Solid State Storage (SSS)
Performance Test Specification (PTS)
Client

Version 1.0

This document has been released and approved by the SNIA. The SNIA believes that the ideas, methodologies and technologies described in this document accurately represent the SNIA goals and are appropriate for widespread distribution. Suggestion for revision should be directed to http://www.snia.org/feedback/.

SNIA Technical Position

August 6, 2011

Revision History

Revision	Release Date	Originator	Comments
V0.97	11JAN11	Dave Landsman	• Inserted Client test definitions previously approved by TWG in SSS PTS V0.9 into Enterprise PTS V1.0. • Changed appropriate references from Enterprise to Client
V1.0 rev A	04APR11	Eden Kim	• F2F Group Edits on active range changes and definitions • Easen Ho active range changes IOPS, TP, LAT • Eden Kim addition of Preamble section 1.1
V1.0 rev B	11APR11	Eden Kim	• TWG Group Edits – agreement on changes • Chuck Paridon – ActiveRange definition

Contributors

The SNIA SSS Technical Work Group, which developed and reviewed this standard, would like to recognize the contributions made by the following members:

Company	Contributor
AmerNet, Inc.	Khaled Amer
Calypso	Eden Kim
Calypso	Easen Ho
Calypso	Mike Peeler
Coughlin Assoc	Tom Coughlin
Dell	Gary Kotzur
EMC	Don Deel
Fusion-IO	Jonathon Thatcher
Hitachi Data Systems	Mel Boksenbaum
Hitachi Global Storage Technologies	Dan Colegrove
Hewlett-Packard	Chuck Paridon
hyperI/O	Tom West
IBM	Phil Mills
Intel	Brady Foster
Intel	Terry Yoshii
Intel	Harry Pon
LSI	Harry Mason
Marvell Semiconductor	Paul Wassenberg
Micron Technology	Doug Rollins
Objective Analysis	Jim Handy
Pliant	Mike Chenery
PMC-Sierra	Martin Harris
Samsung	Steven Peng
SandForce	Jeremy Werner
SanDisk	Dave Landsman
SanDisk	Spencer Ng
Seagate	Marty Czekalski
Seagate	Alvin Cox
Smart Modular	Esther Spanjer
SNIA	Arnold Jones
Toshiba TAIS	Sumit Puri
Western Digital	Gary Drossel

Usage

Disclaimer

The information contained in this publication is subject to change without notice. The SNIA makes no warranty of any kind with regard to this specification, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The SNIA shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this specification.

Contacting SNIA

SNIA Web Site

Current SNIA practice is to make updates and other information available through their web site at http://www.snia.org

SNIA Address

Requests for interpretation, suggestions for improvement and addenda, or defect reports are welcome. They should be sent via the SNIA Feedback Portal at http://www.snia.org/feedback/ or by mail to the Storage Networking Industry Association, 425 Market Street, Suite 1020, San Francisco, CA 94105, U.S.A.

Suggestions for revisions should be directed to http://www.snia.org/feedback/.

Intended Audience

This document is intended for use by individuals and companies engaged in the development of this Specification and in validating the tests and procedures incorporated herein. After approvals and release to the public, this Specification is intended for use by individuals and companies engaged in the design, development, qualification, manufacture, test, acceptance and failure analysis of SSS devices and systems and sub systems incorporating SSS devices.

Changes to the Specification

Each publication of this Specification is uniquely identified by a two-level identifier, comprised of a version number and a release number. Future publications of this specification are subject to specific constraints on the scope of change that is permissible from one publication to the next and the degree of interoperability and backward compatibility that should be assumed between products designed to different publications of this standard. The SNIA has defined three levels of change to a specification:

- Major Revision: A major revision of the specification represents a substantial change to the underlying scope or architecture of the specification. A major revision results in an increase in the version number of the version identifier (e.g., from version 1.x to version

2.x). There is no assurance of interoperability or backward compatibility between releases with different version numbers.

- Minor Revision: A minor revision of the specification represents a technical change to existing content or an adjustment to the scope of the specification. A minor revision results in an increase in the release number of the specification's identifier (e.g., from x.1 to x.2). Minor revisions with the same version number preserve interoperability and backward compatibility.

Table of Contents

Revision History **2**

Contributors **3**

Usage **4**

Disclaimer 4
Contacting SNIA 4
Intended Audience 4
Changes to the Specification 4

Table of Contents **6**

List of Tables **8**

List of Figures **8**

1 Introduction **9**

1.1 Preamble 9
1.2 Purpose 9
1.3 Background 9
1.4 Scope 10
1.5 Not in Scope 10
1.6 Disclaimer 10
1.7 Normative References 11
1.7.1 Approved references 11
1.7.2 References under development 11
1.7.3 Other references 11

2 Definitions, symbols, abbreviations, and conventions **12**

2.1 Definitions 12
2.2 Acronyms and Abbreviations 13
2.3 Keywords 14
2.4 Conventions 14
2.4.1 Number Conventions 14
2.4.2 Pseudo Code Conventions 14

3 Key Test Process Concepts **16**

3.1 Steady State 16
3.2 Purge 16
3.3 Preconditioning 16
3.4 ActiveRange 17
3.5 ActiveRange Segmentation 17
3.6 Data Patterns 18
3.7 Multiple Thread Guideline 18
3.8 Caching 18

4 Overview of Common Test Flow **19**

5 Common Reporting Requirements **20**

5.1 General 20
5.2 Test System Hardware 20
5.3 Test System Software 20
5.4 Device Under Test 20

6 Test Tool Guidelines **21**

7 IOPS Test **22**

7.1 Test Specific Reporting for IOPS Test 25
7.1.1 Purge 25

7.1.2 Workload Independent Preconditioning..25
7.1.3 Steady State Convergence..25
7.1.4 Steady State Verification..26
7.1.5 Measurement..27
7.1.5.1 Measurement Window Summary Data Table..27
7.1.5.2 Measurement Plot – 2D..27
7.1.5.3 Measurement Plot – 3D..28
8 Throughput Test..29
8.1 Test Specific Reporting for Throughput Test..31
8.1.1 Purge Report..31
8.1.2 Workload Independent Preconditioning Report..31
8.1.3 Steady State Convergence Report..31
8.1.4 Steady State Verification Report..32
8.1.5 Measurement Window Report..32
8.1.5.1 Measurement Window Summary Data Table..32
9 Latency Test..33
9.1 Test Specific Reporting for Latency Test..35
9.1.1 Purge..35
9.1.2 Workload Independent Preconditioning..35
9.1.3 Steady State Convergence..35
9.1.4 Steady State Verification Report..36
9.1.5 Measurement Report..36
9.1.5.1 Measurement Window Summary Data Table..36
9.1.5.2 Measurement Plots..36
10 Informative Annex A – Sample Test Report..38
11 Informative Annex B – Performance Test Platform Example..44
11.1 RTP Configurations..44
11.2 RTP Components..44

List of Tables

<List generated by MS Word>

List of Figures

Figure 1-1 – NAND-based SSS Performance States for 8 Devices 10
Figure 8-1 - Steady State Verification Example Plot 26

1 Introduction

1.1 Preamble

This Client Performance Test Specification 1.0 is a companion specification to the SNIA Solid State Storage Enterprise Performance Test Specification 1.0. Both Performance Test Specifications (PTS) are intended to be used to obtain reliable and comparative measurement of NAND Flash based solid state storage devices. In both PTS in their present form, the tests and methodologies are designed to use a synthetic, or known and repeatable, test stimulus applied to a solid state storage product at the device level. In these PTS, "device level" refers to measurement of block IO at the physical device level as opposed to file system IO in the host Operating System.

Both of these PTS are based on test and preparation methodologies developed by the SNIA SSS TWG for performance test of NAND based solid state storage. NAND Flash based solid state storage (SSS) performance tends to be highly dependent on the write history of the SSS device, the type of stimulus applied to the SSS device, as well as the test environment (both hardware and software) in which the test stimulus is applied and measurements taken. Much of the preconditioning, test condition set up and parameters take these SSS behaviors into consideration.

These PTS do not require the use a specific test environment, but test tool requirements, capabilities and examples are set forth in the specifications. Care should be taken by the test operator to ensure that the test hardware does not bottleneck the SSS device performance, that the OS or test software tool has minimal contribution to test measurements, and that the same hardware and software test combination is used when comparing performance results of different SSS devices.

This Client PTS 1.0 differs from its companion Enterprise PTS 1.0 in the preparation of the Device Under Test (DUT) for steady state performance measurement and in the amount and type of test stimulus applied to the DUT. For example, preconditioning LBA ranges may be limited in the Client PTS to less than 100% of the available LBAs while the test stimulus Active Range may be limited to a reduced number of uniquely touched LBAs (see Client PTS Specification). The use of limited preconditioning and test active ranges are meant to provide a test stimulus that share more characteristics with a typical Client user workload.

Future work of this SSS TWG is chartered to investigate IO Trace based or user workload based performance benchmark test methodologies and may result in a Client PTS 2.0 at some future date. Readers and industry members are encouraged to participate in the further SNIA SSS TWG works and can contact the the TWG at its website portal at http://www.snia.org/feedback/.

1.2 Purpose

Manufacturers need to set, and customers need to compare, the performance of Solid State Storage (SSS) devices. This Specification defines a set of device level tests and methodologies intended to enable comparative testing of SSS devices in Client (see 2.1.4) systems.

Note: While the tests defined in this specification could be applied to SSS devices based on any technology (RAM, NAND, etc.), the emphasis in this specification, in particular regarding Preconditioning and Steady State, is oriented towards NAND.

1.3 Background

A successful device level performance test isolates the device being tested from the underlying test platform (HW, OS, Tools, Applications) so the only limiting variable in the test environment is the device being tested. To achieve this goal with NAND-based SSS devices, in addition to typical system/device isolation issues, the test, and test methodologies, must address attributes unique to NAND-based flash media.

NAND-based SSS device controllers map Logical to Physical blocks on the NAND media, in order to achieve the best NAND performance and endurance. The SSS device manages this

LBA-to-PBA mapping with internal processes that operate independently of the host. The sum of this activity is referred to as "flash management".

The performance of the flash management during a test, and hence the overall performance of the SSS device during the test, depends critically on:

1) Write History and Preconditioning: The state of the device prior to the test
2) Workload Pattern: Pattern of the I/O (r/w mix, block size, etc.) written to device during test
3) Data Pattern: The actual bits in the data payload written to the device

The methodologies defined in the SSS Performance Test Specification (SSS PTS) attempt to create consistent conditions for items 1-3 so that the only variable is the device under test.

The importance of the SSS PTS methodologies on SSS performance measurement is shown in Figure 1-1. A typical SSS device, taken Fresh Out of the Box (FOB), and exposed to a workload, experiences a brief period of elevated performance, followed by a transition to Steady State performance. The SSS PTS ensures that performance measurements are taken in the Steady State region, representing the device's performance during its normal working life.

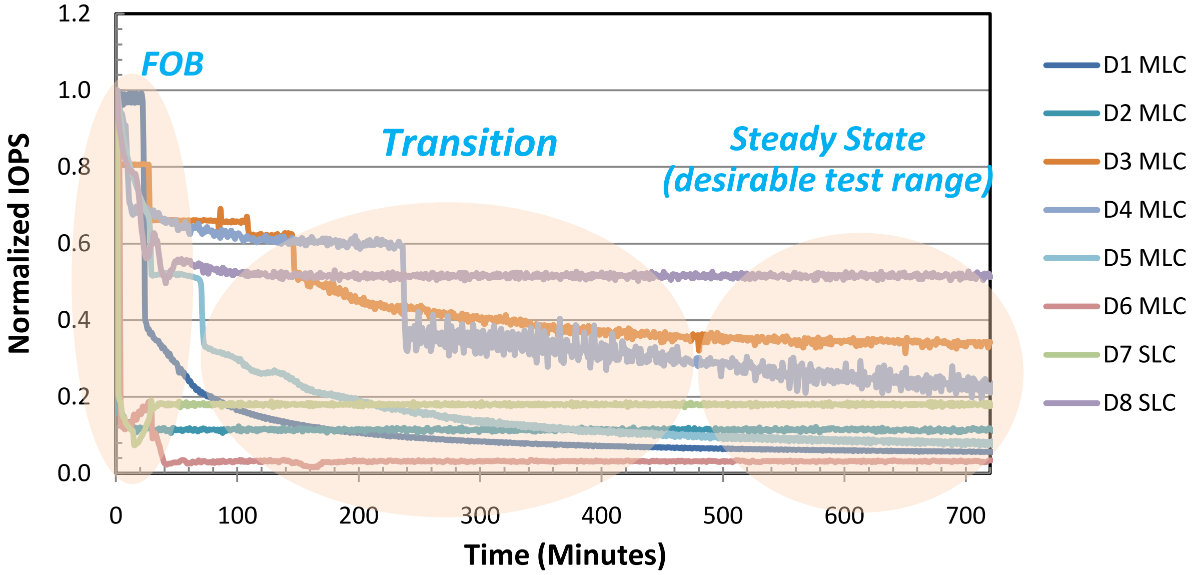


Figure 1-1 – NAND-based SSS Performance States for 8 Devices (4KiB Rnd Wrt)

1.4 Scope

1) Preconditioning methods
2) Performance tests
3) Test reporting requirements

1.5 Not in Scope

1) Application Workload Tests
2) Test Platform (HW/OS/Tools)
3) Certification/Validation procedures for this specification
4) Device reliability, availability, or data integrity

1.6 Disclaimer

Use or recommended use of any public domain, third party or proprietary software does not imply nor infer SNIA or SSS TWG endorsement of the same. Reference to any such test or measurement software, stimulus tools, or software programs is strictly limited to the specific use and purpose as set forth in this Specification and does not imply any further endorsement or verification on the part of SNIA or the SSS TWG.

1.7 Normative References

1.7.1 Approved references

These are the standards, specifications and other documents that have been finalized and are referenced in this specification.

- IDEMA Document LBA1-02 -- LBA Count for IDE Hard Disk Drives Standard
- JEDEC JESD218 – Solid-State Drive (SSD) Requirements and Endurance Test Method
- JEDEC JESD219 – Solid-State Drive (SSD) Endurance Workloads

1.7.2 References under development

- ATA/ATAPI Command Set - 2 (ACS-2) – INCITS/T13 2015-D

1.7.3 Other references

- TBD

2 Definitions, symbols, abbreviations, and conventions

2.1 Definitions

2.1.1 **ActiveRange:** The range of LBA's that can be used for a given Test Code or Preconditioning Code (expressed as a percent of the total addressable LBAs as set forth in section 3.4)

2.1.2 **ActiveRange Amount:** The sum of the capacity referenced by the LBA's that are accessed for a given Test or Preconditioning Code equal to, or less than the capacity referenced by the AcrtiveRange LBAs (such as 8GB or 16GB ActiveRange Amount set forth in Section 3.5).

2.1.3 **ActiveRange Segments:** A collection of contiguous and equal sized LBA ranges within the ActiveRange where the Test Code and Preconditioning Codes are allowed to access. The starting LBA of each ActiveRange Segment is randomly distributed across the entire ActiveRange. The set of ActiveRange segments spans the ActiveRange. Note that ActiveRange segments shall not touch to form a single segment.

2.1.4 **ActiveRange Segment Size:** The size of a single ActiveRange Segment is determined by taking the ActiveRange Amount and dividing by the number of ActiveRange Segments as prescribed in the Test Code or Preconditioning Code.

2.1.5 **Cache:** A volatile or non-volatile data storage area outside the User Capacity that may contain a subset of the data stored within the User Capacity.

2.1.6 **Client:** Single user desktop or laptop system used in home or office.

2.1.7 **Enterprise:** Servers in data centers, storage arrays, and enterprise wide / multiple user environments that employ direct attached storage, storage attached networks and tiered storage architectures.

2.1.8 **Fresh Out of the Box (FOB):** State of SSS prior to being put into service.

2.1.9 **IO Demand:** Measured # of OIOs executing in the host.

2.1.10 **Logical Block Address (LBA):** The address of a logical block, i.e., the offset of the block from the beginning of the logical device that contains it.

2.1.11 **Latency:** The time between when the workload generator makes an IO request and when it receives notification of the request's completion.

2.1.12 **MaxUserLBA:** The maximum LBA # addressable in the User Capacity.

2.1.13 **Measurement Window:** The interval, measured in Rounds, during which test data is collected, bounded by the Round in which the device has been observed to have maintained Steady State for the specified number of Rounds (Round x), and five Rounds previous (Round x-4).

2.1.14 **Nonvolatile Cache:** A cache that retains data through power cycles.

2.1.15 **Outstanding IO (OIO):** The number of IO operations issued by a host, or hosts, awaiting completion.

2.1.16 **OIO/Thread:** The number of OIO allowed per Thread (Worker, Process)

2.1.17 **Over-Provisioned Capacity:** LBA range provided by the manufacturer for performance and endurance considerations, but not accessible by the host file system, operating system, applications, or user.

2.1.18 **Preconditioning:** The process of writing data to the device to prepare it for Steady State measurement consisting of two (2) Preconditioning steps as follows:

18.1. **Workload Independent Preconditioning (WIPC):** The first Preconditioning step comprised of a prescribed workload, unrelated to the test workload, as a means to facilitate convergence to Steady State.

18.2. **Workload Dependent Preconditioning (WDPC):** The second Preconditioning step comprised of running the test workload itself, after Workload Independent Preconditioning, as a means to put the device in a Steady State relative to the dependent variable being tested.

2.1.19 Preconditioning Code: Refers to the Preconditioning steps set forth in this Specification.

Purge: The process of returning an SSS device to a state in which subsequent writes execute, as closely as possible, as if the device had never been used and does not contain any valid data.

2.1.20 **Round**: A complete pass through all the prescribed test points for any given test.

2.1.21 **Steady State:** A device is said to be in Steady State when, for the dependent variable (y) being tracked:

a) Range(y) is less than 20% of Ave(y): Max(y)-Min(y) within the Measurement Window is no more than 20% of the Ave(y) within the Measurement Window; and

b) Slope(y) is less than 10%: Max(y)-Min(y), where Max(y) and Min(y) are the maximum and minimum values on the best linear curve fit of the y-values within the Measurement Window, is within 10% of Ave(y) value within the Measurement Window.

2.1.22 Test Code: Refers to the measurement steps set forth in the test sections contained in this Specification.

2.1.23 **Thread:** Execution context defined by host OS/CPU (also: Process, Worker)

2.1.24 **Thread Count (TC):** Number of Threads (or Workers or Processes) specified by a test.

2.1.25 **Total OIO:** Total outstanding IO Operations specified by a test = (OIO/Thread) * (TC)

2.1.26 **User Capacity:** LBA range directly accessible by the file system, operating system and applications, not including Over-Provisioned Capacity.

2.1.27 **Workload Based Preconditioning:** The technique of running the test workload itself, typically after Workload Independent Preconditioning, as a means to put the device in a Steady State relative to the dependent variable being tested.

2.1.28 **Workload Independent Preconditioning:** The technique of running a prescribed workload, unrelated, except by possible coincidence, to the test workload, as a means to facilitate convergence to Steady State.

2.1.29 **Volatile Cache:** A cache that does not retain data through power cycles.

2.2 Acronyms and Abbreviations

2.2.1 **IOPS:** I/O Operations per Second

2.2.2 **DUT:** Device Under Test

2.2.3 **FOB:** Fresh Out of Box

2.2.4 **OIO:** Outstanding IO

2.2.5 **R/W:** Read/Write

2.2.6 **SSSI:** Solid State Storage Initiative

2.2.7 **SSS TWG:** Solid State Storage Technical Working Group

2.2.8 **TC:** Thread Count

2.3 Keywords

The key words “shall”, “required”, “shall not”, “should”, “recommended”, “should not”, “may”, and “optional” in this document are to be interpreted as:

2.3.1 **Shall:** This word, or the term "required", means that the definition is an absolute requirement of the specification.

2.3.2 **Shall Not:** This phrase means that the definition is an absolute prohibition of the specification.

2.3.3 **Should:** This word, or the adjective "recommended", means that there may be valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and weighed before choosing a different course.

2.3.4 **Should Not:** This phrase, or the phrase "not recommended", means that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.

2.3.5 **May:** This word, or term “optional”, indicates flexibility, with no implied preference.

2.4 Conventions

2.4.1 Number Conventions

Numbers that are not immediately followed by lower-case b or h are decimal values.

Numbers immediately followed by lower-case b (xxb) are binary values.

Numbers immediately followed by lower-case h (xxh) are hexadecimal values.

Hexadecimal digits that are alphabetic characters are upper case (i.e., ABCDEF, not abcdef).

Hexadecimal numbers may be separated into groups of four digits by spaces. If the number is not a multiple of four digits, the first group may have fewer than four digits (e.g., AB CDEF 1234 5678h).

Storage capacities shall be reported in Base-10. IO transfer sizes and offsets shall be reported in Base-2. The associated units and abbreviations used in this specification are:

- 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
- A kibibyte (KiB) is equal to 2^{10} bytes.
- A mebibyte (MiB) is equal to 2^{20} bytes.
- A gibibyte (GiB) is equal to 2^{30} bytes.
- A tebibyte (TiB) is equal to 2^{40} bytes.
- A pebibyte (PiB) is equal to 2^{50} bytes

2.4.2 Pseudo Code Conventions

The specification uses an informal pseudo code to express the test loops. It is important to follow the precedence and ordering information implied by the syntax. In addition to nesting/indentation, the main syntactic construct used is the “For” statement.

A “For” statement typically uses the syntax: For (variable = x, y, z). The interpretation of this construct is that the Test Operator sets the variable to x, then performs all actions specified in the indented section under the “For” statement, then sets the variable to y, and again performs the actions specified, and so on. Sometimes a “For” statement will have an explicit “End For” clause, but not always; in these cases, the end of the For statement’s scope is contextual.

Take the following loop as an example:

```
For (R/W Mix % = 100/0, 95/5, 65/35, 50/50, 35/65, 5/95, 0/100)
    For (Block Size = 1024KiB, 128KiB, 64KiB, 32KiB, 16KiB, 8KiB, 4KiB, 0.5KiB)
        - Execute random IO, per (R/W Mix %, Block Size), for 1 minute
        - Record Ave IOPS(R/W Mix%, Block Size)
```

This loop is executed as follows:

- Set R/W Mix% to 100/0 >>>>> Beginning of Round 1
- Set Block Size to 1024KiB
- Execute random IO…
- Record Ave IOPS…
- Set Block Size to 128KiB
- Execute…
- Record…
- …
- Set Block Size to 0.5KiB
- Execute…
- Record… >>>>> End of Round 1
- Set R/W Mix% to 95/5 >>>>> Beginning of Round 2
- Set Block Size to 1024 KiB
- Execute…
- Record…
- …

3 Key Test Process Concepts

The performance of an SSS device is highly dependent on its prior usage, the pre-test state of the device and test parameters. This section describes key SSS test methodology concepts.

3.1 Steady State

SSS devices that are Fresh Out of the Box, or in an equivalent state, typically exhibit a transient period of elevated performance, which evolves to a stable performance state relative to the workload being applied. This state is referred to as a Steady State (Section 2.1.18).

It is important that the test data be gathered during a time window when the device is in Steady State, for two primary reasons:

1) To ensure that a device's initial performance (FOB or Purged) will not be reported as "typical", since this is transient behavior and not a meaningful indicator of the drive's performance during the bulk of its operating life.

2) To enable Test Operators and reviewers to observe and understand trends. For example, oscillations around an average are "steady" in a sense, but might be a cause for concern.

Steady State may be verified:

- by inspection, after running a number of Rounds and examining the data;
- programmatically, during execution; or
- by any other method, as long as the attainment of Steady State, per Definition 2.1.18, is demonstrated and documented.

Steady State reporting requirements are covered in the respective test sections.

3.2 Purge

The purpose of the Purge process (Definition 2.1.16) is to put the device in a consistent state prior to preconditioning and testing, and to facilitate a clear demonstration of Steady State convergence behavior.

Purge shall be run prior to each preconditioning and testing cycle. If the device under test does not support any kind of Purge method, and the Test Operator chooses to run the PTS, the fact that Purge was not supported/run must be documented in the test report.

The Test Operator may select any valid method of implementing the Purge process, including, but not limited to, the following:

a) ATA: SECURITY ERASE, SANITIZE DEVICE (BLOCK ERASE EXT)
b) SCSI: FORMAT UNIT
c) Vendor specific methods

The Test Operator shall report what method of Purge was used.

3.3 Preconditioning

The goal of preconditioning is to facilitate convergence to Steady State during the test itself.

The SSS PTS defines two (2) Preconditioning steps:

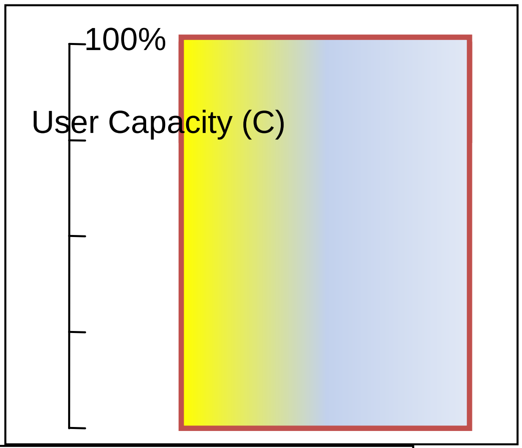
- Workload Independent Preconditioning (WIPC Definition 2.1.23); and
- Workload Dependent Preconditioning (WDPC Definition 2.1.24)

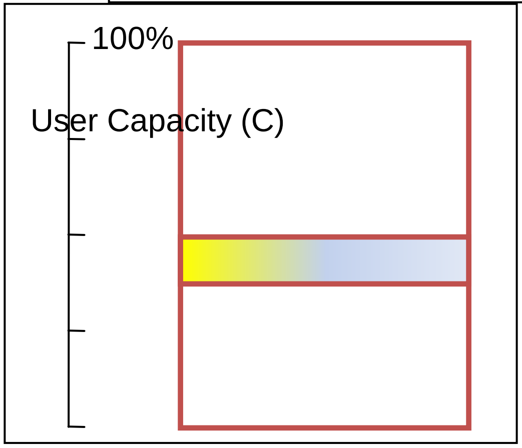
3.4 ActiveRange

It is desirable to be able to test the performance characteristics of workloads which issue IO across a wide range of the LBA space vs. those which issue IO across only a narrow range. To enable this, the SSS Performance Specification defines ActiveRange. (Definition 2.1.1)

The test scripts define required and optional settings for ActiveRange.

The figures below show two examples of ActiveRange.

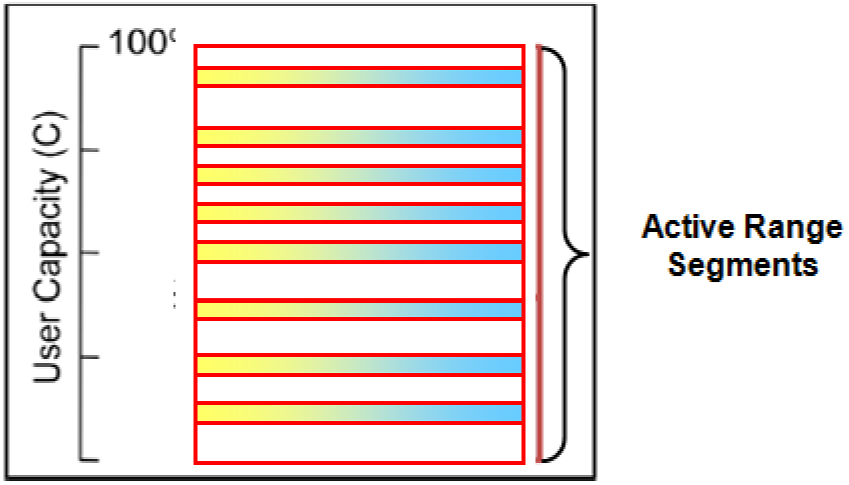




ActiveRange (0:100) **ActiveRange (40:50)**

3.5 ActiveRange Segmentation

The Limited ActivRange as set forth in this specification shall be segmented into segments as defined and prescribed. The following illustration shows the alignment of Random Contiguous Segments.



Random Contiguous Active Range Segments

3.6 Data Patterns

All tests shall be run with a random data pattern. The Test Operator may execute additional runs with non-random data patterns. If non-random data patterns are used, the Test Operator must report the data pattern.

Note: Some SSS devices look for and optimize certain data patterns in the data payloads written to the device. It is not feasible to test for all possible kinds of optimizations, which are vendor specific and often market segment specific. The SSS TWG is still trying to characterize “how random is random enough” with respect to data patterns.

3.7 Multiple Thread Guideline

If the Test Operator wishes to run a test using multiple Threads, it is recommended that OIO/Thread for all Threads be equal, so Total OIO is equal to (OIO/Thread) * (Thread Count). This will enable more direct comparisons.

3.8 Caching

All tests should be run with all volatile write caches disabled. The cache state shall be reported.

4 Overview of Common Test Flow

The tests in the SSS PTS use the same general steps and flow, described below. Test-specific parameter settings, reports, and other requirements are documented in the test sections themselves.

Basic Test Flow:

For (ActiveRange = the specified values)

1) **Purge the device**

   Note: Test Operator may use any values for ActiveRange and Test Parameters for this step; no parameter reporting is required.

2) **Run Workload Independent Preconditioning**

   Note: Test Operator shall use specified ActiveRange ("For ActiveRange ="), but may choose other Test Parameter values to optimize this step, and shall report them.

3) **Run Test (includes Workload Dependent Preconditioning):**

   a) Set Test Parameters (OIO/Thread, Thread Count, Data Pattern, etc.) as specified in the test script.

   b) Run test loop until Steady State reached, or a maximum of 25 Rounds. Accumulate/Record intermediate data, as specified in test, for each Round.

4) **Post process & plot the Rounds data:**

   a) If Steady State is reached by Round x<=25, where the Measurement Window is Round x-4:x, the Test Operator shall:
      i) Plot rounds 1:x per "Steady State Convergence Plot";
      ii) Plot Rounds x-4:x per "Steady State Verification Plot"; and
      iii) Plot Rounds x-4:x per "Measurement Plot".

   b) If Steady State is not reached by Round x<=25, the Test Operator shall either:
      i) Continue at 3b until Steady State reached (x>25), and then report per 4a(i-iii); or
      ii) Stop at Round x and report per 4a(i-iii).

End "For ActiveRange"

The Test Operator may re-run the entire "For ActiveRange" loop with alternate test parameters, which may be optional or required, depending on the test.

End "Basic Test Flow"

Note: Steps (2) and (3) must each be run with no interruptions, and there must be no delay between Step (2) and Step (3), to maintain consistent test conditions for all devices.

Note: With respect to the reports in Step (4):

- The **Steady State Convergence Plot** shows general visual convergence to Steady State by plotting the dependent variable (IOPS, Throughput, etc.) for each Round.
- The **Steady State Verification Plot** shows, via either graph or table, that the device has reached Steady State per definition 2.1.18, by examining dependent variable behavior within the Measurement Window.
- The **Measurement Plot** is not one, but a set of, plots/reports, which summarize the test data in the Measurement Window, for the metric being measured.
- The content of these plots, and other test-specific reporting, is specified in each test.

5 Common Reporting Requirements

The following items, common to all tests, shall be included in the final test report. These items only need to be reported once in the test report. Test-specific report items are defined in the relevant test sections themselves. A sample test report can be found in Informative Annex A – Sample Test Report.

5.1 General

1) Test Date
2) Report Date
3) Test Operator name
4) Auditor name, if applicable
5) Test Specification Version

5.2 Test System Hardware

1) Manufacturer/Model #
2) Mother Board/Model #
3) CPU
4) DRAM
5) Host Bus Adapter
6) Primary Storage
7) Peripherals

5.3 Test System Software

1) Operating System Version
2) File System and Version
3) Test Software

5.4 Device Under Test

1) Manufacturer
2) Model Number
3) Serial Number
4) Firmware Revision
5) User Capacity
6) Interface/Speed
7) Form Factor (e.g. 2.5")
8) Media Type (e.g. MLC NAND Flash)
9) Optional: Other major relevant features (e.g. NCQ, Hot plug, Sanitize support, etc.)

6 Test Tool Guidelines

The SSS PTS is platform (HW/OS/Tool) agnostic. A sample platform is outlined in SW tools used to test SSS devices pursuant to this PTS shall have the ability to:

1) Act as workload stimulus generator as well as data recorder
2) Issue Random and Sequential block level I/O
3) Ability to access contiguous LBA ranges sequentially across test steps
4) Restrict LBA accesses to a particular range, or collection of ranges, within the available user LBA space
5) Set R/W percentage mix %
6) Set Random/Sequential IO mix %
7) Set IO Transfer Size
8) Generate and maintain multiple outstanding IO requests. All test sequence steps shall be executed immediately one after the other. This ensures that drives are not given time to recover between processing steps, unless recovery is the explicit goal of the test.
9) Provide output, or output that can be used to derive, IOPS, MB/s, maximum latency and average response time (latency if OIO=1) within some measurement period.

The random function for generating random LBA #’s during random IO tests shall be:

1) seedable;
2) have an output >= 48-bit; and
3) deliver a uniform random distribution independent of capacity.

7 IOPS Test

For (ActiveRange=100% and ActiveRange= 75%)

For (ActiveAmount=8GB and ActiveAmount=16GB)

1 Purge the device. (Note: Active Range and other Test Parameters are not applicable to Purge step; any values can be used and none need to be reported.)

2 Workload Independent Preconditioning

2.1 Run SEQ Workload Independent Preconditioning

2.1.1 Set and record test conditions:

2.1.1.1 Device write cache = disabled

2.1.1.2 OIO/Thread: Test Operator Choice

2.1.1.3 Thread Count: Test Operator Choice

2.1.1.4 Data Pattern: Required = Random, Optional = Test Operator

2.1.2 Run SEQ Workload Independent Preconditioning - Write 2X User Capacity w/ 128KiB SEQ writes, writing to the entire ActiveRange without LBA restrictions.

2.2 Run RND Workload Independent Preconditioning

2.2.1 Set and record test conditions:

2.2.1.1 Device write cache = disabled

2.2.1.2 OIO/Thread: Test Operator Choice

2.2.1.3 Thread Count: Test Operator Choice

2.2.1.4 Data Pattern: Required = Random, Optional = Test Operator Choice

2.2.2 For (R/W Mix % = 100/0, 95/5, 65/35, 50/50, 35/65, 5/95, 0/100)

2.2.2.1 For (Block Size = 1024KiB, 128KiB, 64KiB, 32KiB, 16KiB, 8KiB, 4KiB, 0.5KiB)

2.2.2.1.1 Execute **random IO**, per (R/W Mix %, Block Size), for 1 minute

2.2.2.1.2 Record Ave IOPS(R/W Mix%, Block Size)

2.2.2.1.3 Use IOPS(R/W Mix% = 0/100, Block Size = 4KiB) to detect Steady State.

2.2.2.1.4 If Steady State is not reached by Round x<=25, then the Test Operator shall either continue running the test until Steady State is reached, or may stop the test at Round x. The Measurement Window to determine WIPC termination is defined as Round x-4 to Round x.

2.2.2.2 End (For Block Size) Loop

2.2.3 End (For R/W Mix %) Loop

3 Run Workload Dependent Preconditioning and Test stimulus. Set test parameters and record for later reporting

3.1 Set and record test conditions:

3.1.1 Device write cache = Disabled

3.1.2 OIO/Thread: Test Operator Choice

3.1.3 Thread Count: Test Operator Choice

3.1.4 Data Pattern: Required= Random, Optional = Test Operator Choice.

3.1.5 Active Range Segmentation Configuration:

3.1.5.1 Number of ActiveRange Segments = 2048

3.1.5.2 ActiveRange Segment Size = ActiveRange Amount divided by 2048

3.1.5.3 Distribution of ActiveRange Segments: randomly distributed within the entire ActiveRange. The ActiveRange Segments are not allowed to touch.

3.2 Run the following test loop until Steady State is reached, or maximum of 25 Rounds:

3.2.1 For (R/W Mix % = 100/0, 95/5, 65/35, 50/50, 35/65, 5/95, 0/100)

3.2.1.1 For (Block Size = 1024KiB, 128KiB, 64KiB, 32KiB, 16KiB, 8KiB, 4KiB, 0.5KiB)

3.2.1.1.1 Execute **random IO**, per (R/W Mix %, Block Size), for 1 minute

3.2.1.1.2 Record Ave IOPS(R/W Mix%, Block Size)

3.2.1.1.3 Use IOPS(R/W Mix% = 0/100, Block Size = 4KiB) to detect Steady State.

3.2.1.1.4 If Steady State is not reached by Round x<=25, then the Test Operator shall either continue running the test until Steady State is reached, or may stop the test at Round x. The Measurement Window is defined as Round x-4 to Round x.

3.2.1.2 End "For Block Size" Loop

3.2.2 End "For R/W Mix%" Loop

4 Process and plot the accumulated Rounds data, per report guidelines in next section, based on current values of ActiveRange, OIO/Thread, etc.

End (For ActiveAmount) loop

End (For ActiveRange) loop

The accumulated Rounds data at the end of step (4) will consist of x matrices, one for each Round. Each table entry is Ave IOPS(R/W Mix%, Block Size) recorded for the 1 minute duration of Round x:

Ave IOPS - Round x

Block Size	*Read/Write Mix %*						
	0/100	*5/95*	*35/65*	*50/50*	*65/35*	*95/5*	*100/0*
0.5 KiB	$AI_{1,1}$	$AI_{1,2}$	$AI_{1,3}$	$AI_{1,4}$	$AI_{1,5}$	$AI_{1,6}$	$AI_{1,7}$
4 KiB	$AI_{2,1}$	$AI_{2,2}$	$AI_{2,3}$	$AI_{2,4}$	$AI_{2,5}$	$AI_{2,6}$	$AI_{2,7}$
8 KiB	$AI_{3,1}$	$AI_{3,2}$	$AI_{3,3}$	$AI_{3,4}$	$AI_{3,5}$	$AI_{3,6}$	$AI_{3,7}$
16 KiB	$AI_{4,1}$	$AI_{4,2}$	$AI_{4,3}$	$AI_{4,4}$	$AI_{4,5}$	$AI_{4,6}$	$AI_{4,7}$
32 KiB	$AI_{5,1}$	$AI_{5,2}$	$AI_{5,3}$	$AI_{5,4}$	$AI_{5,5}$	$AI_{5,6}$	$AI_{5,7}$
64 KiB	$AI_{6,1}$	$AI_{6,2}$	$AI_{6,3}$	$AI_{6,4}$	$AI_{6,5}$	$AI_{6,6}$	$AI_{6,7}$
128 KiB	$AI_{7,1}$	$AI_{7,2}$	$AI_{7,3}$	$AI_{7,4}$	$AI_{7,5}$	$AI_{7,6}$	$AI_{7,7}$
1024 KiB	$AI_{8,1}$	$AI_{8,2}$	$AI_{8,3}$	$AI_{8,4}$	$AI_{8,5}$	$AI_{8,6}$	$AI_{8,7}$

Note: The Test Operator may structure the intermediate matrices differently, per Test Operator convenience and data gathering methods.

After generating report data for current parameter values, the Test Operator may re-run “For ActiveRange” loop with alternate Test Parameters, if specified in (3), and may also run the entire test again with an optional value (or values) of ActiveRange.

7.1 Test Specific Reporting for IOPS Test

The following sub-sections list the reporting requirements specific to the IOPS test. Reporting requirements common to all tests are documented in Section 5, Common Reporting Requirements. See also Informative Annex A – Sample Test Report.

If Steady State was reached at Round x, the Test Operator shall:

- Plot rounds 1:x per "Throughput Steady State Convergence Plot";
- Plot Rounds x-4:x per "Throughput Steady State Verification Plot"; and
- Plot Rounds x-4:x per "Throughput Measurement Plot."

If Steady State was not reached then the Test Operator may report results per above, picking the last Round run as Round x. **In the case where Steady State was not reached, the Test Operator must state this fact in the final report.**

7.1.1 Purge

The Test Operator shall report the method used to run the Purge operation.

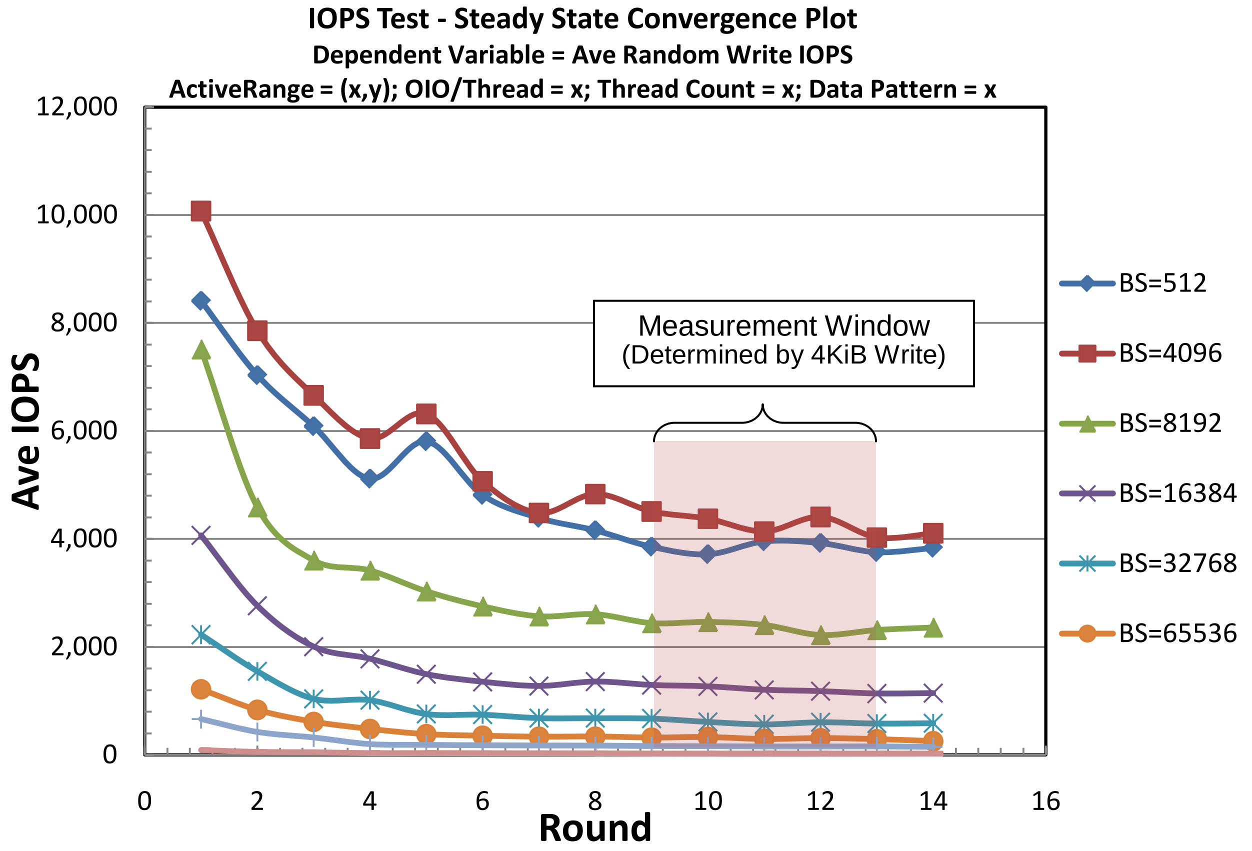
7.1.2 Workload Independent Preconditioning

The Test Operator shall report the following regarding the Workload Independent Preconditioning step: 1) ActiveRange; 2) OIO/Thread; 3) Thread Count; and 4: Data Pattern

7.1.3 Steady State Convergence

The Test Operator shall generate a Steady State Convergence plot as follows:

- **y-axis:** IOPS | **x-axis:** Round Number
- **Data Set to Plot:** Ave IOPS (R/W Mix% = 0/100, All Block Sizes), per Round
- **Additional Info on Plot:**
  - Designation of calculated Measurement Window
  - Test Parameters: ActiveRange, OIO/Thread, Thread Count, Data Pattern



7.1.4 Steady State Verification

The Test Operator shall document the following for Steady State Verification, using Ave 4KiB Random Write IOPS as the dependent variable:

IOPS Test - Steady State Verification
Dependent Variable = Ave 4KiB Random Write IOPS
ActiveRange = (x,y); OIO/Thread = x; Thread Count = x; Data Pattern = x

- Measurement Window: Start __; End __
- Ave. value in Measurement Window: __
- Calculated allowed range in Measurement Window (+-10% of Ave.): Max__; Min __
- Measured range in Measurement Window: Max __; Min __ (pass/fail)
- Slope of best linear fit in Measurement Window (must be <= 10%): __% (pass/fail)
- Correlation coefficient for best linear fit: __

Note: The Test Operator may optionally include a plot showing Steady State verification. See Figure 7-1 - Steady State Verification Example Plot.

<xyz> Test - Steady State Verification Plot
Dependent Variable (Ave 4K Rnd Write IOPS)
ActiveRange = (x,y); OIO/Thread = x; Thread Count = x; Data Pattern = x

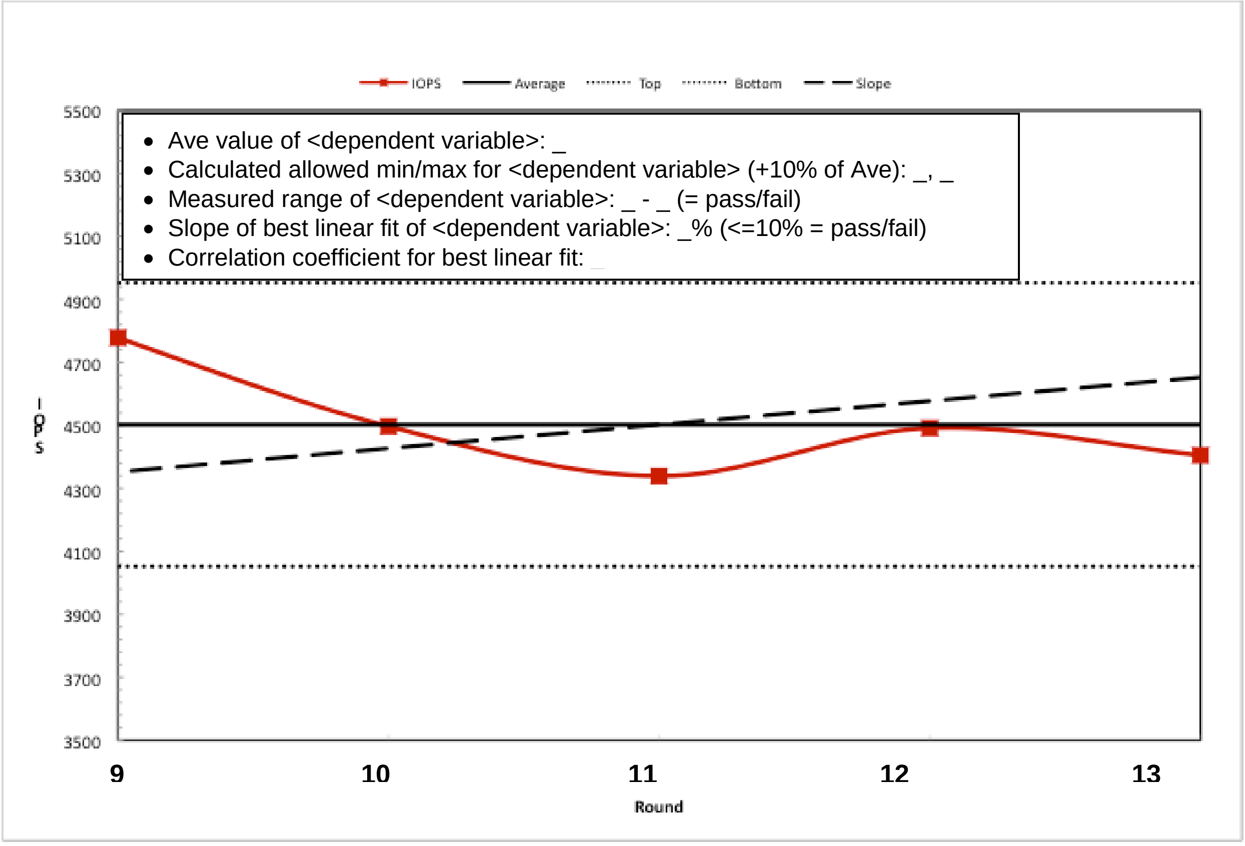


Figure 7-1 - Steady State Verification Example Plot

7.1.5 Measurement

The results for the test must be reported in the following series of tables and graphs which record and report the data from the Steady State Measurement Window.

7.1.5.1 Measurement Window Summary Data Table

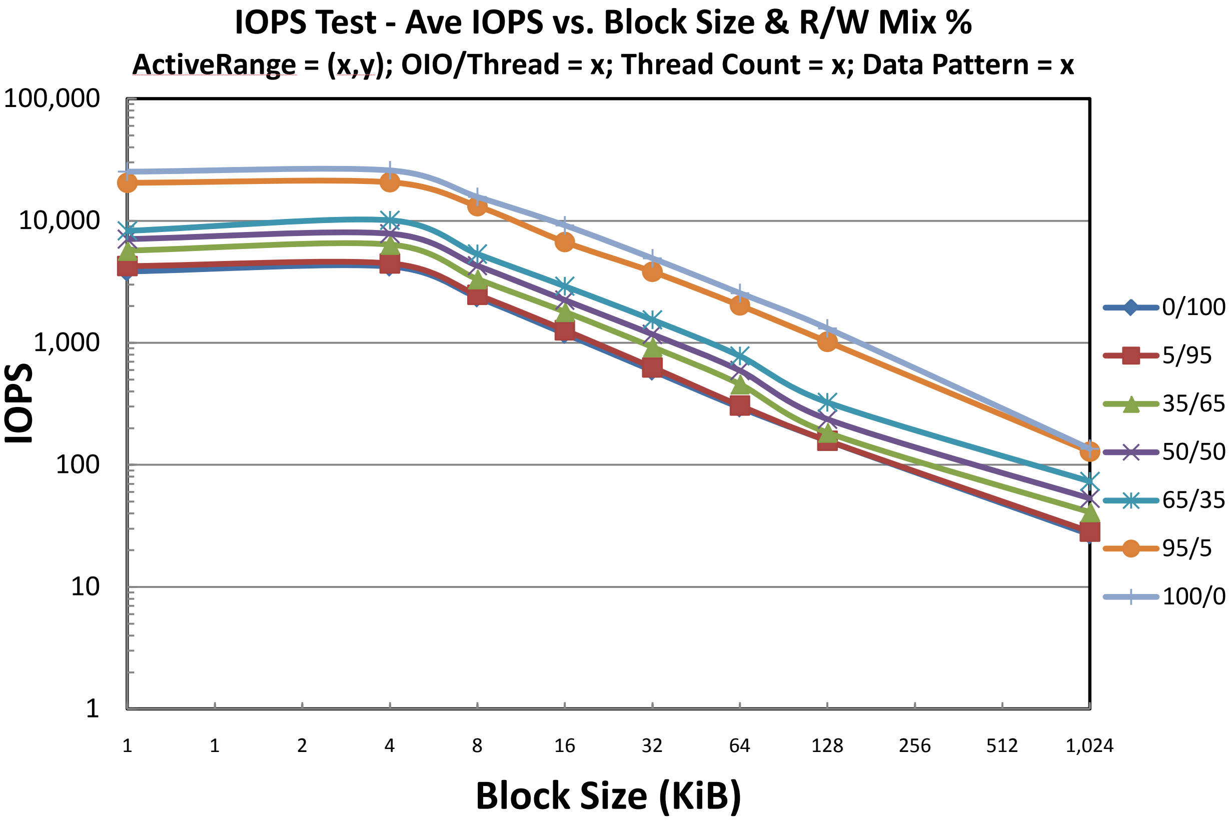
Average IOPS vs. Block Size and R/W Mix %
ActiveRange = (x,y); OIO/Thread = x; Thread Count = x; Data Pattern = x

	Read / Write Mix %						
Block Size	0/100	95/5	65/35	50/50	35/65	5/95	100/0
0.5 KiB	4104	4368	5941	7243	8913	20550	26015
4 KiB	4502	4671	6647	8260	10593	22773	25819
8 KiB	2425	2547	3606	4413	5688	13132	16051
16 KiB	1273	1334	1901	2319	3023	6663	9402
32 KiB	656	673	982	1266	1574	3995	5036
64 KiB	315	346	526	643	804	2112	2629
128 KiB	167	169	207	269	345	1051	1341
1024 KiB	29	30	44	58	74	131	142

Each entry in the table is the average of the values in the data series *Average_IOPS(x,y)*, recorded in the per-Round IOPS matrices within the Measurement Window, for the selected (R/W Mix%, Block Size) pair.

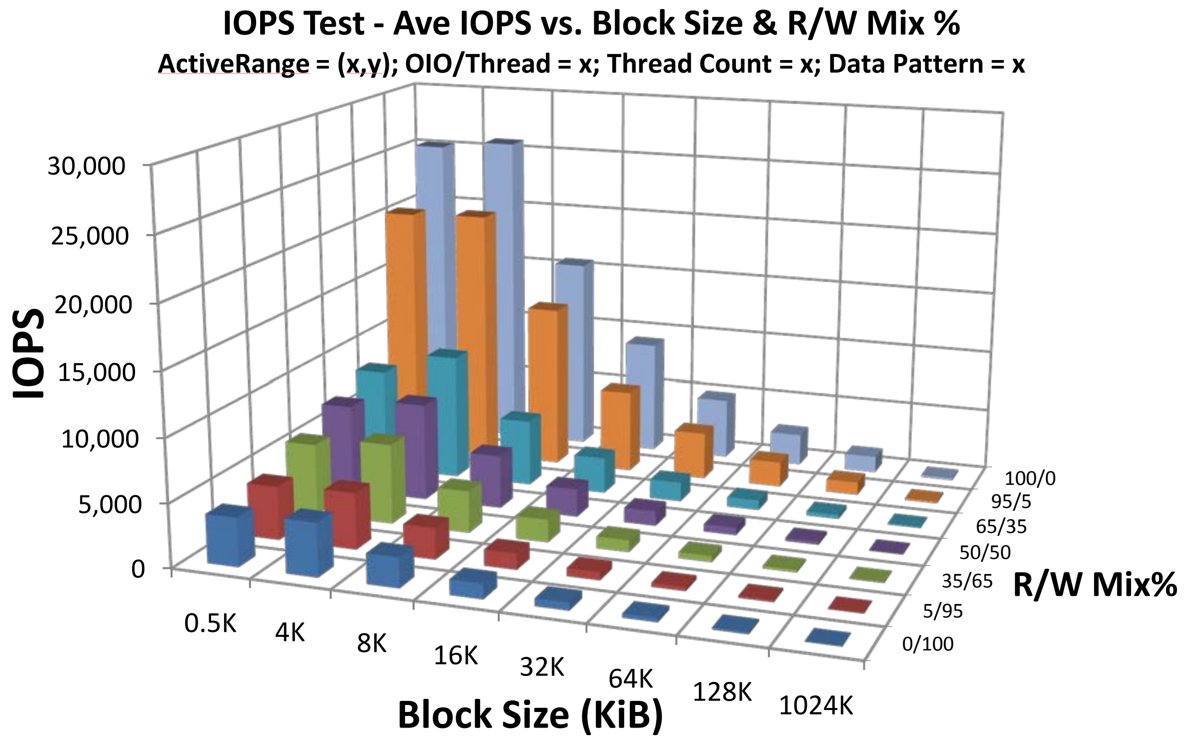
7.1.5.2 Measurement Plot – 2D

- **y-axis:** IOPS | **x-axis:** Block Size (KiB)
- **Plot Style:** 2D Line Plot
- **Additional Info:**
  - Test Parameters: ActiveRange, OIO/Thread, Thread Count, Data Pattern

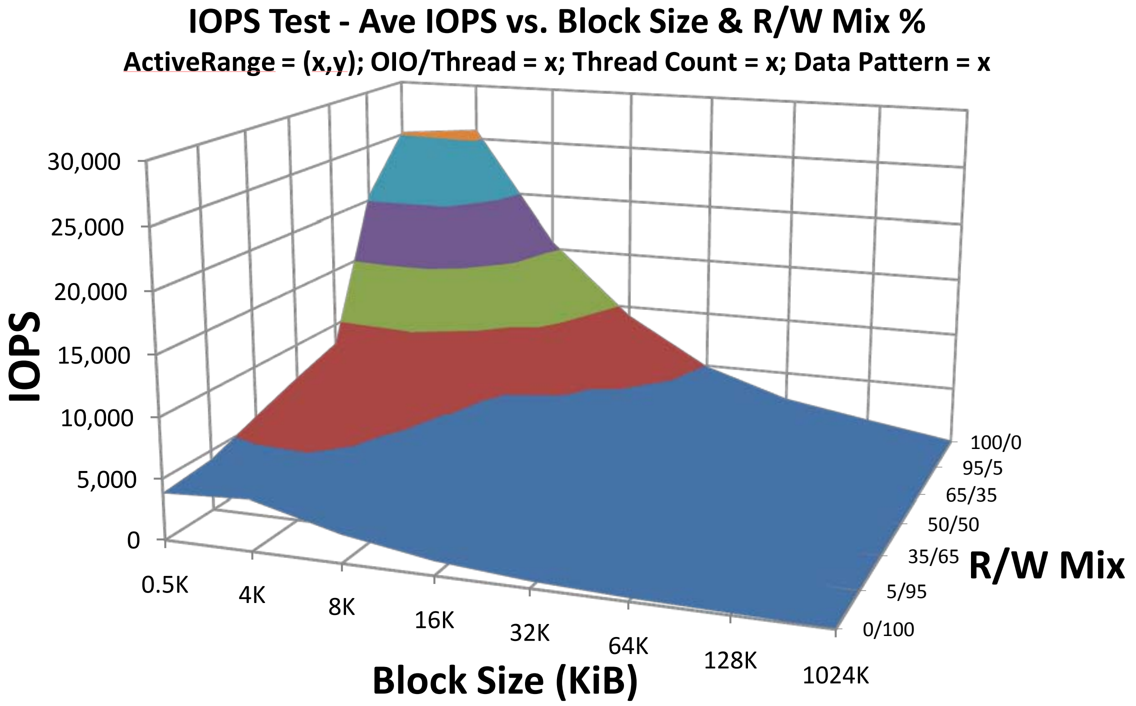


7.1.5.3 Measurement Plot – 3D

- **x-axis:** Block Size | **y-axis:** IOPS | **z-axis:** R/W Mix %
- **Plot Style:** 3D Bar and/or 3D Surface Plot
- **Additional Info:**
    - List the Test Parameters: ActiveRange, OIO/Thread, Thread Count, Data Pattern



AND/OR



8 Throughput Test

```
For (ActiveRange=100% and ActiveRange= 75%)

    For (ActiveAmount=8GB and ActiveAmount=16GB)

        1. Purge the device. (Note: Active Range and other Test Parameters are not applicable
           to Purge step; any values can be used and none need to be reported.)

        2. Workload Independent Preconditioning

            2.1. Set and record parameters for later reporting.
                2.1.1. Write cache: disabled
                2.1.2. Thread Count: Test Operator Choice
                2.1.3. OIO/Thread: Test Operator Choice
                2.1.4. Data Pattern: Required = Random, Optional = Test Operator
                       Choice
            2.2. Run SEQ Workload Independent Preconditioning - Write 2X User Capacity w/
                 1024KiB SEQ writes to the entire ActiveRange without any LBA restrictions

        3. Run Workload Dependent Preconditioning and Test Stimulus

            3.1. Set parameters and record for later reporting
                3.1.1. Write cache: disabled
                3.1.2. Thread Count: Test Operator Choice
                3.1.3. OIO/Thread: Test Operator Choice
                3.1.4. Data Pattern: Required = Random, Optional = Test Operator
                       Choice
                3.1.5. Active Range Segmentation Configuration:
                    3.1.5.1. Number of ActiveRange Segments = 2048
                    3.1.5.2. ActiveRange Segment Size = ActiveRange Amount divided
                             by 2048
                    3.1.5.3. Distribution of ActiveRange Segments: randomly
                             distributed within the entire ActiveRange. The
                             ActiveRange Segments are not allowed to touch.
            3.2. Run the following until Steady State is reached, or maximum of 25 Rounds
                3.2.1. For (R/W Mix % = 100/0, 0/100)

                    3.2.1.1. Execute sequential IO, per (R/W Mix%, Block Size), for 1
                             minute
                    3.2.1.2. Record Ave MB/s (R/W Mix%, Block Size)
                    3.2.1.3. Use Ave MB/s(RW Mix%, Block Size) to detect Steady
                             State.
                    3.2.1.4. If Steady State is not reached by Round x<=25, then the
                             Test Operator shall either continue running the test until
                             Steady State is reached, or may stop the test at Round x.
                             The Measurement Window is defined as Round x-4 to
                             Round x.
                    3.2.1.5. Note that the sequential accesses shall be continuous and
                             use the entire ActiveRange between test steps
                3.2.2. End (For R/W Mix%) Loop
```

4. Process and plot the accumulated Rounds data, per report guidelines in next section, based on current values of ActiveRange etc.

End (For ActiveAmount) loop

End (For ActiveRange) loop

The accumulated Rounds data at the end of the (For Block Size) loop will consist of x matrices, one for each Block Size. Each table entry is Ave MB/s(R/W Mix%, Block Size) recorded for the 1 minute duration of Round x:

Ave MB/s – Block Size = ()KiB

Round #	*Read/Write Mix %*	
	0/100	*100/0*
1	*Ave MB/S* $_{1,1}$	*Ave MB/S* $_{1,2}$
2	*Ave MB/S* $_{2,1}$	*Ave MB/S* $_{2,2}$
…	*Ave MB/S* $_{4,1}$	*Ave MB/S* $_{4,2}$
x	*Ave MB/S* $_{5,1}$	*Ave MB/S* $_{5,2}$

Note: The Test Operator may structure the intermediate matrices differently, per Test Operator convenience and data gathering methods.

End (For ActiveRange) loop

After generating report data for current parameter values, the Test Operator may re-run "For ActiveRange" loop with alternate Test Parameters, if specified in (2), and may also run the entire test again with an optional value (or values) of ActiveRange.

Note: It is important to adhere to the nesting of the loops. For each Block Size, the entire [purge | precondition | set parameters | run test] loop is executed. This was done to avoid creating unwanted performance interactions by interspersing block sizes, which would not happen during an actual Sequential IO sequence.

8.1 Test Specific Reporting for Throughput Test

The following sub-sections list the reporting requirements specific to the Throughput test. Reporting requirements common to all tests are documented in Section 5, Common Reporting Requirements. See also Informative Annex A – Sample Test Report.

If Steady State was reached at Round x, the Test Operator shall:

- Plot rounds 1:x per “Throughput Steady State Convergence Plot”;
- Plot Rounds x-4:x per “Throughput Steady State Verification Plot”; and
- Plot Rounds x-4:x per “Throughput Measurement Plot.”

If Steady State was not reached then the Test Operator may report results per above, picking the last Round run as Round x. **In the case where Steady State was not reached, the Test Operator must state this fact in the final report.**

8.1.1 Purge Report

The Test Operator shall report the method used to run the Purge operation.

8.1.2 Workload Independent Preconditioning Report

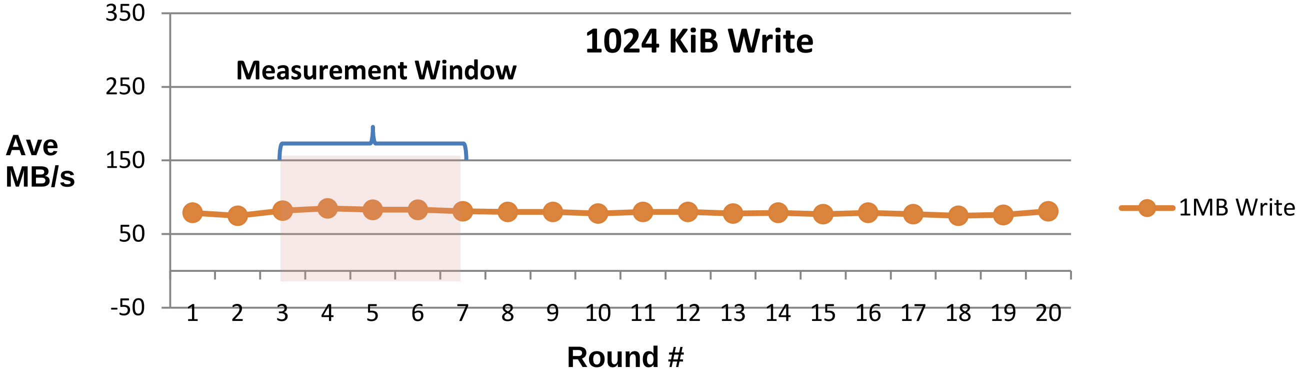
The Test Operator shall report the following regarding the Workload Independent Preconditioning step: 1) ActiveRange; 2) OIO/Thread; 3) Thread Count; and 4: Data Pattern

8.1.3 Steady State Convergence Report

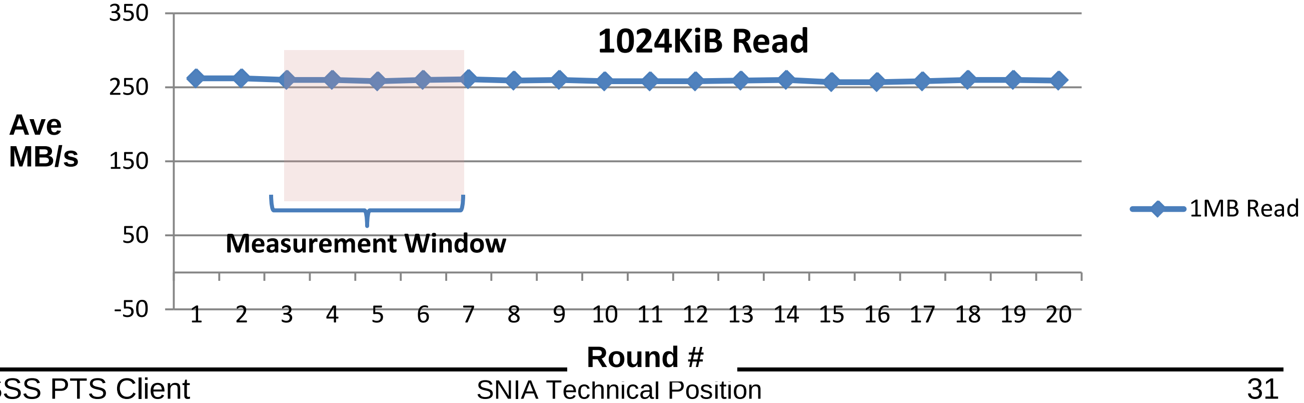
The Test Operator shall generate Write and Read Steady State Convergence plots, as follows:

- **Data Set to Plot:** Ave MB/s (R/W Mix% = 0/100 **AND** 100/0, All Block Sizes), per Round
- **Additional Info on Plot:**
  - Designation of calculated Measurement Window
  - Test Parameters: ActiveRange, OIO/Thread, Thread Count, Data Pattern

Steady State Convergence – Write Throughput
Dependent Variable = Ave Sequential Write MB/s
ActiveRange = (x,y); OIO/Thread = x; Thread Count = x; Data Pattern = x



Steady State Convergence – Read Throughput
Dependent Variable = Ave Sequential Read MB/s
ActiveRange = (x,y); OIO/Thread = x; Thread Count = x; Data Pattern = x



8.1.4 Steady State Verification Report

The Test Operator shall document the following for Steady State Verification, using *Ave 1024KiB Sequential Write MB/s* as the dependent variable:

Throughput Test - Steady State Verification
Dependent Variable = Ave 1024KiB Sequential Write MB/s
ActiveRange = (x,y); OIO/Thread = x; Thread Count = x; Data Pattern = x

- Measurement Window: Start __; End __
- Ave. value in Measurement Window: __
- Calculated allowed range in Measurement Window (+-10% of Ave.): Max__; Min __
- Measured range in Measurement Window: Max __; Min __ (pass/fail)
- Slope of best linear fit in Measurement Window (must be <= 10%): __% (pass/fail)
- Correlation coefficient for best linear fit: __

Note: The Test Operator may optionally include a plot showing Steady State verification. See Figure 7-1 - Steady State Verification Example Plot.

8.1.5 Measurement Window Report

The results for the test must be reported in the following series of tables and graphs which record and report the data from the Steady State Measurement Window.

8.1.5.1 Measurement Window Summary Data Table

Average MB/s vs. Block Size and R/W Mix %
ActiveRange = (x,y); OIO/Thread = x; Thread Count = x; Data Pattern = x

Block Size	Read	Write
1024 KiB	260 MB/s	83 MB/s

Each entry in the table is the average of the values in the data series *Average_MB/s(x,y)*, recorded in the per-Round MB/s matrices within the Measurement Window, for the selected (R/W Mix%, Block Size) pair.

There are no required Measurement Plots for the Throughput test.

9 Latency Test

For (ActiveRange=100% and ActiveRange= 75%)

For (ActiveAmount=8GB and ActiveAmount=16GB)

1. Purge the device. (Note: Active Range and other Test Parameters are not applicable to Purge step; any values can be used and none need to be reported.)

2. Workload Independent Preconditioning

2.1. Run SEQ Workload Independent Preconditioning
2.1.1. Set and record test conditions:
2.1.1.1. Device write cache = disabled
2.1.1.2. OIO/Thread: Test Operator Choice
2.1.1.3. Thread Count: Test Operator Choice
2.1.1.4. Data Pattern: Required = Random, Optional = Test Operator Choice
2.1.2. Run SEQ Workload Independent Preconditioning - Write 2X User Capacity w/ 128KiB sequential writes, writing to the entire ActiveRange without other LBA restrictions.
2.2. Run RND WIPC –
2.2.1. Set and record test parameters:
2.2.1.1. Device write cache = disabled
2.2.1.2. Thread Count: 1
2.2.1.3. OIO/Thread: 1
2.2.1.4. Data Pattern: Required = Random, Optional = Test Operator Choice
2.2.2. For (R/W% = 100/0, 65/35, 0/100)
2.2.2.1. For (Block Size = 8KiB, 4KiB, 0.5KiB)
2.2.2.1.1. Execute random IO per (R/W%, Block Size), for 1 minute
2.2.2.1.2. Record Max and Ave Latency (R/W%, Block Size)
2.2.2.1.3. Use Ave Latency (R/W Mix%=0/100, Block Size=4KiB) to detect Steady State.
2.2.2.1.4. If Steady State is not reached by Round x<=25, then the Test Operator shall either continue running the test until Steady State is reached, or may stop the test at Round x. The Measurement Window to determine WIPC termination is defined as Round x-4 to Round x
2.2.2.2. End (For Block Size) Loop
2.2.3. End (For R/W%) Loop

3. Run Workload Dependent Preconditioning and Test stimulus

3.1. Set test parameters and record for later reporting.
3.1.1. Device write cache = disabled
3.1.2. OIO/Thread: Test Operator Choice
3.1.3. Thread Count: Test Operator Choice
3.1.4. Data Pattern: Required = Random, Optional = Test Operator Choice
3.1.5. Active Range Segmentation Configuration:
3.1.5.1. Number of ActiveRange Segments = 2048
3.1.5.2. ActiveRange Segment Size = ActiveRange Amount divided by 2048

3.1.5.3. Distribution of ActiveRange Segments: randomly distributed within the entire ActiveRange. The ActiveRange Segments are not allowed to touch.

3.2. Run the following test loop until Steady State is reached, or maximum of 25 Rounds:

3.2.1. For (R/W Mix % = 100/0, 65/35, 0/100)

3.2.1.1. For (Block Size = 0.5 KiB, 4KiB, 8KiB)

3.2.1.1.1. Execute **random IO**, per (R/W Mix %, Block Size), for 1 minute

3.2.1.1.2. Record Max and Ave Latency (R/W Mix%, Block Size)

3.2.1.1.3. Use AVE Latency(R//W Mix% = 0/100, Block Size = 4KiB) to detect Steady State.

3.2.1.1.4. If Steady State is not reached by Round x<=25, then the Test Operator shall either continue running the test until Steady State is reached, or may stop the test at Round x. The Measurement Window is defined as Round x-4 to Round x.

3.2.1.2. End (For Block Size) Loop

3.2.2. End (For R/W Mix %) Loop

End (For ActiveAmount) Loop

End (For ActiveRange) Loop

4. Process and plot the accumulated Rounds data, per report guidelines in next section, based on current values of ActiveRange etc.

The accumulated Rounds data at the end of step (4) will consist of x matrices, one for each Round. Each table entry is either Max or Ave Latency(R/W Mix%, Block Size), recorded for the 1 minute duration of Round x:

Latency Test – Raw Data – Round x Max Latency (ms)			
Block Size	Read/Write Mix %		
	0/100	65/35	100/0
0.5 KiB	$PL_{1,1}$	$PL_{1,2}$	$PL_{1,3}$
4 KiB	$PL_{2,1}$	$PL_{2,2}$	$PL_{2,3}$
8 KiB	$PL_{3,1}$	$PL_{3,2}$	$PL_{3,3}$

Latency Test – Raw Data – Round x Average Latency (ms)			
Block Size	Read/Write Mix%		
	0/100	65/35	100/0
0.5 KiB	$AL_{1,1}$	$AL_{1,2}$	$AL_{1,3}$
4 KiB	$AL_{2,1}$	$AL_{2,2}$	$AL_{2,3}$
8 KiB	$AL_{3,1}$	$AL_{3,2}$	$AL_{3,3}$

Note: The Test Operator may structure the intermediate matrices differently, per Test Operator convenience and data gathering methods.

End “For ActiveRange” loop

After generating report data for current parameter values, the Test Operator may re-run “For ActiveRange” loop with alternate Test Parameters, if specified in (3), and may also run the entire test again with an optional value (or values) of ActiveRange.

9.1 Test Specific Reporting for Latency Test

The following sub-sections list the reporting requirements specific to the Latency test. Reporting requirements common to all tests are documented in Section 5, Common Reporting Requirements. See also Informative Annex A – Sample Test Report.

If Steady State was reached at Round x, the Test Operator shall:

- Plot rounds 1:x per “Latency Test Steady State Convergence Plot”;
- Plot Rounds x-4:x per “Latency Test Throughput Steady State Verification Plot”; and
- Plot Rounds x-4:x per “Latency Test Measurement Plot.”

If Steady State was not reached then the Test Operator may report results per above, picking the last Round run as Round x. **In the case where Steady State was not reached, the Test Operator must state this fact in the final report.**

9.1.1 Purge

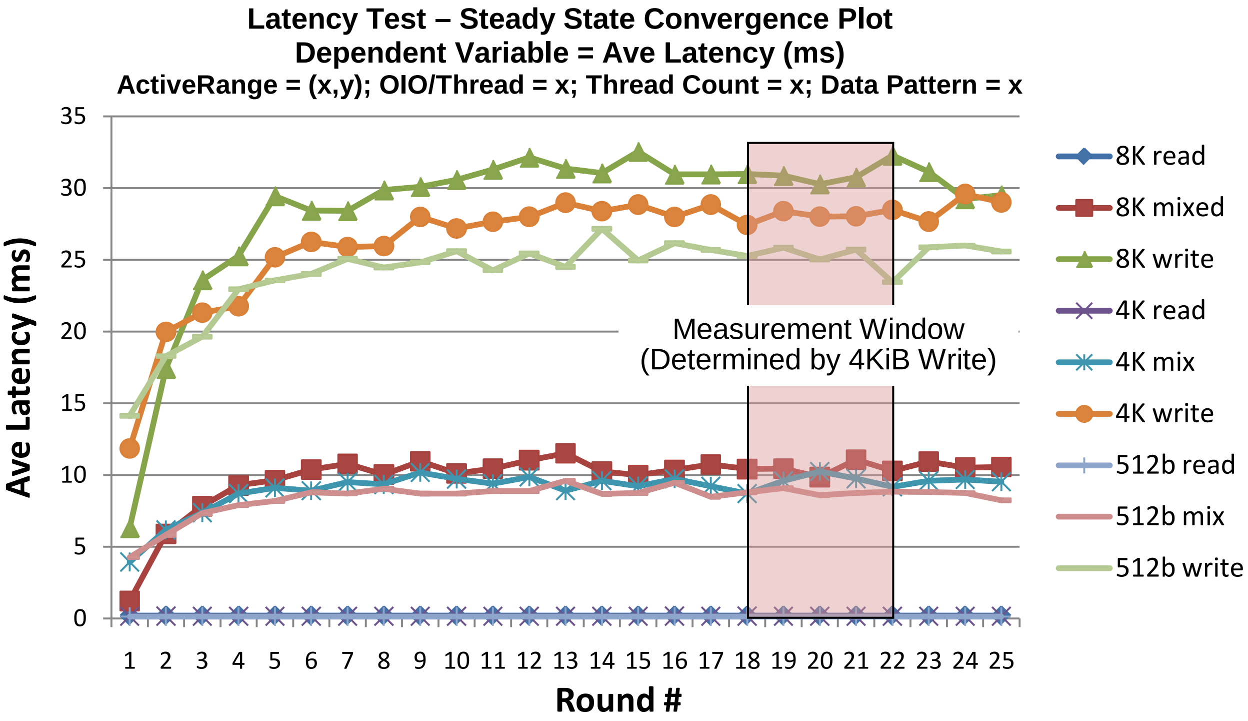
The Test Operator shall report the method used to run the Purge operation.

9.1.2 Workload Independent Preconditioning

The Test Operator shall report the following regarding the Workload Independent Preconditioning step: 1) ActiveRange; 2) OIO/Thread; 3) Thread Count; and 4: Data Pattern

9.1.3 Steady State Convergence

The Test Operator shall generate Steady State Convergence plots, as follows:



9.1.4 Steady State Verification Report

The Test Operator shall document the following for Steady State Verification, using *Ave 4KiB Random Write Latency (ms)* as the dependent variable:

Latency Test - Steady State Verification
Dependent Variable = Ave 4KiB Random Write Latency (ms)
ActiveRange = (x,y); OIO/Thread = x; Thread Count = x; Data Pattern = x

- Measurement Window: Start __; End __
- Ave. value in Measurement Window: __
- Calculated allowed range in Measurement Window (+-10% of Ave.): Max__; Min __
- Measured range in Measurement Window: Max __; Min __ (pass/fail)
- Slope of best linear fit in Measurement Window (must be <= 10%): __% (pass/fail)
- Correlation coefficient for best linear fit: __

Note: The Test Operator may optionally include a plot showing Steady State verification. See Figure 7-1 - Steady State Verification Example Plot.

9.1.5 Measurement Report

The results for the test must be reported in the following series of tables and graphs which record and report the data from the Steady State Measurement Window.

9.1.5.1 Measurement Window Summary Data Table

Ave and Max Latency vs. Block Size and R/W Mix %
ActiveRange = (x,y); OIO/Thread = x; Thread Count = x; Data Pattern = x

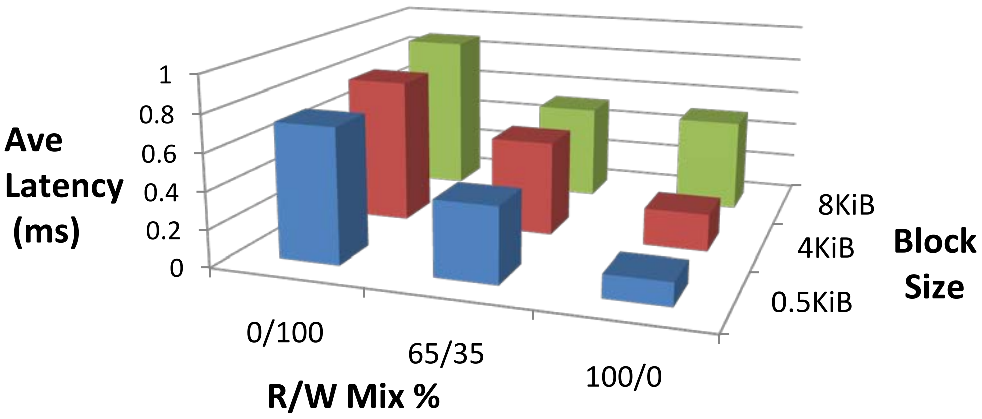
Average Latency (ms)				Maximum Latency (ms)			
Block Size	RW=0/100	RW=65/35	RW=100/0	Block Size	RW=0/100	RW=65/35	RW=100/0
0.5 KiB	0.726	0.402	0.123	0.5 KiB	31.9	38.9	8.4
4 KiB	0.786	0.514	0.201	4 KiB	33.6	42.2	35.7
8 KiB	0.876	0.522	0.507	8 KiB	34.0	36.7	38.2

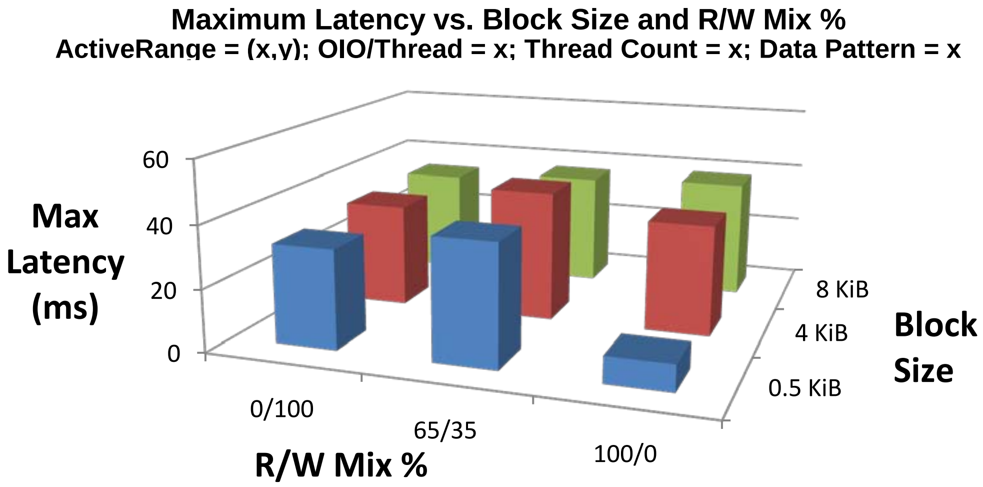
- Each entry in the Average table is the average of the values in the data series ***Average_Latency(x,y)***, recorded in the per-Round matrices within the Measurement Window, for the selected (R/W Mix%, Block Size) pair.

- Each entry in the Maximum table is the maximum value from the data series ***Maximum_Latency(x,y)***, recorded in the per-Round matrices within the Measurement Window, for the selected (R/W Mix%, Block Size) pair.

9.1.5.2 Measurement Plots

Average Latency vs. Block Size and R/W Mix %
ActiveRange = (x,y); OIO/Thread = x; Thread Count = x; Data Pattern = x




Maximum Latency vs. Block Size and R/W Mix %
ActiveRange = (x,y); OIO/Thread = x; Thread Count = x; Data Pattern = x
Max Latency (ms)
60
40
20
0
0/100
65/35
100/0
R/W Mix %
8 KiB
4 KiB
0.5 KiB
Block Size

10 Informative Annex A – Sample Test Report

The following is an Informative example of a test report for the IOPS test.

Informative Annex A – Sample Test Report
Summary Report Page

SNIA SSS PTS - Client	**Rev.**	**0.97**
	Pg. 1	of 5

Device Under Test (DUT)	**ABC Co.**	**SSS TWG PTS Summary Report**		SNIA
Model No.:	ABC123	Test Specification:	Test Run Date:	Apr 3-23 2010
Form Factor:	2.5”		Report Date:	June 01, 2010
NAND Capacity:	256 GB SLC	SNIA SSS TWG PTS v1.0	Test Sponsor:	Calypso Systems
DUT Interface:	SATAII, SAS HBA		Auditor Name:	N/A

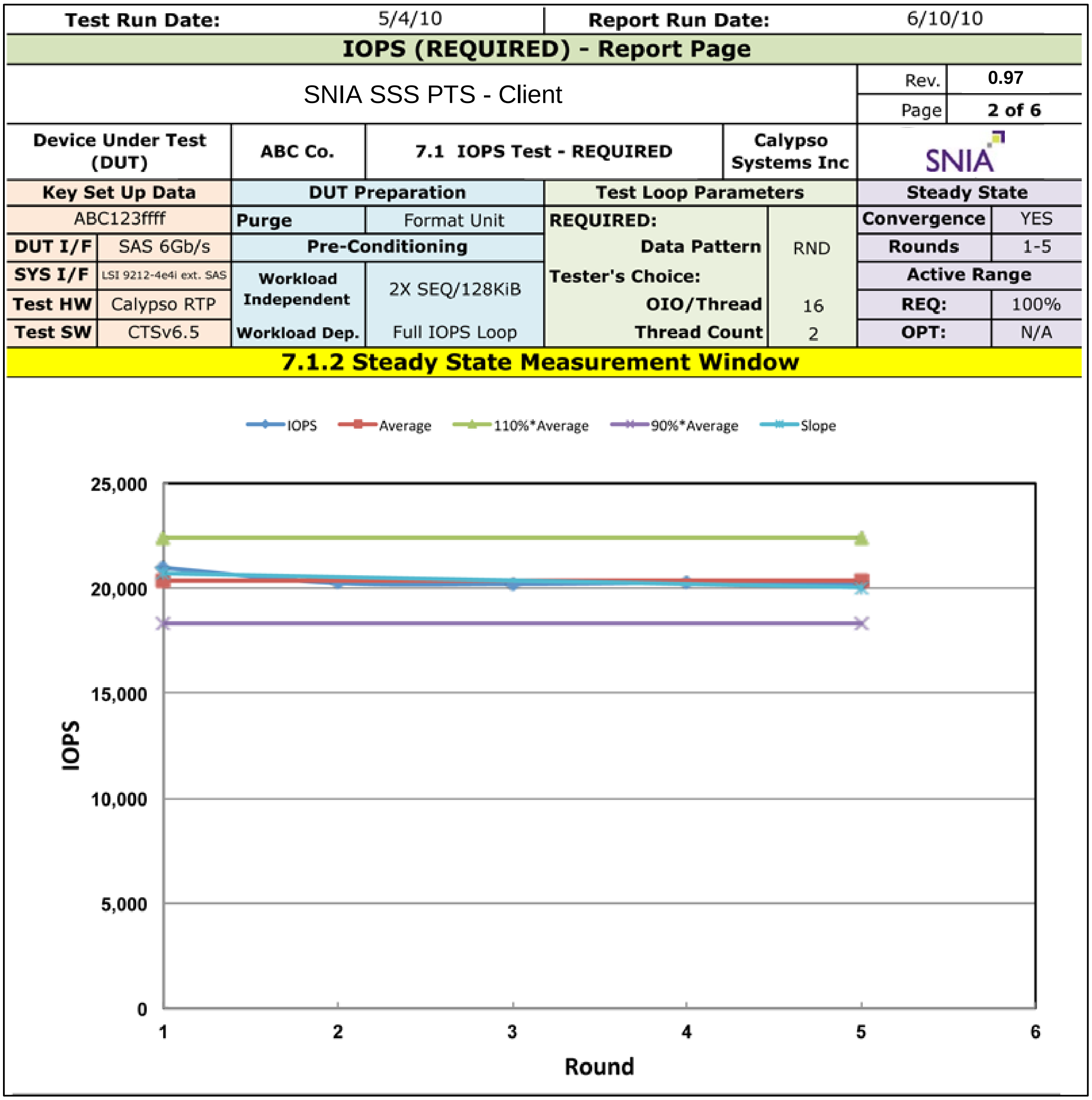
Testing Summary: Tests Run

Test		**Preparation**			**Test Loop Parameters**	
		Purge Type	Workload Independent Preconditioning	Active Range %	QD / TC / OIO	Data Pattern
7.1	IOPS	Format Unit	2X 128K SEQ	100%	QD 16 / TC 4 / OIO 64	RND
7.2	IOPS AR = 10%	Format Unit	2X 128K SEQ	10%	QD 16 / TC 4 / OIO 64	RND
7.3	IOPS File Data	Format Unit	2X 128K SEQ	100%	QD 16 / TC 4 / OIO 64	NON RND File as Data Pattern

General Device Description

Device Under Test (DUT)		System Hardware Configuration		System Software Configuration	
Manufacturer	ABC Co.	System Mfgr	Calypso Systems, Inc.	Operating Sys	Linux CentOS 5.4
Model No.	ABC123	Model No.	RTP 2.0		
Serial No.	123.xxx.ffff	Motherboard	Intel 5520HC	Test SW Tool(s)	Calypso CTSv6.5
Firmware Rev No.	Ffff.hhhh.abc.123	Chasis	Intel SC5560DP		
User Capacity	256 GB	CPU Type	Intel 3.2GHz W5580	Other SWTool(s)	
Interface/Speed	6Gb/s SATAII	No. CPUx	Single		
Form Factor	2.5”	DRAM Type	1333MHz DDR3 ECC		
Media Type	MLC	DRAM Amt	12 GB		
Major Features:		DUT I/F	SAS HBA		
NCQ:	YES	SAS HBA	LSI 6Gb/s 9212-4e4i		
Hot Plug:	YES	SATAII	IC10HR		
Sanitize Support:	NO	PCI-e	Gen 2 (8) lane		
Other 1:		Boot HDD	160 GB 7200RPM		
Other 2:		Optical Drive			

xxxx



Test Run Date:	5/4/10	Report Run Date:	6/10/10

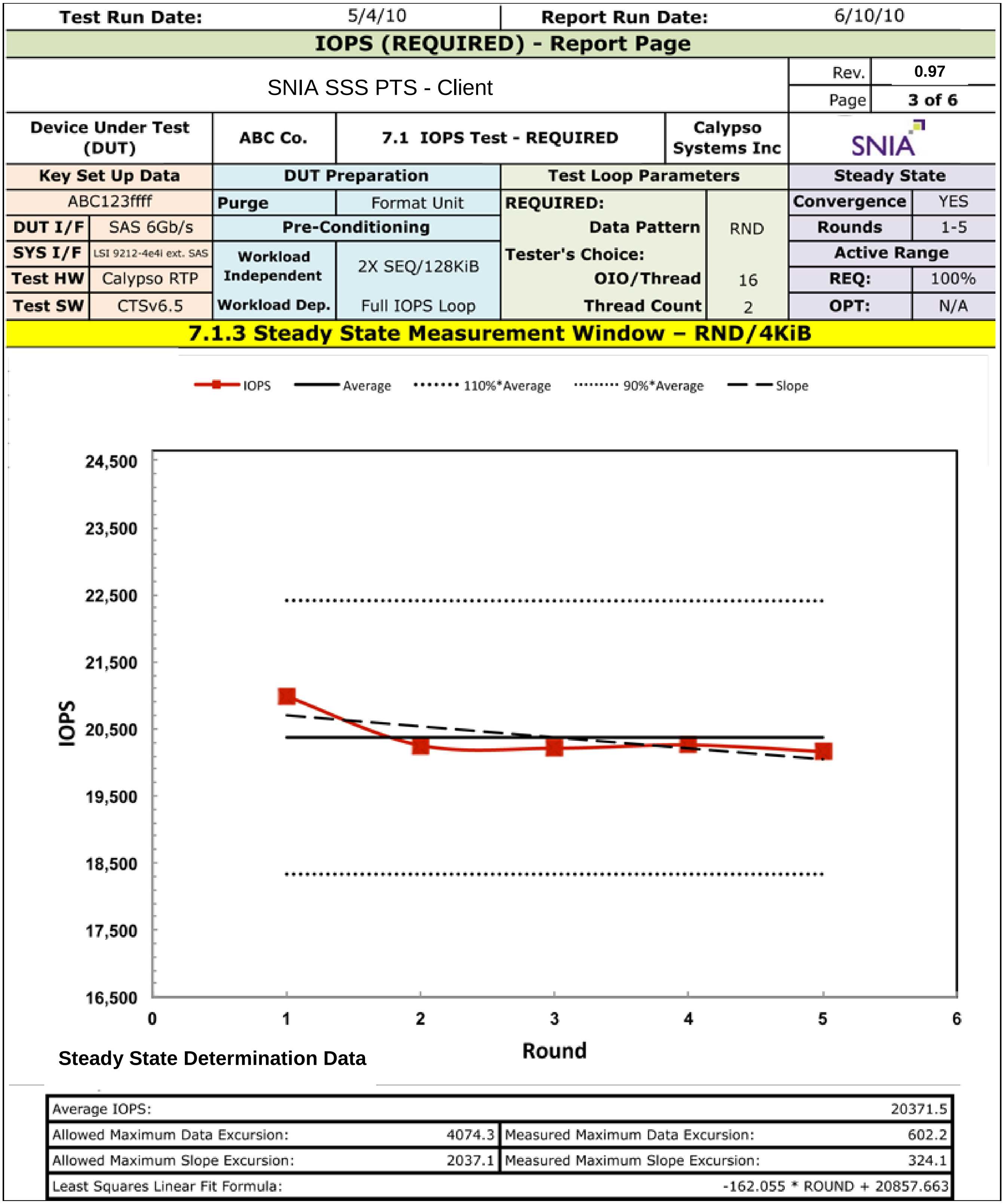
IOPS (REQUIRED) - Report Page

SNIA SSS PTS - Client	Rev.	0.97
	Page	2 of 6

Device Under Test (DUT)	ABC Co.	7.1 IOPS Test - REQUIRED	Calypso Systems Inc	SNIA

Key Set Up Data		DUT Preparation		Test Loop Parameters		Steady State	
ABC123ffff		Purge	Format Unit	REQUIRED:		Convergence	YES
DUT I/F	SAS 6Gb/s	Pre-Conditioning		Data Pattern	RND	Rounds	1-5
SYS I/F	LSI 9212-4e4i ext. SAS	Workload Independent	2X SEQ/128KiB	Tester's Choice:		Active Range	
Test HW	Calypso RTP			OIO/Thread	16	REQ:	100%
Test SW	CTSv6.5	Workload Dep.	Full IOPS Loop	Thread Count	2	OPT:	N/A

7.1.2 Steady State Measurement Window



Test Run Date:	5/4/10	Report Run Date:	6/10/10

IOPS (REQUIRED) - Report Page

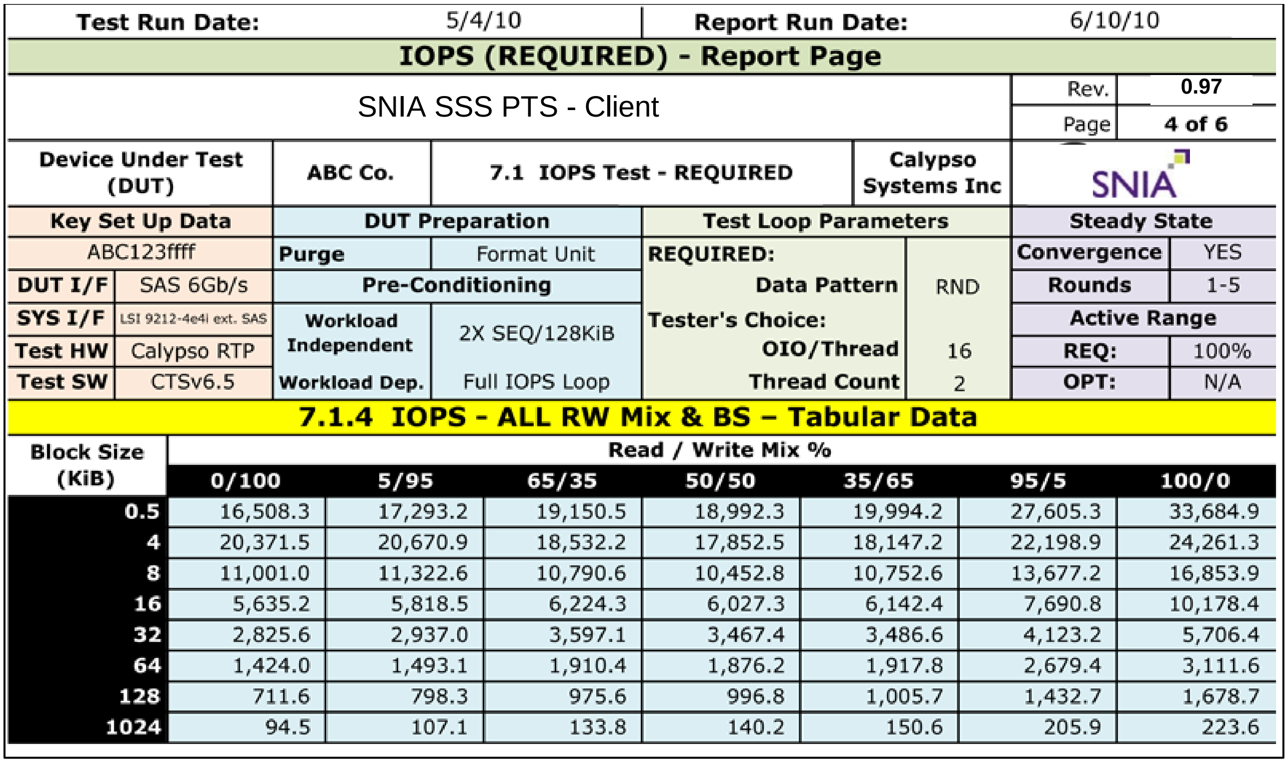
SNIA SSS PTS - Client	Rev.	0.97
	Page	3 of 6

Device Under Test (DUT)		ABC Co.		7.1 IOPS Test - REQUIRED		Calypso Systems Inc	SNIA
Key Set Up Data		**DUT Preparation**		**Test Loop Parameters**		**Steady State**	
ABC123ffff		Purge	Format Unit	REQUIRED:		Convergence	YES
DUT I/F	SAS 6Gb/s	Pre-Conditioning		Data Pattern	RND	Rounds	1-5
SYS I/F	LSI 9212-4e4i ext. SAS	Workload Independent	2X SEQ/128KiB	Tester's Choice:		Active Range	
Test HW	Calypso RTP			OIO/Thread	16	REQ:	100%
Test SW	CTSv6.5	Workload Dep.	Full IOPS Loop	Thread Count	2	OPT:	N/A

7.1.3 Steady State Measurement Window – RND/4KiB

Steady State Determination Data

Average IOPS:			20371.5
Allowed Maximum Data Excursion:	4074.3	Measured Maximum Data Excursion:	602.2
Allowed Maximum Slope Excursion:	2037.1	Measured Maximum Slope Excursion:	324.1
Least Squares Linear Fit Formula:			-162.055 * ROUND + 20857.663



Test Run Date:	5/4/10	Report Run Date:	6/10/10

IOPS (REQUIRED) - Report Page

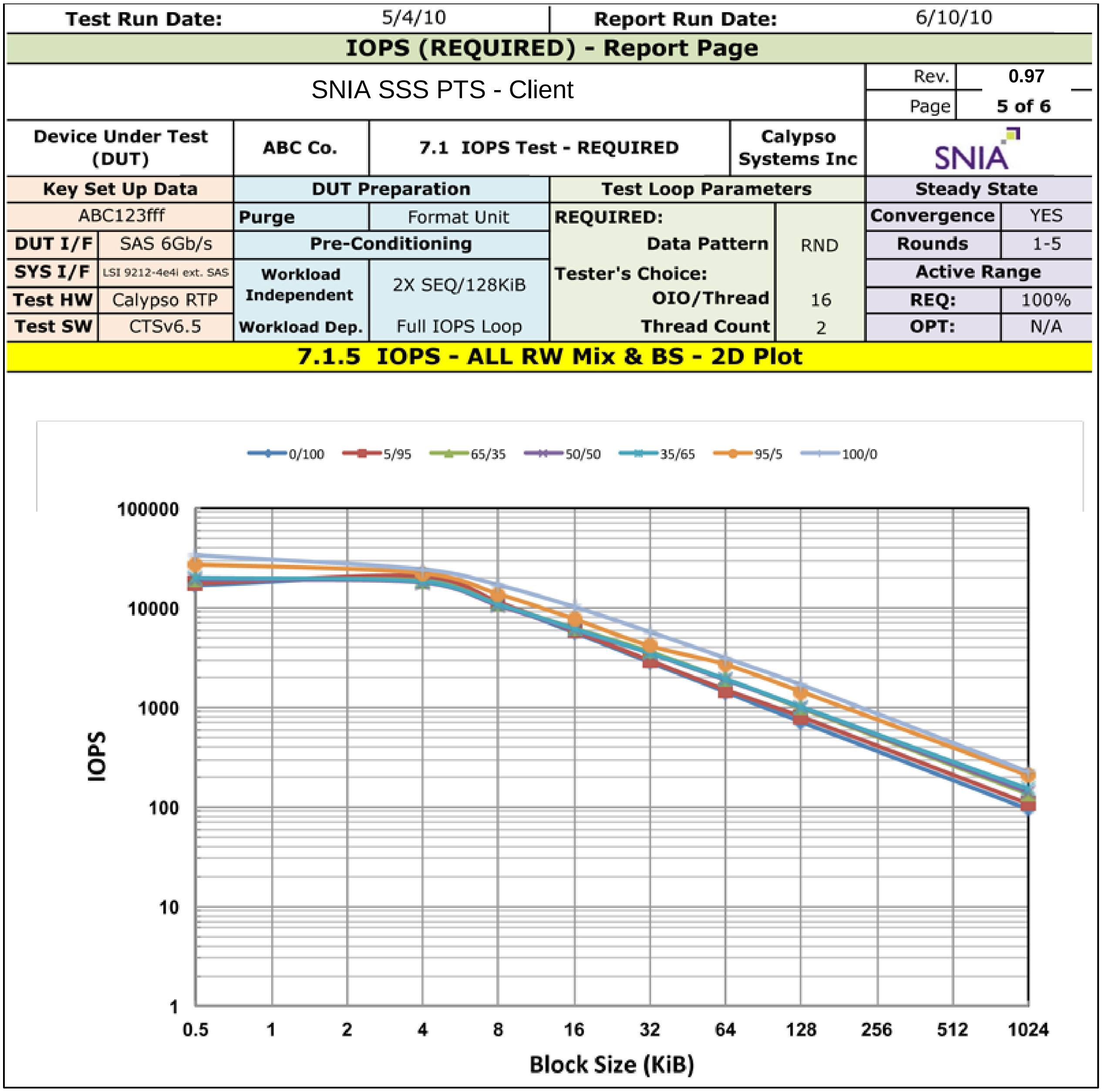
SNIA SSS PTS - Client	Rev.	0.97
	Page	4 of 6

Device Under Test (DUT)	ABC Co.	7.1 IOPS Test - REQUIRED	Calypso Systems Inc	SNIA

Key Set Up Data		DUT Preparation		Test Loop Parameters		Steady State	
ABC123ffff		Purge	Format Unit	REQUIRED:		Convergence	YES
DUT I/F	SAS 6Gb/s	Pre-Conditioning		Data Pattern	RND	Rounds	1-5
SYS I/F	LSI 9212-4e4i ext. SAS	Workload Independent	2X SEQ/128KiB	Tester's Choice:		Active Range	
Test HW	Calypso RTP			OIO/Thread	16	REQ:	100%
Test SW	CTSv6.5	Workload Dep.	Full IOPS Loop	Thread Count	2	OPT:	N/A

7.1.4 IOPS - ALL RW Mix & BS – Tabular Data

Block Size (KiB)	Read / Write Mix %						
	0/100	5/95	65/35	50/50	35/65	95/5	100/0
0.5	16,508.3	17,293.2	19,150.5	18,992.3	19,994.2	27,605.3	33,684.9
4	20,371.5	20,670.9	18,532.2	17,852.5	18,147.2	22,198.9	24,261.3
8	11,001.0	11,322.6	10,790.6	10,452.8	10,752.6	13,677.2	16,853.9
16	5,635.2	5,818.5	6,224.3	6,027.3	6,142.4	7,690.8	10,178.4
32	2,825.6	2,937.0	3,597.1	3,467.4	3,486.6	4,123.2	5,706.4
64	1,424.0	1,493.1	1,910.4	1,876.2	1,917.8	2,679.4	3,111.6
128	711.6	798.3	975.6	996.8	1,005.7	1,432.7	1,678.7
1024	94.5	107.1	133.8	140.2	150.6	205.9	223.6



Test Run Date:	5/4/10	Report Run Date:	6/10/10

IOPS (REQUIRED) - Report Page

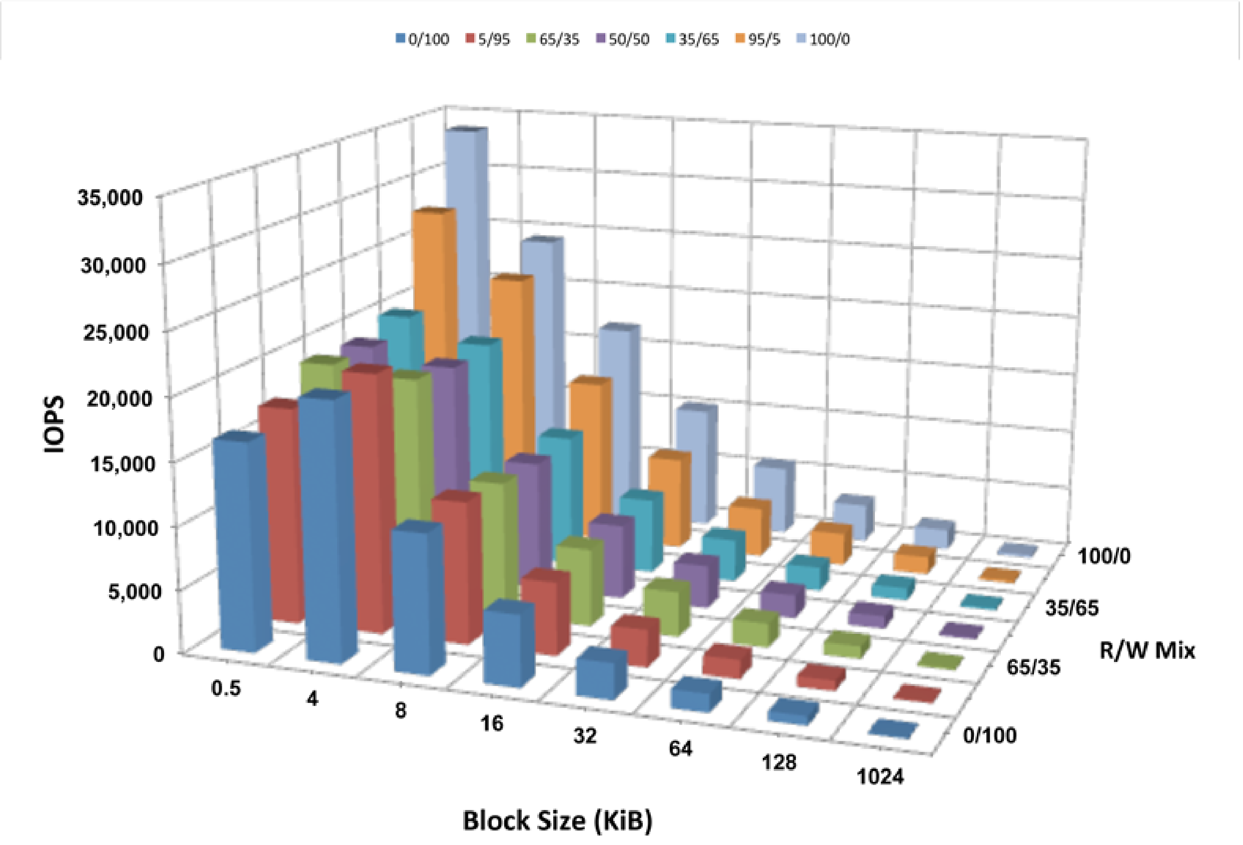
SNIA SSS PTS - Client		Rev.	0.97
		Page	5 of 6
Device Under Test (DUT)	ABC Co.	7.1 IOPS Test - REQUIRED	Calypso Systems Inc

Key Set Up Data		DUT Preparation		Test Loop Parameters		Steady State	
ABC123fff		Purge	Format Unit	REQUIRED:		Convergence	YES
DUT I/F	SAS 6Gb/s	Pre-Conditioning		Data Pattern	RND	Rounds	1-5
SYS I/F	LSI 9212-4e4i ext. SAS	Workload Independent	2X SEQ/128KiB	Tester's Choice:		Active Range	
Test HW	Calypso RTP			OIO/Thread	16	REQ:	100%
Test SW	CTSv6.5	Workload Dep.	Full IOPS Loop	Thread Count	2	OPT:	N/A

7.1.5 IOPS - ALL RW Mix & BS - 2D Plot

Test Run Date:		5/4/10		Report Run Date:		6/10/10	
IOPS (REQUIRED) - Report Page							
SNIA SSS PTS - Client						Rev.	0.97
						Page	6 of 6
Device Under Test (DUT)		ABC Co.	7.1 IOPS Test - REQUIRED		Calypso Systems Inc	SNIA	
Key Set Up Data		DUT Preparation		Test Loop Parameters		Steady State	
ABC123ffff		Purge	Format Unit	REQUIRED:		Convergence	YES
DUT I/F	SAS 6Gb/s	Pre-Conditioning		Data Pattern	RND	Rounds	1-5
SYS I/F	LSI 9212-4e4i ext. SAS	Workload Independent	2X SEQ/128KiB	Tester's Choice:		Active Range	
Test HW	Calypso RTP			OIO/Thread	16	REQ:	100%
Test SW	CTSv6.5	Workload Dep.	Full IOPS Loop	Thread Count	2	OPT:	N/A

7.1.6 IOPS - ALL RW Mix & BS – 3D Columns



11 Informative Annex B – Performance Test Platform Example

This annex describes the hardware/software Reference Test Platform (RTP) that was used by the SSS TWG to do the bulk of the research and validation of the SSS PTS.

The RTP is not required to run the SSS PTS tests; it is an example of a platform that was used to run the PTS.

In addition to the RTP, several other hardware/software platforms and software tools were used in the development and refinement of the PTS, such as IOmeter, Vdbench and several in-house stimulus generators running on several versions of the Windows and Linux O/S.

11.1 RTP Configurations

The RTP was designed to enable the testing of most SSS devices. It can also be extended in order for the testing of higher performance SSS products.

The table below shows the two currently defined configurations of the RTP; the validation of the PTS was performed using the Standard configuration.

Component	Type / Part Number	Standard	Extended Performance
Chassis	Intel SC5650DP or similar	1	1
Motherboard	Intel S5520HC	1	1
Processor	Intel 3.2GHz W5580	1	2
Main Memory	1333MHz DDR3, ECC	12GB	Up to 96GB
HDD	160GB, 7200K RPM	1	1
6Gb/s SAS/SATA HBA	LSI 9212-4i4e	1	1
Operating System	Linux	CentOS 5.4	CentOS 5.4
Test Software	Calypso CTS	CTS V6.5	CTS V7.0

11.2 RTP Components

Following is a more detailed description of the components used in the RTP.

The chassis used was an Intel SC5650DP. The chassis has a limited effect on performance, so a similar chassis could be used, but it should have a 600W or larger power supply.

The RTP motherboard was an Intel S5520HC. Again, a similar motherboard could be used, but it is desirable to have two processor sockets to be able to support the Extended Performance configuration.

The processor used in the RTP was an Intel S5520HC. The Standard configuration contained one processor, and the Extended Performance option would contain two processors.

The RTP main memory utilized 1333MHz DDR3 DIMMs with ECC. The Standard configuration included 12GB of RAM, and the Extended Performance configuration would be fully populated with 96GB of RAM. The DIMMs were selected from the following list of Intel approved memory:

http://serverconfigurator.intel.com/configure-memory.aspx?id=MTY2NCMxLDI1MTcjMSwyNjE4IzIsMjUzMyMx&ShowNavigation=false

It is advisable to install the memory in multiples of three DIMMs, to optimize performance.

The HDD used in the RTP was 160 GB, 7200K RPM, with a SATA interface. It is not clear what impact the HDD has on performance testing, so the highest performance SATA drive that is widely available was selected.

The RTP used a LSI 9212-4i4e Host Bus Adaptor containing the IT firmware set. Other sufficiently high performance HBAs are available, but were not tested.

The RTP hardware platform had a dedicated power supply for all test DUT bays separate from the main power supply.

RTP utilized CTS test software made by Calypso Systems, Inc. www.calypsotesters.com

- CTSv6.5 is the Standard version.
- CTSv7.0 is the Extended version.

The Calypso software requires the CentOS 5.4 Linux OS.