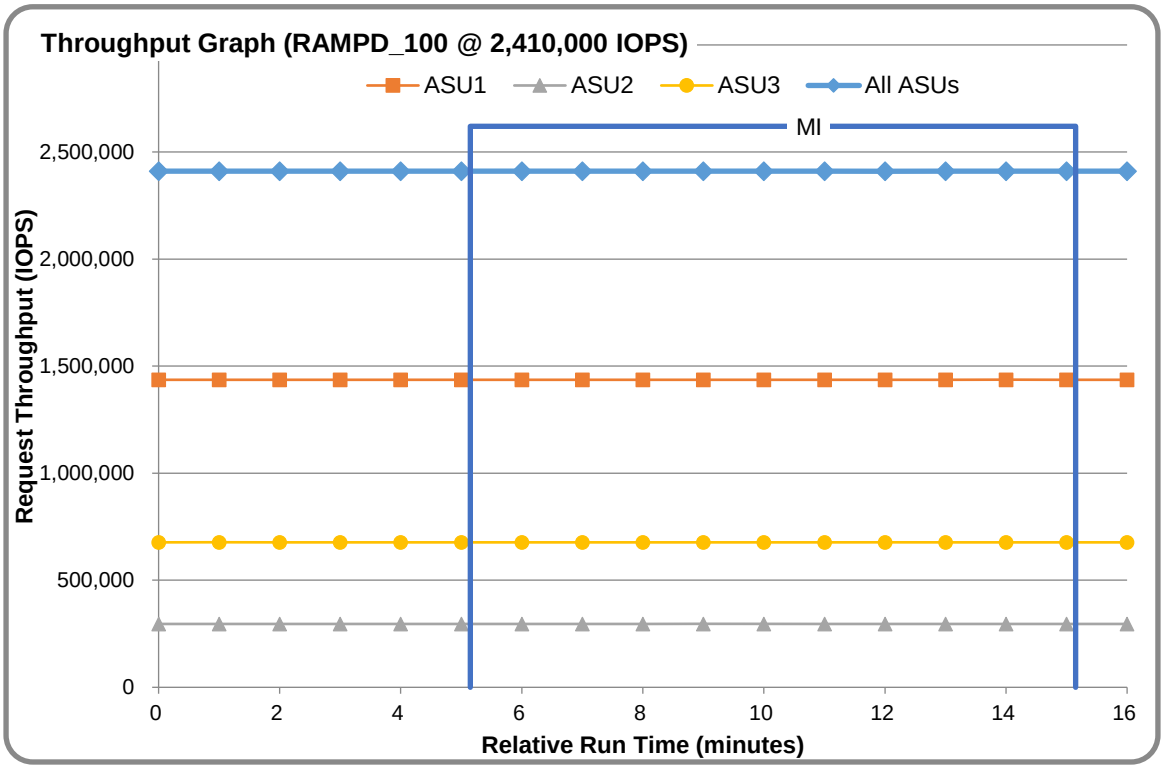
The results file generated during the execution of the RAMPD_100 Test Phase is included in the Supporting Files (see Appendix A) as follows:

- **SPC1_METRICS_0_Raw_Results.xlsx**

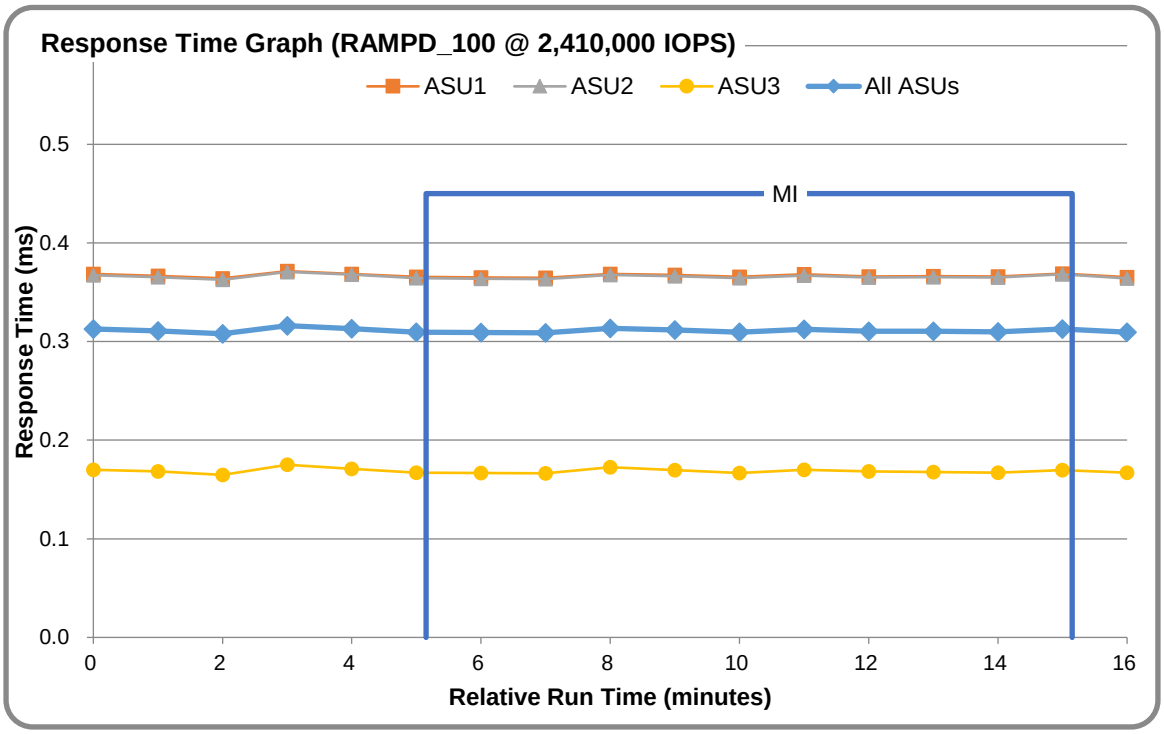
RAMPD_100 – Execution Times

Interval	Start Time	End Time	Duration
Transition Period	15-Nov-18 05:36:25	15-Nov-18 05:41:26	0:05:01
Measurement Interval	15-Nov-18 05:41:26	15-Nov-18 05:51:26	0:10:00

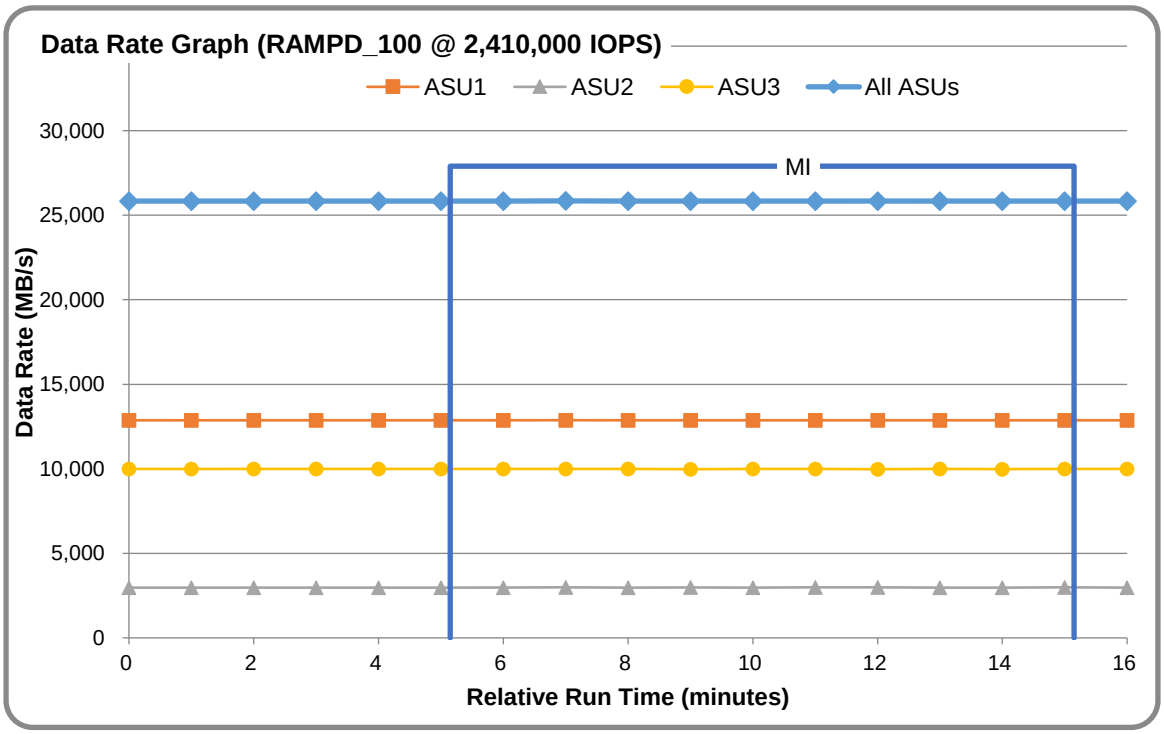
RAMPD_100 – Throughput Graph



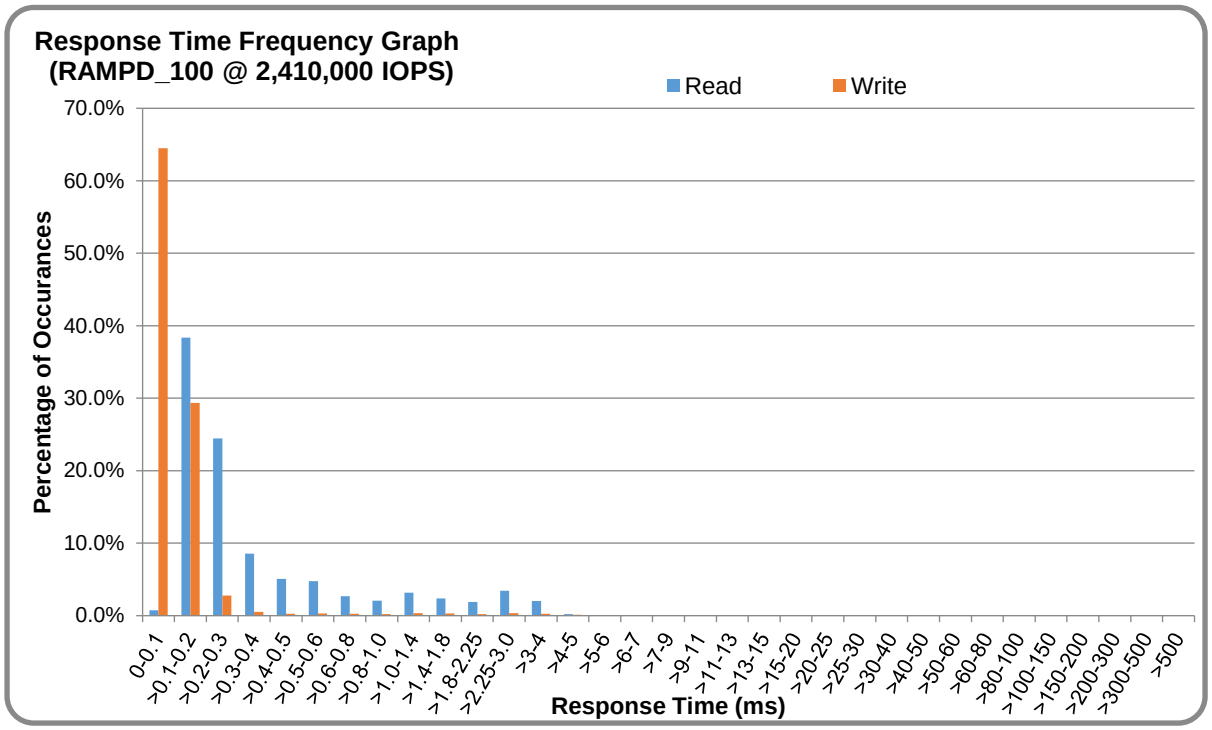
RAMPD_100 – Response Time Graph



RAMPD_100 – Data Rate Graph



RAMPD_100 – Response Time Frequency Graph



RAMPD_100 – Intensity Multiplier

The following table lists the targeted intensity multiplier (Defined), the measured intensity multiplier (Measured) for each I/O STREAM, its coefficient of variation (Variation) and the percentage of difference (Difference) between Target and Measured.

	ASU1-1	ASU1-2	ASU1-3	ASU1-4	ASU2-1	ASU2-2	ASU2-3	ASU3-1
Defined	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
Measured	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
Variation	0.0003	0.0001	0.0003	0.0001	0.0006	0.0003	0.0004	0.0001
Difference	0.008%	0.001%	0.005%	0.003%	0.046%	0.000%	0.010%	0.000%

RAMPD_100 – I/O Request Summary

I/O Requests Completed in the Measurement Interval	1,446,176,219
I/O Requests Completed with Response Time <= 30 ms	1,446,170,555
I/O Requests Completed with Response Time > 30 ms	5,664

Response Time Ramp Test

Response Time Ramp Test – Results File

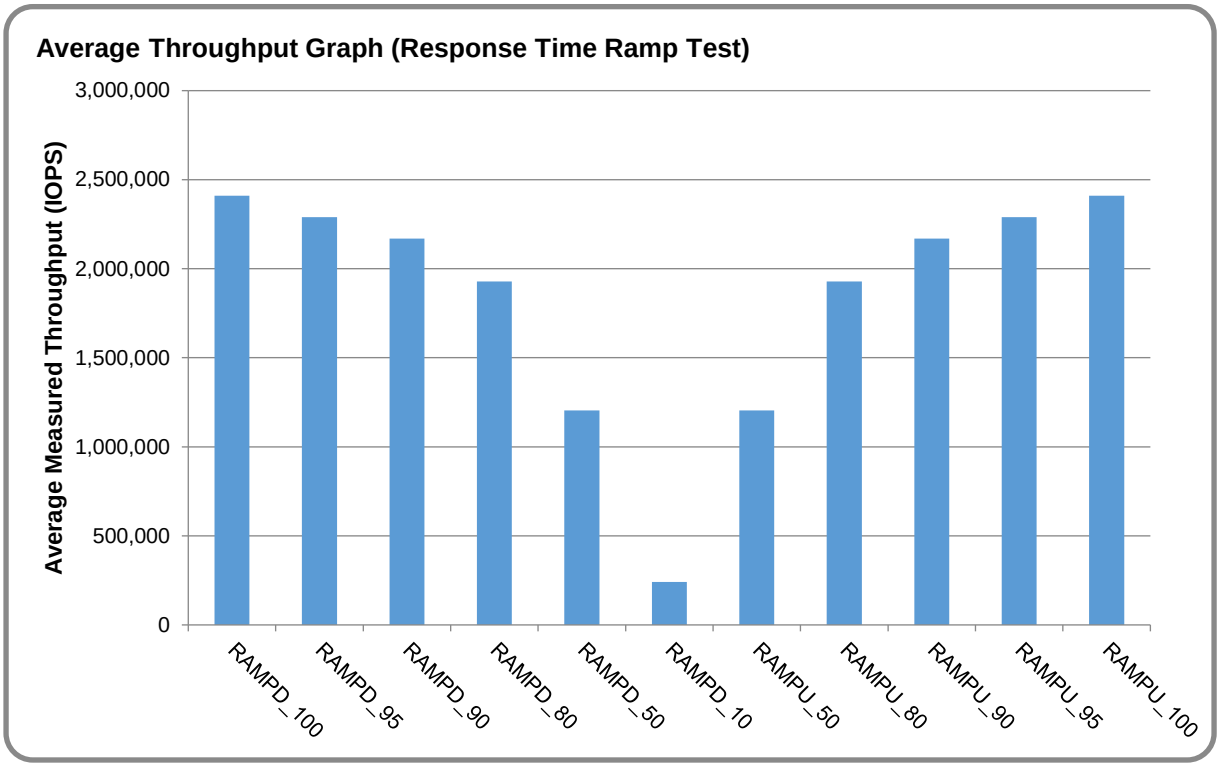
The results file generated during the execution of the Response Time Ramp Test is included in the Supporting Files (see Appendix A) as follows:

- **SPC1_METRICS_0_Raw_Results.xlsx**

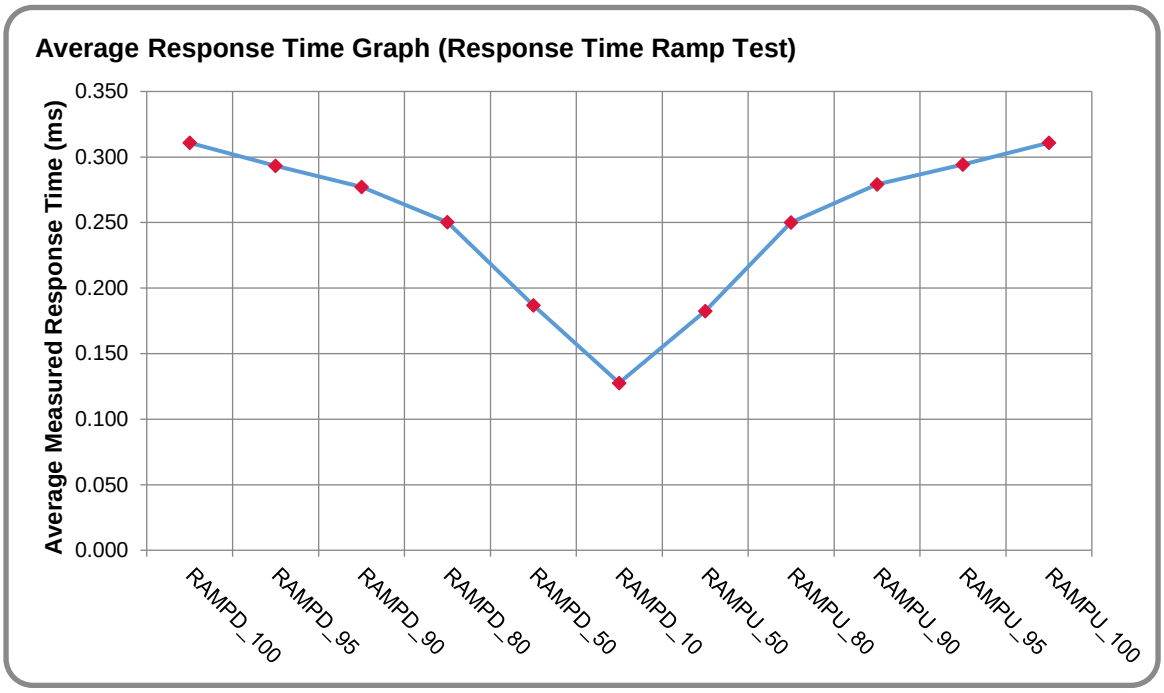
Response Time Ramp Test – Phases

The Response Time Ramp Test is comprised of 11 Test Phases, including six Ramp-Down Phases (executed at 100%, 95%, 90%, 80%, 50%, and 10% of the Business Scaling Unit) and five Ramp-Up Phases (executed at 50%, 80%, 90%, 95%, and 100% of the Business Scaling Unit).

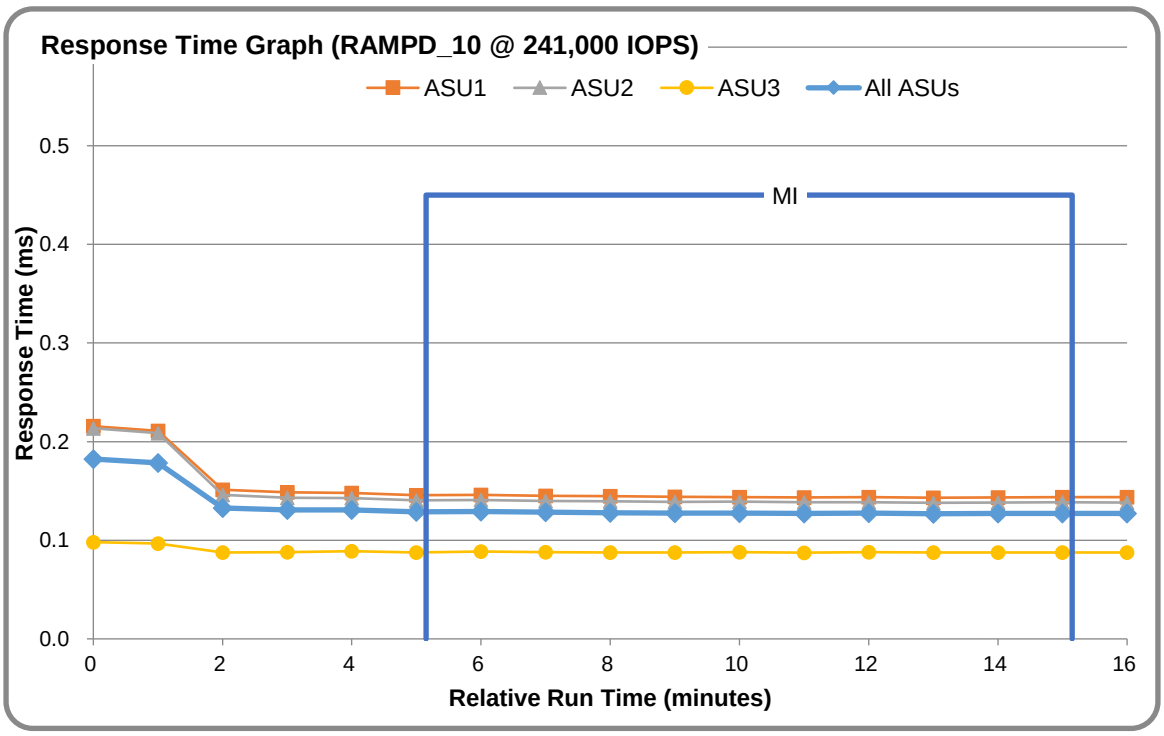
Response Time Ramp Test – Average Throughput Graph



Response Time Ramp Test – Average Response Time Graph



Response Time Ramp Test – RAMPD_10 Response Time Graph



Repeatability Test

Repeatability Test Results File

The results file generated during the execution of the Repeatability Test is included in the Supporting Files (see Appendix A) as follows:

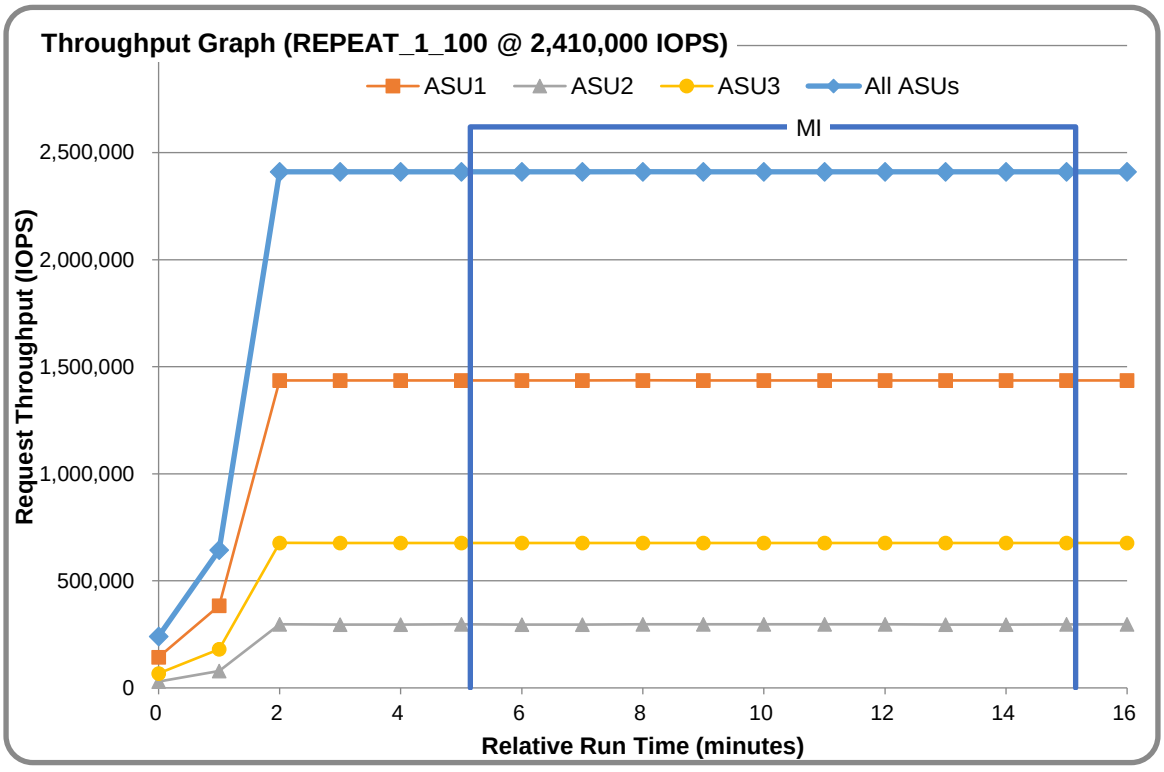
- **SPC1_METRICS_0_Raw_Results.xlsx**

Repeatability Test Results

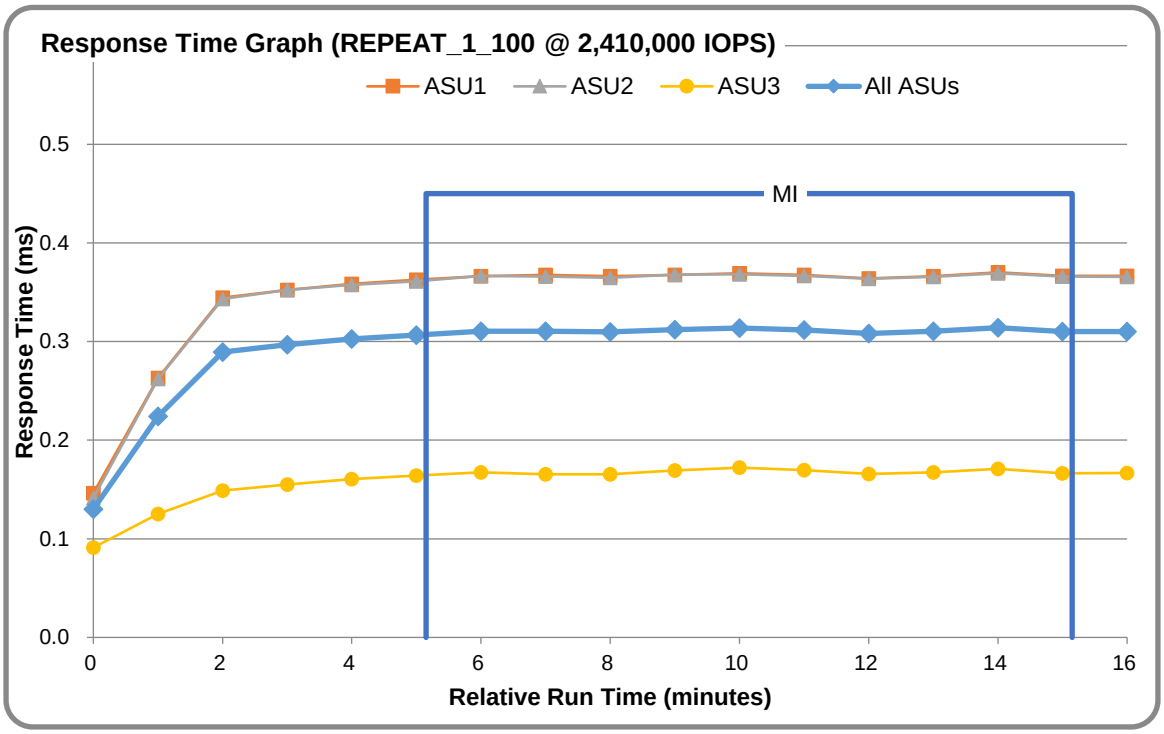
The throughput measurements for the Response Time Ramp Test (RAMPD) and the Repeatability Test Phases (REPEAT_1 and REPEAT_2) are listed in the tables below.

Test Phase	100% IOPS	10% IOPS
RAMPD	2,410,271.3	241,011.0
REPEAT_1	2,410,246.6	241,014.7
REPEAT_2	2,410,270.6	241,042.6

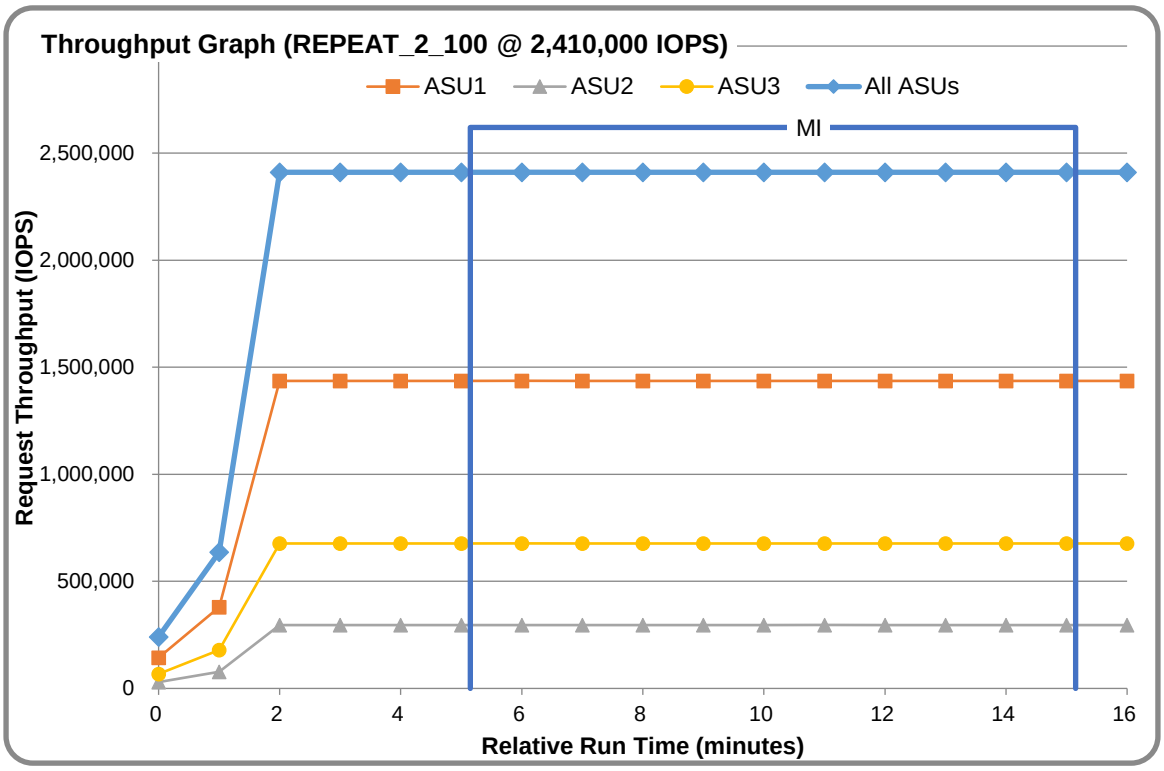
REPEAT_1_100 – Throughput Graph



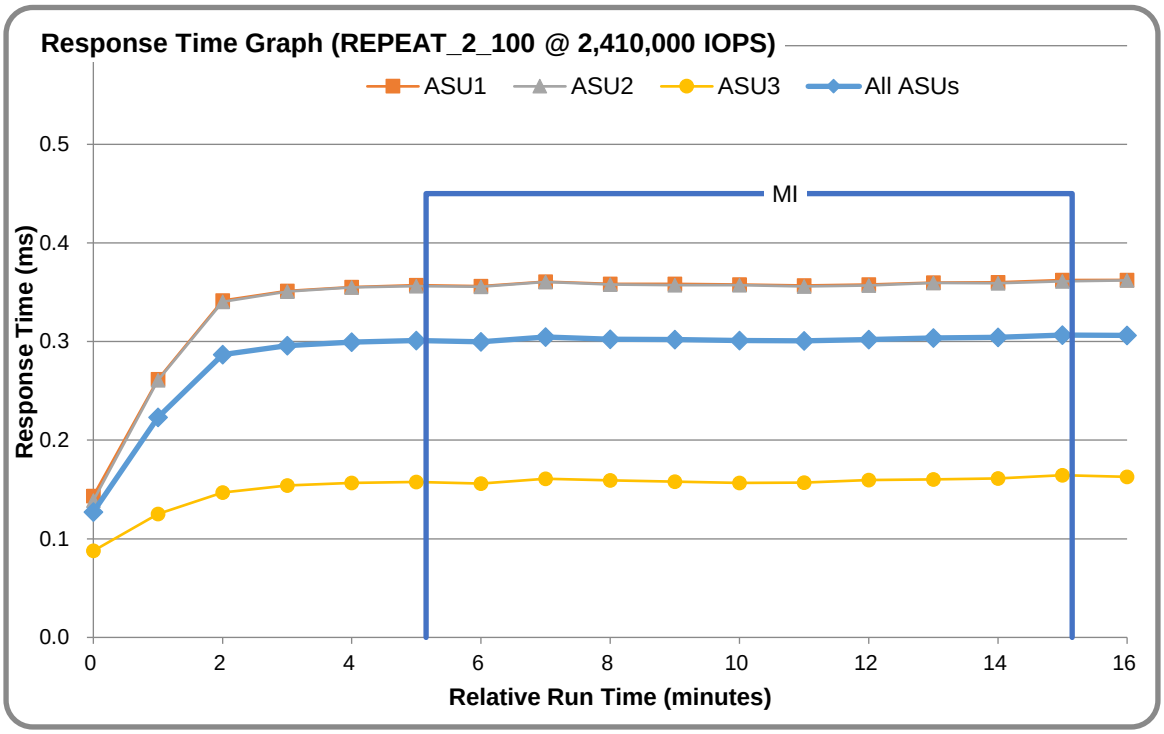
REPEAT_1_100 – Response Time Graph



REPEAT_2_100 – Throughput Graph



REPEAT_2_100 – Response Time Graph



Repeatability Test – Intensity Multiplier

The following tables lists the targeted intensity multiplier (Defined), the measured intensity multiplier (Measured) for each I/O STREAM, its coefficient of variation (Variation) and the percent of difference (Difference) between Target and Measured.

REPEAT_1_100 Test Phase

	ASU1-1	ASU1-2	ASU1-3	ASU1-4	ASU2-1	ASU2-2	ASU2-3	ASU3-1
Defined	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
Measured	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
Variation	0.0004	0.0001	0.0003	0.0001	0.0004	0.0004	0.0004	0.0001
Difference	0.019%	0.000%	0.006%	0.005%	0.007%	0.000%	0.019%	0.002%

REPEAT_2_100 Test Phase

	ASU1-1	ASU1-2	ASU1-3	ASU1-4	ASU2-1	ASU2-2	ASU2-3	ASU3-1
Defined	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
Measured	0.0350	0.2810	0.0700	0.2100	0.0180	0.0700	0.0350	0.2810
Variation	0.0004	0.0002	0.0005	0.0002	0.0004	0.0004	0.0006	0.0002
Difference	0.012%	0.008%	0.003%	0.002%	0.012%	0.027%	0.015%	0.003%

Space Optimization Reporting

Description of Techniques Used

No space optimization was used for this SPC-1 result.

Physical Free Space Measurements

The following table lists the Physical Free Space as measured at each of the required points during test execution. If space optimization techniques were not used, "NA" is reported.

Physical Free Space Measurement	Free Space (GB)
After Logical Volume Creation	NA
After ASU Pre-Fill	NA
After Repeatability Test Phase	NA

Space Optimization Metrics

The following table lists the required space optimization metrics. If space optimization techniques were not used, "NA" is reported.

Space Optimization Metric	Value
SPC-1 Space Optimization Ratio	NA
SPC-1 Space Effectiveness Ratio	NA

Data Persistence Test

Data Persistence Test Results File

The results files generated during the execution of the Data Persistence Test is included in the Supporting Files (see Appendix A) as follows:

- **SPC1_PERSIST_1_0_Raw_Results.xlsx**
- **SPC1_PERSIST_2_0_Raw_Results.xlsx**

Data Persistence Test Execution

The Data Persistence Test was executed using the following sequence of steps:

- The PERSIST_1_0 Test Phase was executed to completion.
- The Benchmark Configuration was taken through an orderly shutdown process and powered off.
- The Benchmark Configuration was powered on and taken through an orderly startup process.
- The PERSIST_2_0 Test Phase was executed to completion.

Data Persistence Test Results

Data Persistence Test Phase: Persist1	
Total Number of Logical Blocks Written	478,651,585
Total Number of Logical Blocks Verified	233,295,120
Total Number of Logical Blocks Overwritten	245,356,465
Total Number of Logical Blocks that Failed Verification	0
Time Duration for Writing Test Logical Blocks (sec.)	601
Size in bytes of each Logical Block	8,192
Number of Failed I/O Requests in the process of the Test	0

Committed Data Persistence Implementation

The persistency of committed data is implemented at the disk level, where data loss is prevented through the use of RAID 1 arrays. At the controller level, the cache is set-up in write-through mode and needs not to be protected to ensure persistence of committed data.

APPENDIX A: SUPPORTING FILES

The following table details the content of the Supporting Files provided as part of this Full Disclosure Report.

File Name	Description	Location
/SPC1_RESULTS	**Data reduction worksheets**	**root**
SPC1_INIT_0_Raw_Results.xlsx	Raw results for INIT Test Phase	/SPC1_RESULTS
SPC1_METRICS_0_Quick_Look.xlsx	Quick Look Test Run Overview	/SPC1_RESULTS
SPC1_METRICS_0_Raw_Results.xlsx	Raw results for Primary Metrics Test	/SPC1_RESULTS
SPC1_METRICS_0_Summary_Results.xlsx	Primary Metrics Summary	/SPC1_RESULTS
SPC1_PERSIST_1_0_Raw_Results.xlsx	Raw results for PERSIST1 Test Phase	/SPC1_RESULTS
SPC1_PERSIST_2_0_Raw_Results.xlsx	Raw results for PERSIST2 Test Phase	/SPC1_RESULTS
SPC1_Run_Set_Overview.xlsx	Run Set Overview Worksheet	/SPC1_RESULTS
SPC1_VERIFY_0_Raw_Results.xlsx	Raw results for first VERIFY Test Phase	/SPC1_RESULTS
SPC1_VERIFY_1_Raw_Results.xlsx	Raw results for second VERIFY Test Phase	/SPC1_RESULTS
/C_Tuning	**Tuning parameters and options**	**root**
rc.local	Set tuning parameters	/C_Tuning
/D_Creation	**Storage configuration creation**	**root**
create_nvmet_md.sh	Create the NVMe over Fabric setup	/D_Creation
lvcreate.sh	Create the Logical Volumes	/D_Creation
mdcreate.sh	Create RAID volumes	/D_Creation
mdfrozen.sh	Stop RAID re-sync activity	/D_Creation
parted.sh	Create partitions on the NVMe devices	/D_Creation
rm_parted.sh	Remove pre-existing partitions	/D_Creation
/E_Inventory	**Configuration inventory**	**root**
inventory_start.out	Storage inventory before INIT	/E_Inventory
inventory_end.out	Storage inventory after restart	/E_Inventory
/F_Generator	**Workload generator**	**root**
hst1_linear.asu	Defining LUNs hosting the ASUs	/F_generator
5host.HST	Host configuration file	/F_generator
spc1.sh	Executing all test phases	/F_generator

APPENDIX B: THIRD PARTY QUOTATION

TAEJIN INFOTECH

Quotation

Date : 10-Nov-2018 NO. TJQ2018-1110001
Attention : C.C
Project : For SPC-1 Test with HHS3124F & HHS2112F
Price Availability : 12-Dec-2018

Company : Taejin Infotech Co., Ltd.
Sign Name : Hyukjong Ahn
Position : CTO Total Amount : 691,767.07 (USD)

NO.	Item	Description	Q'ty	Unit Price	Amount	DC %	Final Amount	Remark
1	90102-0001-00A	HHS2112F All NVMe Storage	8	50,379.73	403,037.82	50%	**201,518.91**	
		Detail Specification						
		2U / 12 disk bays Dual Intel Xeon E5-2600v3/v4 Family, 16x DIMM Slots, 800W redundant PSU 80Plus platinum						
2	90102-0002-00A	HHS3124F All NVMe Storage	2	91,777.78	183,555.56	50%	**91,777.78**	
		Detail Specification						
		3U / 24 disk bays Dual Intel Xeon E5-2600v3/v4 Family, 16x DIMM Slots, 1000W redundant PSU 80Plus platinum						
3	20204-0001-00A	DDR4 16GB PC4-17000 ECC/REG	28	179.64	5,029.89	0%	**5,029.89**	Resale (Samsung)
		Detail Specification						
		16GB PC4-17000 DDR Rdimm Ecc Reg RX8 CL17 1.2V						
4	20204-0002-00A	DDR4 16GB PC4-19200 ECC/REG	12	179.64	2,155.67	0%	**2,155.67**	Resale (Samsung)
		Detail Specification						
		16GB PC4-19200 DDR Rdimm Ecc Reg RX8 CL17 1.2V						
5	22209-0001-00A	100G IB EDR HCA CARD CX556A - ConnectX-5	15	951.00	14,265.00	0%	**14,265.00**	Resale (Mellanox)
		Detail Specification						
		ConnectX-5 Ex VPI Adapter Card EDR IB and 100GbE Dual-port QSFP28 PCIe4.0 x16 Tall Bracket ROHS R6 Intelligent RDMA-enabled network adapter card with advanced application offload capabilities for High-Performance Computing, Web2.0, Cloud and Storage platforms						
6	22202-0001-00A	Raid Controller SAS/SATA Megaraid 9361-8i	10	931.61	9,316.10	0%	**9,316.10**	Resale (Broadcom)
		Detail Specification						
		Connect up to 8 SATA or SAS drives via direct connection inside the box or up to 240 drives leveraging SAS expander technology Fit into rack-mounted servers with low-profile form factor and side-mounted SAS connectors Support critical, high-bandwidth applications with PCIe® 3.0 connectivity Balance protection and performance for critical applications with RAID levels 0, 1, 5, 6, 10, 50, and 60						
7	22506-0003-00A	SSD 480GB SATA3 2.5"	4	281.64	1,126.56	0%	**1,126.56**	Resale (Samsung)
		Detail Specification						
		SATA 6Gb/s SSD SAMSUNG MZ7KM480HAHP-00005 480GB SM863 480GB SATA-6GBPS 2.5INCH MLC INTERNAL SOLID STATE DRIVE. BRAND NEW WITH STANDARD MFG WARRANTY CAPACITY : 480 GB INTERFACE : SATA-6GBPS DRIVE DIMENSIONS : 2.5 INCH NAND FLASH MEMORY TYPE : MULTI-LEVEL CELL (MLC) TRIM SUPPORT : YES MTBF : 2,000,000 HOURS POWER CONSUMPTION : 2.2 WATT (READ) 2.8 WATT (WRITE)						
8	22505-0001-00A	HDD 300GB SAS 2.5" 10KRPM	4	55.00	220.00	0%	**220.00**	Resale (HGST)
		Detail Specification						
		Hitachi GST Ultrastar C10K300 HUC103030CSS600 300GB 10000 RPM 64MB Cache SAS 6Gb/s 2.5" Enterprise Internal Hard Drive Bare Drive • 10000 RPM • 64MB Cache • SAS 6Gb/s • 3 years limited						
9	22505-0002-00A	HDD 146GB SAS 2.5" 10KRPM	6	40.07	240.42	0%	**240.42**	Resale (Seagate)
		Detail Specification						
		Seagate Savvio 10K.3 ST9146803SS 146GB 10000 RPM 16MB Cache SAS 6Gb/s 2.5" Internal Enterprise Hard Drive Bare Drive • 10000 RPM • 16MB Cache • SAS 6Gb/s • Ideal for Servers						

10	22505-0003-00A	HDD 300GB SAS 2.5" 10.5KRPM	2	55.00	110.00	0%	**110.00**	Resale (Toshiba)
		Detail Specification						
		Toshiba AL15SEB 300GB 10500 RPM 64MB Cache SAS 6Gb/s 2.5" Internal Enterprise Hard Drive Bare Drive • 10500 RPM • 64MB Cache • SAS 6Gb/s • Ideal for Servers						
11	22505-0004-00A	HDD 147GB SAS 2.5" 15KRPM	4	42.10	168.40	0%	**168.40**	Resale (Hitachi)
		Detail Specification						
		Hitachi GST Ultrastar C15K147 HUC151414CSS600 147GB 15000 RPM 64MB Cache SAS 6Gb/s 2.5" Enterprise Internal Hard Drive Bare Drive • 15000 RPM • 64MB Cache • SAS 6Gb/s • 3 years limited						
12	22501-0003-00A	NVMe SSD HGST 1.6TB, SN200	8	1,306.62	10,452.96	0%	**10,452.96**	Resale (HGST)
		Detail Specification						
		Capacity: 1.6TB Speed: Solid State Memory Interface Types: PCIe Form Factor: 2.5inx15mm SFF Server Drive Sector Size: 512 / 512e		Sustained Throughput: 3300/2100 Electrical Interface: PCIe - v3.0 - 32GBps SSD Type: MLC SSD Endurance DWPD: Enterprise Read Intensive Manufacture: HGST Ultrastar SN200 HUSPR3216ADP301				
13	22501-0001-00A	NVMe SSD SAMSUNG 1.6TB, PM1725A	14	1,306.62	18,292.68	0%	**18,292.68**	Resale (Samsung)
		Detail Specification						
		Capacity: 1.6TB Speed: Solid State Memory Interface Types: PCIe Form Factor: 2.5inx15mm SFF Server Drive Sector Size: 512 / 512e		Sustained Throughput: 3300/2950 Electrical Interface: PCIe - v3.0 - 32GBps SSD Type: Samsung V-NAND SSD Endurance DWPD: 5 DWPD for 5 years Manufacture: Samsung PM1725A MZWLL1T6HEHP-00003				
14	22501-0002-00A	NVMe SSD Intel 1.6TB, DCP46xx Series	56	1,306.62	73,170.73	0%	**73,170.73**	Resale (Intel)
		Detail Specification						
		Capacity: 1.6TB Speed: Solid State Memory Interface Types: PCIe Form Factor: 2.5inx15mm SFF Server Drive Sector Size: 512 / 512e		Sustained Throughput: 3200/2100 Electrical Interface: PCIe - v3.0 - 32GBps SSD Type: 64-Layer 3D TLC NAND SSD Endurance: 12.25PBW Manufacture: Intel DCP46xx SSDPE2KE016T7, SSDPE2KE016T8				
15	31020-0008-00	Board PCI-e Switching GEN3 x8 4Port for HHS2112F,HHS3124F	20	5,088.95	100,779.00	50%	**50,389.50**	
		Detail Specification						
		Connect up to 4 NVMe drives via direct connection inside the box Fit into rack-mounted servers with low-profile form factor and side-mounted SAS connectors Support critical, high-bandwidth applications with PCIe® 3.0 connectivity						
16	10401-0001-00A	MSB7890ES2F 100G IB SWITCH By Mellanox	1	18,423.08	18,423.08	0%	**18,423.08**	Resale (Mellanox)
		Detail Specification						
		InfiniBand EDR 100Gb/s Switch System 36-port Non-blocking Externally-managed EDR 100Gb/s InfiniBand Smart Switch 19" rack mountable 1U chassis 36 QSFP28 non-blocking ports with aggregate data throughput up to 7.2Tb/s (EDR) Typical Power with Passive Cables (ATIS): 122W						
17	22809-0001-00A	MCP1600-E002 IB EDR CABLE	30	136.76	4,102.79	0%	**4,102.79**	Resale (Mellanox)
		Detail Specification						
		Passive Copper Cable IB EDR up to 100Gb/s QSFP LSZH 2m 26AWG Technology: InfiniBand Max Speed: EDR Material: Copper Connector Type: QSFP28 Passive/Active: Passive Length: 2m ECCN: EAR99						
18	58090-0001-00A	42U Rack Cabinet	1	776.59	776.59	0%	**776.59**	Resale (Defog)
		Detail Specification						
		Construction : Bolted Material : Aluminum extrusion and Sheet steel Color : BLACK [EN302K] / [U-BK0556-W0] Size : H2030(42U) x D1000 x W600 Product Weight (KG) : 87.5						

19	60304-0002-00A	Storage Software per Node	10	20,000.00	200,000.00	50%	**100,000.00**	Resale (Gluesys)
		Detail Specification						
		Anystor Enterprise NAS O/S (Good SW Certified) - RAID: 0, 1, 10, 5, 6 Support Protocol Support - NFS, CIFS, AFP NVMe Options - SW RAID based Block Storage Support - Web UI NVMe & NVMe-oF Management Tool		AnyManager - Web UI NAS Management Tool - Volume Managent & Monitoring - Auto / Manual recovery - POSIX FS API Support - SSD Acceleration module - Monitoring Tool on WEB (WMS)				
Sub Total					1,045,223.26	42%	601,537.07	
20	A0103-0001-00A	Premium Package 3-Year Support & Maintenance per Node	10	18,046.00	180,460.00	50%	**90,230.00**	
Total					**1,225,683.26**	44%	**691,767.07**	

Note
1. Payment: L/C or T/T
2. Shipment: F.O.B. Inchon

APPENDIX C: TUNING PARAMETERS AND OPTIONS

The following scripts, listed below, were used to set tuning parameters and options:

- The script ***rc.local*** was used to set `aio-max-nr` to the value of `max_user_watches` and to set `ulimit` to 1000.

The script described above are included in the Supporting Files (see Appendix A) and listed below.

rc.local

```
#!/bin/bash
# THIS FILE IS ADDED FOR COMPATIBILITY PURPOSES
#
# It is highly advisable to create own systemd services or udev rules
# to run scripts during boot instead of using this file.
#
# In contrast to previous versions due to parallel execution during boot
# this script will NOT be run after all other services.
#
# Please note that you must run 'chmod +x /etc/rc.d/rc.local' to ensure
# that this script will be executed during boot.

touch /var/lock/subsys/local
cat /proc/sys/fs/epoll/max_user_watches >> /proc/sys/fs/aio-max-nr
ulimit -n 1000
```

APPENDIX D: STORAGE CONFIGURATION CREATION

Environment

First, the following shell scripts are executed on each of the storage nodes.

- ***parted.sh***
- ***mdcreate.sh***
- ***rm_parted.sh***
- ***mdfrozen.sh***
- ***create_nvmet_md.sh***

Step 1 - Create Partitions, RAID volumes

The ***parted.sh*** shell scripts listed below, performs the following actions:

- Create 2 partitions on each NVMe device

The ***mdcreate.sh*** shell scripts listed below, performs the following actions:

- On each node, create 8 RAID 1 volumes using 2 partitions across 2 nearby NVMe devices (only 6 volumes were created on nodes with only 6 NVMe SSDs)
- Collects RAID configuration information for use during reboot
- Invoke the ***rm_parted.sh*** script to remove any pre-existing partitions from the RAID volumes
- Invoke the ***mdfrozen.sh*** script to stop RAID re-sync activity

The command files described above are included in the Supporting Files (see Appendix A) and listed below.

parted.sh

```
#!/bin/sh

devs=$(ls /dev/nvme?n?)

for dev in $devs
do
   parted -a optimal -s $dev mklabel gpt mkpart primary 1 50% mkpart primary 50%
   100%; parted -s $dev unit s print
done

sleep 2
lsblk

sleep 2
/root/mgmt_script/set_kernel_parameters.sh
```

mdcreate.sh

```
#!/bin/sh

(echo "y") | mdadm -C /dev/md1 -l 1 -n 2 -c 512 /dev/nvme0n1p1 /dev/nvme1n1p1 --
   force
(echo "y") | mdadm -C /dev/md2 -l 1 -n 2 -c 512 /dev/nvme1n1p2 /dev/nvme0n1p2 --
   force

(echo "y") | mdadm -C /dev/md3 -l 1 -n 2 -c 512 /dev/nvme2n1p1 /dev/nvme3n1p1 --
   force
(echo "y") | mdadm -C /dev/md4 -l 1 -n 2 -c 512 /dev/nvme3n1p2 /dev/nvme2n1p2 --
   force

(echo "y") | mdadm -C /dev/md5 -l 1 -n 2 -c 512 /dev/nvme4n1p1 /dev/nvme5n1p1 --
   force
(echo "y") | mdadm -C /dev/md6 -l 1 -n 2 -c 512 /dev/nvme5n1p2 /dev/nvme4n1p2 --
   force

(echo "y") | mdadm -C /dev/md7 -l 1 -n 2 -c 512 /dev/nvme6n1p1 /dev/nvme7n1p1 --
   force
(echo "y") | mdadm -C /dev/md8 -l 1 -n 2 -c 512 /dev/nvme7n1p2 /dev/nvme6n1p2 --
   force

mdadm --verbose --detail --scan > /etc/mdadm.conf
/root/mgmt_script/rm_parted.sh
sleep 2
/root/mgmt_script/mdfrozen.sh
```

rm_parted.sh

```
#!/bin/sh

parted -s /dev/md1 rm 1 rm 2
parted -s /dev/md2 rm 1 rm 2
parted -s /dev/md3 rm 1 rm 2
parted -s /dev/md4 rm 1 rm 2
parted -s /dev/md5 rm 1 rm 2
parted -s /dev/md6 rm 1 rm 2
parted -s /dev/md7 rm 1 rm 2
parted -s /dev/md8 rm 1 rm 2

lsblk
```

mdfrozen.sh

```
#!/bin/sh
cmd=$1

devs=$(ls /dev/md? | awk -F '/' '{print $3}')

for dev in $devs;
do
   echo frozen >> /sys/block/$dev/md/sync_action
done

sleep 2
cat /proc/mdstat
```

Step 2 – Set-Up NVMe Over Fabric

The ***create_nvmet_md.sh*** command file, listed below, includes all the CLI commands to perform the following actions:

- Create an NVMe subsystem
- Create an NVMe namespace for each RAID volume
- Set the NVMe device paths
- Enable the namespace
- Create NVMe_oF connections

The command file described above is included in the Supporting Files (see Appendix A) and listed below.

create_nvmet_md.sh

```
#!/bin/sh
nvmetcli clear

mkdir /sys/kernel/config/nvmet/subsystems/nvme-ib0
echo 1 > /sys/kernel/config/nvmet/subsystems/nvme-ib0/attr_allow_any_host

mkdir /sys/kernel/config/nvmet/subsystems/nvme-ib0/namespaces/1
echo        -n        /dev/md1        >/sys/kernel/config/nvmet/subsystems/nvme-
   ib0/namespaces/1/device_path
echo 1 > /sys/kernel/config/nvmet/subsystems/nvme-ib0/namespaces/1/enable

mkdir /sys/kernel/config/nvmet/subsystems/nvme-ib0/namespaces/2
echo        -n        /dev/md2        >/sys/kernel/config/nvmet/subsystems/nvme-
   ib0/namespaces/2/device_path
echo 1 > /sys/kernel/config/nvmet/subsystems/nvme-ib0/namespaces/2/enable

mkdir /sys/kernel/config/nvmet/subsystems/nvme-ib0/namespaces/3
echo        -n        /dev/md3        >/sys/kernel/config/nvmet/subsystems/nvme-
   ib0/namespaces/3/device_path
echo 1 > /sys/kernel/config/nvmet/subsystems/nvme-ib0/namespaces/3/enable

mkdir /sys/kernel/config/nvmet/subsystems/nvme-ib0/namespaces/4
echo        -n        /dev/md4        >/sys/kernel/config/nvmet/subsystems/nvme-
   ib0/namespaces/4/device_path
echo 1 > /sys/kernel/config/nvmet/subsystems/nvme-ib0/namespaces/4/enable

mkdir /sys/kernel/config/nvmet/ports/1
echo 4420 > /sys/kernel/config/nvmet/ports/1/addr_trsvcid
echo 1.1.1.201 > /sys/kernel/config/nvmet/ports/1/addr_traddr
echo rdma > /sys/kernel/config/nvmet/ports/1/addr_trtype
echo ipv4 > /sys/kernel/config/nvmet/ports/1/addr_adrfam

ln              -s              /sys/kernel/config/nvmet/subsystems/nvme-ib0
   /sys/kernel/config/nvmet/ports/1/subsystems/nvme-ib0

mkdir /sys/kernel/config/nvmet/subsystems/nvme-ib1
echo 1 > /sys/kernel/config/nvmet/subsystems/nvme-ib1/attr_allow_any_host

mkdir /sys/kernel/config/nvmet/subsystems/nvme-ib1/namespaces/5
echo        -n        /dev/md5        >/sys/kernel/config/nvmet/subsystems/nvme-
   ib1/namespaces/5/device_path
echo 1 > /sys/kernel/config/nvmet/subsystems/nvme-ib1/namespaces/5/enable
```

```
mkdir /sys/kernel/config/nvmet/subsystems/nvme-ib1/namespaces/6
echo        -n        /dev/md6        >/sys/kernel/config/nvmet/subsystems/nvme-
   ib1/namespaces/6/device_path
echo 1 > /sys/kernel/config/nvmet/subsystems/nvme-ib1/namespaces/6/enable

mkdir /sys/kernel/config/nvmet/subsystems/nvme-ib1/namespaces/7
echo        -n        /dev/md7        >/sys/kernel/config/nvmet/subsystems/nvme-
   ib1/namespaces/7/device_path
echo 1 > /sys/kernel/config/nvmet/subsystems/nvme-ib1/namespaces/7/enable

mkdir /sys/kernel/config/nvmet/subsystems/nvme-ib1/namespaces/8
echo        -n        /dev/md8        >/sys/kernel/config/nvmet/subsystems/nvme-
   ib1/namespaces/8/device_path
echo 1 > /sys/kernel/config/nvmet/subsystems/nvme-ib1/namespaces/8/enable

mkdir /sys/kernel/config/nvmet/ports/2
echo 4420 > /sys/kernel/config/nvmet/ports/2/addr_trsvcid
echo 1.1.2.201 > /sys/kernel/config/nvmet/ports/2/addr_traddr
echo rdma > /sys/kernel/config/nvmet/ports/2/addr_trtype
echo ipv4 > /sys/kernel/config/nvmet/ports/2/addr_adrfam

ln               -s               /sys/kernel/config/nvmet/subsystems/nvme-ib1
   /sys/kernel/config/nvmet/ports/2/subsystems/nvme-ib1

sleep 2
nvmetcli save
nvmetcli ls
```

Step 3 - Create Volumes on the Host Systems

The following CLI commands were executed on the master host to perform the following actions:

- `nvme discover` to query the NVMe-over-Fabrics
- `nvme connect-all` to discover and connect to the Fabric controllers

The ***lvcreate.sh*** shell scripts listed below, performs the following actions:

- Create physical volumes for each RAID volume
- Create 1 volume group
- Create 9 logical volumes for ASU1
- Create 9 logical volumes for ASU2
- Create 1 logical volume for ASU3

The remaining hosts were connected to the newly created NVMe_oF using the CLI commands described above.

The shell script described above is included in the Supporting Files (see Appendix A) and listed below.

lvcreate.sh

```
#!/bin/sh
dev='/dev/nvme*n*'
```

```
pvcreate $dev
pvs --units G

vgcreate vg1 $dev
vgs --units G

num=$(ls $dev | wc -l)
lvcreate -i$num -l5%VG -I512 vg1 -n asu1_1 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu1_2 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu1_3 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu1_4 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu1_5 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu1_6 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu1_7 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu1_8 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu1_9 $dev

lvcreate -i$num -l5%VG -I512 vg1 -n asu2_1 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu2_2 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu2_3 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu2_4 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu2_5 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu2_6 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu2_7 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu2_8 $dev
lvcreate -i$num -l5%VG -I512 vg1 -n asu2_9 $dev

lvcreate -i$num -l10%VG -I512 vg1 -n asu3_1 $dev

lvs --units G -v --segment
```

APPENDIX E: CONFIGURATION INVENTORY

An inventory of the Tested Storage Configuration (TSC) was collected during the execution the ***full_run.sh*** script. It generated the following log file:

- ***inventory_start.out*** List of configured volumes before the INIT Phase.
- ***inventory_end.out*** List of configured volumes after the PERSIST_2 Phase.

The above log files are included in the Supporting Files (see Appendix A).

APPENDIX F: WORKLOAD GENERATOR

The ASUs accessed by the SPC-1 workload generator, are defined using the script ***hst1_linear.asu***.

The phases of the benchmark are executed using the script ***spc1.sh***. The script pauses before and after the PERSIST_1 test phase. The operator starts the PERSIST_1 test phase by pressing ENTER from the console where the script has been invoked. Once the TSC has been restarted, the PERSIST_2 test phase is executed by pressing ENTER once more.

The above scripts are included in the Supporting Files (see Appendix A) and listed below.

hst1_linear.asu

```
-- SPC-1 ASU definition file
-- $:id:
--
Offset = 0
--
ASU=1
device=/dev/vg1/asu1_1
device=/dev/vg1/asu1_2
device=/dev/vg1/asu1_3
device=/dev/vg1/asu1_4
device=/dev/vg1/asu1_5
device=/dev/vg1/asu1_6
device=/dev/vg1/asu1_7
device=/dev/vg1/asu1_8
device=/dev/vg1/asu1_9
--
ASU=2
device=/dev/vg1/asu2_1
device=/dev/vg1/asu2_2
device=/dev/vg1/asu2_3
device=/dev/vg1/asu2_4
device=/dev/vg1/asu2_5
device=/dev/vg1/asu2_6
device=/dev/vg1/asu2_7
device=/dev/vg1/asu2_8
device=/dev/vg1/asu2_9
---
ASU=3
device=/dev/vg1/asu3_1
```

spc1.sh

```
#!/bin/sh
#spc1.sh A B
#A : SPC1 TEST PHASE
#B : SPC1 IOPS
cmd=$1
iops=$2

date=$(date '+%Y-%m-%d_%H:%M:%S')
```

```
date_time=$(date '+%H:%M:%S')
spc_home="/root/spc"
spc_master=$spc_home"/5host.HST"
log_home="/root/Logs"
target_ip="10.144.144.201    10.144.144.202    10.144.144.203    10.144.144.204
   10.144.144.205  10.144.144.206  10.144.144.207  10.144.144.208  10.144.144.209
   10.144.144.210"
host_ip="10.144.144.101     10.144.144.102     10.144.144.103     10.144.144.104
   10.144.144.105"

for ip in $target_ip;
do
   ping $ip -c 3
   echo ----------------------------------------------------------------
done

for ip in $host_ip;
do
   ping $ip -c 3
   echo ----------------------------------------------------------------
done

echo "ready?"
read

mkdir -p $log_home"/"$date
log_dir=$log_home"/"$date
sys_log=$log_dir"/inventory_start.out"
sys_log_end=$log_dir"/inventory_end.out"
mdadm_log=$log_dir"/mdadm.log"
stat_log_dir=$log_dir"/stat"
parm_log=$log_dir"/parm.log"
mkdir -p $stat_log_dir

cp /root/mgmt_script/spc1.sh $log_dir"/spc1-backup.sh"
cp $spc_home"/5host.HST" $log_dir"/5host.HST"
cp $spc_home"/SPC1_METRICS" $log_dir"/SPC1_METRICS"

echo "1. Get host information."

cat /etc/os-release >> $sys_log
echo -e "==============================================\n" >> $sys_log
uname -r >> $sys_log
echo -e "==============================================\n" >> $sys_log
nvme list >> $sys_log
echo -e "==============================================\n" >> $sys_log
lsblk -b >> $sys_log
echo -e "==============================================\n" >> $sys_log
fdisk -l >> $sys_log
echo -e "==============================================\n" >> $sys_log
#cat /proc/mdstat >> $sys_log
#echo -e "==============================================\n" >> $sys_log
pvs --units G >> $sys_log
echo -e "==============================================\n" >> $sys_log
vgs --units G >> $sys_log
echo -e "==============================================\n" >> $sys_log
lvs --units G -v --segment >> $sys_log
echo -e "==============================================\n" >> $sys_log

echo "2. Get NVMe information.(smartctl)"

for ip in $target_ip;
```

```
do
   echo ====================$ip================= >> $sys_log
   dn=$(ssh root@$ip 'ls /dev/nvme?' | wc -l)
   #dn=`expr $dn - 1`
   for ((i=0;i<$dn;i++));
   do
      echo ++++++++++++++++++++/dev/nvme$i++++++++++++++++++ >> $sys_log
       ssh root@$ip smartctl -a /dev/nvme$i >> $sys_log
   done
done

echo "3. Get Storage-RAID information."

for ip in $target_ip;
do
   echo ====================$ip================= >> $mdadm_log
   dn=$(ssh root@$ip 'ls /dev/md?' | wc -l)
   for ((i=1;i<=$dn;i++));
   do
      echo ++++++++++++++++++++/dev/md$i++++++++++++++++++ >> $mdadm_log
       ssh root@$ip mdadm -D /dev/md$i >> $mdadm_log
   done

   echo ++++++++++++++++++++/etc/mdadm.conf++++++++++++++++++ >> $mdadm_log
       ssh root@$ip cat /etc/mdadm.conf >> $mdadm_log
done

echo "4. Get host disk parameter."

for ip in $host_ip;
do
   devs=$(ssh root@$ip 'ls /dev/nvme?n?' | awk -F '/' '{print $3}')

   echo ====================$ip==================== >> $parm_log

   for dev in $devs;
   do
      echo +++++++++++++++/dev/$dev+++++++++++++++ >> $parm_log
      #devices configuration setup
      echo -n 'nr_requests : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/nr_requests >> $parm_log
      echo -n 'scheduler : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/scheduler >> $parm_log
      echo -n 'max_sectors_kb : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/max_sectors_kb >> $parm_log
      echo -n 'add_random : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/add_random >> $parm_log
      echo -n 'nomerges : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/nomerges >> $parm_log
   done
   #OS configuration setup
   echo '+++++++++++++++system parameter+++++++++++++++' >> $parm_log
   echo -n 'aio-max-nr : '>> $parm_log
   echo '-----------------------------------------------' >> $parm_log
   ssh root@$ip cat /proc/sys/fs/aio-max-nr >> $parm_log
   echo 'ulimit -a' >> $parm_log
   ssh root@$ip ulimit -a >> $parm_log
   echo '-----------------------------------------------' >> $parm_log
   echo 'ulimit -aH' >> $parm_log
   ssh root@$ip ulimit -aH >> $parm_log
done
```

```
echo "5. Get storage NVMe parameter."

for ip in $target_ip;
do
   devs=$(ssh root@$ip 'ls /dev/nvme?n?' | awk -F '/' '{print $3}')

   echo ====================$ip==================== >> $parm_log

   for dev in $devs;
   do
      echo +++++++++++++++/dev/$dev+++++++++++++++ >> $parm_log
      #devices configuration setup
      echo -n 'nr_requests : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/nr_requests >> $parm_log
      echo -n 'scheduler : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/scheduler >> $parm_log
      echo -n 'max_sectors_kb : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/max_sectors_kb >> $parm_log
      echo -n 'add_random : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/add_random >> $parm_log
      echo -n 'nomerges : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/nomerges >> $parm_log
   done
   #OS configuration setup
   echo '+++++++++++++++system parameter+++++++++++++++' >> $parm_log
   echo -n 'aio-max-nr : ' >> $parm_log
   ssh root@$ip cat /proc/sys/fs/aio-max-nr >> $parm_log
   echo '----------------------------------------------' >> $parm_log
   echo 'ulimit -a' >> $parm_log
   ssh root@$ip ulimit -a >> $parm_log
   echo '----------------------------------------------' >> $parm_log
   echo 'ulimit -aH' >> $parm_log
   ssh root@$ip ulimit -aH >> $parm_log
done

echo "6. Get storage RAID-Volume parameter."

for ip in $target_ip;
do
   devs=$(ssh root@$ip 'ls /dev/md?' | awk -F '/' '{print $3}')

   echo ====================$ip==================== >> $parm_log

   for dev in $devs;
   do
      echo +++++++++++++++/dev/$dev+++++++++++++++ >> $parm_log
      #devices configuration setup
      echo -n 'nr_requests : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/nr_requests >> $parm_log
      echo -n 'scheduler : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/scheduler >> $parm_log
      echo -n 'max_sectors_kb : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/max_sectors_kb >> $parm_log
      echo -n 'add_random : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/add_random >> $parm_log
      echo -n 'nomerges : ' >> $parm_log
      ssh root@$ip cat /sys/block/$dev/queue/nomerges >> $parm_log
   done
done

echo "7. Start monitoring system."
```

```
#storage
for ip in $target_ip;
do
   ssh root@$ip '/root/mgmt_script/kill.sh vmstat'
   ssh root@$ip '/root/mgmt_script/kill.sh iostat'
   ssh root@$ip 'rm -rf /tmp/*stat.log'
   ssh root@$ip 'vmstat -t 10 > /tmp/vmstat.log &'
   ssh root@$ip 'iostat -mtx 10 > /tmp/iostat.log &'
done

#host
for ip in $host_ip;
do
   ssh root@$ip '/root/mgmt_script/kill.sh vmstat'
   ssh root@$ip '/root/mgmt_script/kill.sh iostat'
   ssh root@$ip 'rm -rf /tmp/*stat.log'
   ssh root@$ip 'vmstat -t 10 > /tmp/vmstat.log &'
   ssh root@$ip 'iostat -mtx 10 > /tmp/iostat.log &'
done

echo "8. Start SPC-1 Test."
sleep 1
echo -n "3.."
sleep 1
echo -n "2.."
sleep 1
echo -n "1.."
sleep 1
echo "Start!!"
sleep 2

## running spc1
/root/spc/spc1 -run SPC1_INIT -iops 15000 -master $spc_master -output $log_dir
/root/spc/spc1 -run SPC1_VERIFY -iops 100 -master $spc_master -output $log_dir
/root/spc/spc1  -run  SPC1_METRICS  -iops  2410000  -master  $spc_master  -output
   $log_dir
/root/spc/spc1 -run SPC1_VERIFY -iops 100 -master $spc_master -output $log_dir
sleep 120

echo "9. Get system monitoring logs."
#storage
for ip in $target_ip;
do
   ssh root@$ip '/root/mgmt_script/kill.sh vmstat'
   ssh root@$ip '/root/mgmt_script/kill.sh iostat'
   scp root@$ip:/tmp/vmstat.log $stat_log_dir"/"$ip"_vmstat.log"
   scp root@$ip:/tmp/iostat.log $stat_log_dir"/"$ip"_iostat.log"
done

#host
for ip in $host_ip;
do
   ssh root@$ip '/root/mgmt_script/kill.sh vmstat'
   ssh root@$ip '/root/mgmt_script/kill.sh iostat'
   scp root@$ip:/tmp/vmstat.log $stat_log_dir"/"$ip"_vmstat.log"
   scp root@$ip:/tmp/iostat.log $stat_log_dir"/"$ip"_iostat.log"
done

echo "Stop"
read
/root/spc/spc1  -run  SPC1_PERSIST_1  -iops  602500  -master  $spc_master  -output
   $log_dir
```

```
echo "reboot"
read
/root/spc/spc1  -run  SPC1_PERSIST_2  -iops  602500  -master  $spc_master  -output
   $log_dir

## finish spc1
nvme list >> $sys_log_end

echo -e "=============================================\n" >> $sys_log_end
lsblk -b >> $sys_log_end
echo -e "=============================================\n" >> $sys_log_end
fdisk -l >> $sys_log_end
echo -e "=============================================\n" >> $sys_log_end
#cat /proc/mdstat >> $sys_log_end
#echo -e "=============================================\n" >> $sys_log_end
pvs --units G >> $sys_log_end
echo -e "=============================================\n" >> $sys_log_end
vgs --units G >> $sys_log_end
echo -e "=============================================\n" >> $sys_log_end
lvs --units G -v --segment >> $sys_log_end
echo -e "=============================================\n" >> $sys_log_end

for ip in $target_ip;
do
   echo ====================$ip================ >> $sys_log_end
   dn=$(ssh root@$ip 'ls /dev/nvme?' | wc -l)
   #dn=`expr $dn - 1`
   for ((i=0;i<=$dn;i++));
   do
      echo ++++++++++++++++++++/dev/nvme$i++++++++++++++++++ >> $sys_log_end
       ssh root@$ip smartctl -a /dev/nvme$i >> $sys_log_end
   done
done

for ip in $target_ip
do
   echo ====================$ip====================
   ssh root@$ip 'rm -rf /tmp/*stat.log'
done

for ip in $host_ip
do
   echo ====================$ip====================
   ssh root@$ip 'rm -rf /tmp/*stat.log'
done
```