Real World Storage Workload (RWSW) Performance Test Specification for Datacenter Storage

Version 1.0.7

ABSTRACT: This document describes a Real-World Storage Workload (RWSW) IO capture, characterization, methodology, test suite and reporting format. It is intended to provide standardized analysis of in-situ target server application storage performance and standardized comparison and qualification of Datacenter storage when using Reference IO Capture Workloads as the test stimuli in RWSW tests.

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

SNIA Technical Position

May 25, 2018

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.

Revision History

Revision	Release Date	Originator	Comments
1.0.03	Aug-21-2017	Eden Kim	• Initial Draft based on Real World Storage Workload Test Methodologies paper v 1.5 dated June 2017 • Initial overview discussion of document organization • This version to be posted as 'Ballot for Comment' • Ballot open Aug. 25 and close on Sept. 15 • Discussion of comments at S3TWG concall on Sept. 18
1.0.04	Aug-25-2017	Eden Kim	• Fixed broken cross reference links
1.0.05	Sep-18-2017	Eden Kim	• TWG line-by-line review of Tom West Ballot Comments • Edit to Specification Title – added 'Performance' • Edit to Abstract – added 'storage' • Overview – Intent to post reference IO Captures to IOTTA repository • Overview – 'reference' changed to 'example' captures on TMW site • Overview – added language re: IO Captures nevertheless are based on the collection of actual IO Trace data • Definitions – placeholder added for 'workload', 'real world', 'capture step' (cf continuous time) • Definitions – 'cumulative workload' clarified re: time frame of capture • Definitions – 'disk utilization' edited • Definitions – 'idle time' edited • IO Stream Threshold – '3%' threshold – eliminated the word 'default' • Definitions – 'PID' edited • Acronyms 'RND' & 'SEQ' – added full spelling prior to first use • Volatile Write Cache – default=WCE changed to optional • Perfmon – deleted as it does not correctly handle block sizes • IO Stream Map – 'resolution' clarified to mean time step • Pseudo Code Replay Test – addition of idle time parameter
1.0.6	Dec-15-2017	Eden Kim	• Narrow scope to Device level test • Redact reference to software tool names • Remove reference to TestMyWorkload.com • Removal of references to compression and deduplication • Removal of definitions related to compression and deduplication • Addition of Section 9 RWSW DIRTH test as optional pre-conditioning for Section 7 Replay Native test.
1.0.7	Jan-23-2018	Eden Kim	• F2F comments • Change Self Test to Target Server Self Test

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

Contributors

The SNIA SSS Technical Work Group (S3 TWG), which developed and reviewed this standard, would like to recognize the contributions made by the following members:

Company	Contributor
Calypso	Eden Kim
dxc	Chuck Paridon
hyperI/O	Tom West
Hewlett Packard Enterprises	Keith Orsack
Micron Technology	Doug Rollins
Micron Technology	Michael Selzler
SK Hynix	Santosh Kumar
Western Digital	Marty Czekalski

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. The capture, characterization and test of Real World Storage Workloads is expected to be of interest to Datacenters, IT Professionals, Storage Server companies, SSD ODMs and OEMs, firmware designers, controller designers and researchers. 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 Datacenter Storage, Storage Servers, SSS devices and systems and sub systems incorporating SSS devices and logical storage.

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.

This page is intentionally left blank.

Table of Contents

Contents

List of Figures, Plots, and Tables

1. Introduction

1.1 Overview

There is a large demand among Datacenter designers, IT managers and storage professionals for a standardized methodology to capture, characterize and test "Real-World Application Workloads" to more effectively assess the performance of datacenter storage and to qualify storage used in datacenters.

In this context, "datacenter storage" and "storage used in datacenters" includes, but is not limited to, SAN (Storage Attached Networks), NAS (Network Attached Storage), DAS (Direct Attached Storage), JBOF (Just a Bunch of Flash), JBOD (Just a Bunch of Drives), SDS (Software Defined Storage), Virtualized Storage, Object Drives, LUNs (Logical Units), SSDs (Solid State Drives), and other logical storage.

This Specification sets forth a methodology for the capture, characterization and test of Real-World Storage Workloads (RWSWs). RWSWs are the collection of IOs generated by applications that traverse the software stack from User space to storage and back. Real-World *Storage* Workloads are those IOs, or IO Streams (see IO Streams below), that present to storage at the File System, Block IO or other specified layer. RWSW IO Streams are modified at each layer of software abstraction by Operating System activities, metadata, journaling, virtualization, encryption, compression, deduplication and other optimizations.

While RWSWs will vary from capture to capture, all real-world IO captures share the common characteristics of constantly changing combinations of IO Streams and Queue Depths over time. This document defines a common test methodology that can be applied to all IO captures.

Various public and private IO capture tools are available and differ by the Operating System(s) supported, the software stack level(s) at which the captures are taken, and the IO metrics that are cataloged. Generally speaking, IO Captures are the presentation of metrics associated with IO Trace data that are collected by the IO Capture tool. While IO Captures present IOs in specified time interval steps, IO Captures are nonetheless based on the collection of actual IO Trace data.

The tests described herein use IO Captures to create workloads to test target storage. While a variety of software tools can be used to create the test scripts described herein, the user should take care to understand and disclose which metrics are provided by the selected IO Capture tool.

Future efforts are planned to gather additional captures, to define standard workloads (or workloads typical of specific classes of applications), and to develop additional tests based on these workloads. It is intended that, when completed and approved, standard workload IO Captures will be listed on the SNIA IOTTA Repository as Reference IO Captures. Readers are encouraged to participate in the further SNIA SSS TWG works and can post comments at the TWG website portal at http://www.snia.org/feedback/.

1.2 Purpose

Datacenter and storage professionals want to understand how storage responds to real world workloads. This Specification is intended to provide a standardized methodology to capture, characterize and test RWSW using IO Captures of IO Trace data and to use IO Captures to define test stimuli to test and qualify device level storage.

RWSW IO Captures and methodologies can be used to analyze in-situ target server storage performance (the performance of the target server storage during the IO Capture process), used as Reference IO Capture workloads for benchmark comparison test, and used as a way to evaluate new IO Captures and associated application specific IO Stream subsets.

1.3 Background

Traditional benchmark tests and specifications have focused on defining a set of synthetic lab workloads, or corner case stress tests, for the comparative performance test of datacenter storage. The SSS Performance Test Specification (SSS PTS) v 2.0, recently updated and released by SNIA, is an example of a set of such synthetic lab tests and workloads.

Synthetic lab tests continue to be important for storage development, validation and qualification. However, today's storage professionals are increasingly interested in workloads generated by applications in deployed laptops, desktops, servers and datacenters because RWSWs are fundamentally different from synthetic lab workloads. Whereas synthetic lab workloads apply a fixed and constant workload (often comprised of a single, or very few, IO Streams) to storage, RWSWs are comprised of constantly changing combinations of many IO Streams with differing Demand Intensities (or Queue Depths, aka QDs), Idle times and IO rates (or IO bursts).

Datacenter storage performance depends, in large part, on how well storage responds to these dynamically changing RWSWs. In addition, RWSW performance measurements often do not align with published manufacturer specifications or single access pattern corner case test results.

IO Streams that comprise RWSWs are affected, or modified, at each level of abstraction in the Software (SW) Stack. See Figure 1-1 below. IO Streams are appended, coalesced, fragmented or merged as they traverse the SW Stack from User (Application) space to storage and back. Solid state storage responds differently to different types of IO Streams depending on the type of access (Random or Sequential), the data transfer size (or Block Size), whether the IO is a Read or Write IO and the associated Demand Intensity (or QD).[1]
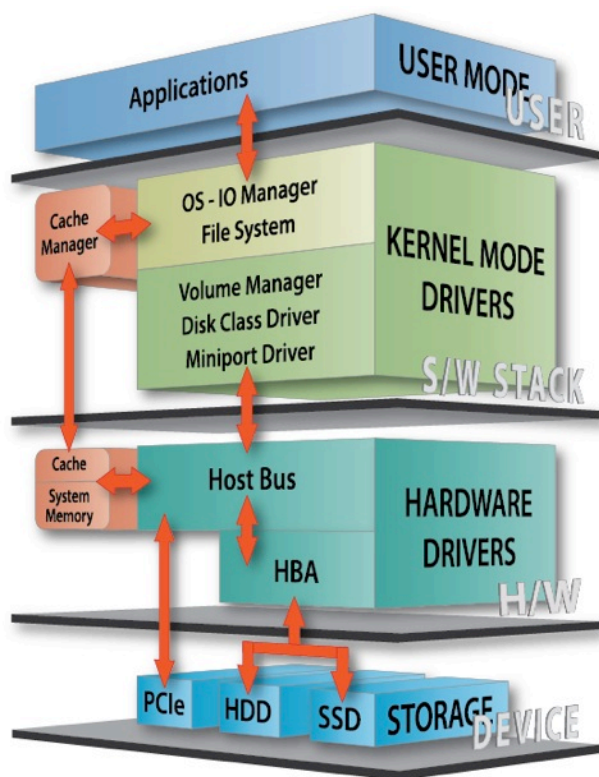
Figure 1-1. Windows Software Stack

It is important to identify IO Stream content at specific levels in the SW Stack (such as the Block IO level) in order to create relevant real-world storage test workloads. Because the performance of solid state storage depends on how well storage responds to the dynamically changing

---

[1] IO Streams used in relation to RWSWs are different than Data Streams used in relation to SSD Endurance where similar write operations are associated with a group of associated data.

combinations of IO Streams and IO bursts, the efficacy of a RWSW test will, therefore, depend on capturing IO Stream content at the appropriate level in the SW Stack.

IO Captures are the collection and tabulation of statistics on IO Streams at a specific SW Stack level over a period of time. Based on IO Trace data, IO Captures parse IO data into discrete time intervals, or steps, for visualization and presentation of IO Streams and their associated metrics. The use of discrete time intervals allows the test operator to vary the resolution, or granularity, of the IO Capture steps depending on the intended purpose(s) of the analysis or test.

For example, using a coarse grain resolution (larger time interval) for an IO Capture conducted over a very long time period can result in a smaller file size whereas using a fine grain resolution (smaller time interval) can facilitate the observation and analysis of IO Bursts, IO Sequentiality, Disk Utilization (or Idle Times) and other storage performance related phenomena.

Key aspects of the proposed RWSW tests in this Specification are based on reproducing the unique combination of IO Streams and Queue Depths that occur during each step of an IO Capture. The ability to apply the observed IO Streams, Idle times and IO bursts provides the test sponsor with an important methodology to compare test storage with actual workloads captured on the target server. In addition, the test sponsor can increase the RWSW test Demand Intensity setting (QDs) to ensure saturation of test storage for thorough performance evaluation.

Note: There is a distinction between IO Capture Step Resolution and IO Trace Continuous Replay. The proposed Applied Test Workloads are based on a selected set of IO Streams that are applied in discrete time intervals, or IO Capture steps, during the Replay-Native test. While the test operator may set the IO Capture step time interval resolution to a small value, the Replay-Native test may not be a true continuous IO Trace replay.

At some point it is not possible to resolve increasingly small time intervals. For example, in one instance, steps could appear discrete at some finite level (small time interval) yet still be continuous at a higher time interval resolution. In another instance, an apparently continuous IO Trace replay can subsequently appear as discrete steps at a very small time resolution.

Information about the IO Capture process, visualization of IO Captures, IO Capture Metrics, extracting a RWSW from IO Captures and performance analysis of IO Captures is discussed in Section 5 Presentation & Evaluation of IO Captures.

Section 6 Target Server Self-Test introduces the concept of Target Server Self-Test of the original IO Capture. Target Server Self-Test is a pseudo-test that presents the performance of the target server during the IO Capture process. This allows for analysis of in-situ target server performance, software optimization, and easy comparison to RWSW Replay-Native test results.

Methodologies, test settings and specific requirements for RWSW tests are set forth in the specific RWSW sections (7 to10) that follow.

1.4 Scope

1) DAS, SAN, NAS, JBOF, JBOD, SSD, LUN and OS recognized Logical Storage
2) IO Capture Tools, Methodology & Metrics
3) Evaluation of Application IO Streams
4) RWSW Test Reporting Requirements
5) RWSW Application Workload tests
6) Reference Real World Storage Workloads – Listed on TestMyWorkload.com

1.5 Not in Scope

1) Solid State Storage Arrays
2) Test Platform (HW/OS/Tools)
3) Certification/Validation procedures for this Specification

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.

- SNIA SSS PTS v 2.0– Solid State Storage Device Performance Test Specification

- IOTTA Repository – public repository for IO Trace, Tools & Analysis

1.7.2 References under development

- SNIA Solid State Storage Systems TWG – Initial Draft Methodology Document

1.7.3 Other references

- None in this version

2 Definitions, symbols, abbreviations, and conventions

2.1 Definitions

2.1.1 **ActiveRange -** Specified as ActiveRange(start:end), where "start" and "end" are percentages. ActiveRange (AR) is the range of LBA's that may be accessed by the pre-conditioning and/or test code, where the starting LBA# = start%*MaxUserLBA and the ending LBA# = end%*MaxUserLBA.

2.1.2 **Applied Test Workload –** either the Cumulative Workload in its entirety, or an extracted subset of the Cumulative Workload, that is used as a test workload. The percentage of occurrence for each IO Stream is normalized such that the total of all the Applied Test Workload IO Streams equals 100%.

2.1.3 **Back-up -** A collection of data stored on (usually removable) non-volatile storage media for purposes of recovery in case the original copy of data is lost or becomes inaccessible; also called a backup copy.

2.1.4 **Block -** The unit in which data is stored and retrieved on disk and tape devices; the atomic unit of data recognition (through a preamble and block header) and protection (through a CRC or ECC).

2.1.5 **Block IO level –** the level of abstraction in the host server used by logical and physical volumes responsible for storing or retrieving specified blocks of data.

2.1.6 **Block Storage System -** A subsystem that provides block level access to storage for other systems or other layers of the same system. See block.

2.1.7 **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.

2.1.8 Capture **Step –** the time interval of an IO Capture used to apply or present IO Capture metrics as in the IO Capture step for an IO Stream Map or as in the step resolution of an IO Capture or Replay-Native test.

2.1.9 **Client -** Single user laptop or desktop computers used in small office, home, mobile, entertainment and other single user applications.

2.1.10 **Cumulative Workload –** a collection of one or more IO Streams listed from an IO Capture that occur over the course of an entire IO Capture. E.g. six dominant IO Streams may occur over a 24-hour capture and be listed as the Cumulative Workload of 6 IO Streams (with each IO Stream % of occurrence over the 24-hours listed).

2.1.11 **CPU Usage -** amount of time for which a central processing unit (CPU) is used for processing instructions. CPU time is also measured as a percentage of the CPU's capacity at any given time.

2.1.12 **Data Excursion -** As used in the definition of Steady State, shall be measured by taking the absolute value of the difference between each sample and the average.

2.1.13 **Disk Performance Utilization –** The percent of time that storage is being accessed during a given time period.

2.1.14 **Drive Fill (level):** The level to which a storage device has been written. Drive Fill is often emulated using the ActiveRange settings whereby a number of LBA ranges are excluded from the available test range.

2.1.15 **Drive Fills (Capacity) –** The number of 'Drive Fills' is used synonymously with User Capacity where a single Drive Fill is an amount equal to the stated User Capacity. Drive Fills are used to indicate normalized storage capacity based on the stated User Capacity. For example, in a 1TB SSD, one Drive fill would be 1TB of writes. 2.5 TB of writes would be equal to 2.5 Drive Fills.

2.1.16 **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.

2.1.17 **File System -** A software component that imposes structure on the address space of one or more physical or virtual disks so that applications may deal more conveniently with abstract named data objects of variable size (files).

2.1.18 **File System level –** a location to access and store files and folders and requires file level protocols to access the storage. File System level storage typically includes system and volatile cache.

2.1.19 **Fresh-Out-of-the-Box (FOB) -** State of SSS prior to being put into service or in a state as if no writes have occurred (as in after a device Purge).

2.1.20 **Idle time –** a period of no host IO operation or storage activity.

2.1.21 **Individual Streams-WSAT -** A test that runs a single Write Saturation test for each IO Stream component of a multiple IO Stream workload.

2.1.22 **In-situ performance (Self-Test) –** Refers to the performance of the target server during the IO Capture (in-situ to the target server). Performance is presented using plots similar to RWSW Tests for easy comparison of the target server and test storage.

2.1.23 **IO -** an Input/Output (IO) operation that transfers data to or from a computer, peripheral or level in the SW Stack.

2.1.24 **IO Burst –** The temporal grouping of IOs during a designated time period. Usually refers to high IO concentration during short time duration.

2.1.25 **IO Capture -** an IO Capture refers to the collection and tabulation of statistics that describe the IO Streams observed at a given level in the software stack over a given time. An IO Capture is run for a specified time (duration), collects the observed IOs in discrete time intervals, and saves the description of the IO Streams in an appropriate table or list.

2.1.26 **IO Capture Tools –** software tools that gather and tabulate statistics on IO Streams and their associated metrics. IO Capture tools support different OSes, levels in the SW Stack where captures can be taken, and metrics associated with IO Streams. There are many public and private tools designed to capture workloads including, but not limited to, perfmon for Windows, blktrace for Linux, hiomon for Windows by hyperIO and IOProfiler for cross platform Operating Systems (Windows, Linux, macOS, FreeBSD, etc.) by Calypso.

2.1.27 **IO Sequentiality –** The observation of IO access locality of reference.

2.1.28 **IO Stream –** A distinct IO access that has a unique data transfer size, RND or SEQ access and is a Read or a Write IO. For example, a RND 4K W would be a unique IO Stream as would a RND 4K R, RND 4.5K Read, SEQ 128K R, etc. If a given IO Stream, such as a RND 4K W, occurs many times over the course of a workload capture, it is still considered a single IO Stream.

2.1.29 **IO Stream Map –** A visual representation of the collection of IO Streams that occur over time related to an IO Capture or associated RWSW. Each time step presents one or more IO Streams that occur during the capture step and can be represented on the IO Stream map as a data series. IO Stream maps typically display some number of capture IO Stream steps and their metrics (as the Y axis) over Time (as the X axis).

2.1.30 **IO Stream Map Percentage Threshold –** The level at which IO Streams are filtered for presentation on an IO Stream Map. For example, an IO Stream Threshold of 3% means that all IO Streams that occur 3% or more over the capture duration are presented. The IO Stream Threshold can be set and reported by the test operator.

2.1.31 **IO Demand -** Measured number of OIOs executing in the host.

2.1.32 **LBA Range Hit Map –** A visualization of IO spatial locality of reference where LBA range percent is on the Y axis and time is on the x axis with IO access frequency indicated by relative size bubbles.

2.1.33 **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.

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

2.1.35 **MaxUserLBA -** The maximum LBA # addressable in the User Capacity.

2.1.36 **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).

2.1.37 **Multi-WSAT -** A Write Saturation test that applies a fixed composite of multiple IO Streams in the percentages at which they were observed in the original capture.

2.1.38 **Native Demand Intensity (or QD) –** The QD observed during an IO Capture step.

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

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

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

2.1.42 **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.

2.1.43 **Pre-conditioning -** The process of writing data to the device to prepare it for Steady State measurement.

(a) Workload Independent Pre-conditioning (WIPC): The technique of running a prescribed workload, unrelated to the test workload, as a means to facilitate convergence to Steady State.

(b) Workload Dependent Pre-conditioning (WDPC): The technique of running the test workload itself, typically after Workload Independent Pre-conditioning, as a means to put the device in a Steady State relative to the dependent variable being tested.

2.1.44 **Pre-conditioning Code -** Refers to the Pre-conditioning steps set forth in this Specification.

2.1.45 **Point-in-Time Snapshot –** A method of data protection that allows a user to make complete copies at a specific date and time.

2.1.46 **Process ID (PID) –** A PID represents the unique execution of a program(s).

2.1.47 **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.

2.1.48 **Real World –** IOs, IO Streams or workloads derived from or captured on a deployed server during actual use.

2.1.49 **Real World Storage Workload (RWSW) -** a collection of discrete IO Streams (aka data streams and/or access patterns) that are observed at a specified level in the Software (SW) Stack.

2.1.50 **Replicate -** A general term for a copy of a collection of data. See data duplication, point in time snapshot.

2.1.51 **Replay Fixed –** A Replay test that sets the QD to a fixed value.

2.1.52 **Replay Test –** A test that reproduces the sequence and combination of IO Streams and QDs for each step of the IO Capture.

2.1.53 **Replay Native –** A test that sets QD to the values observed during the capture.

2.1.54 **Replay Scaled -** A test that multiplies QD values observed during the capture by a scaling factor.

2.1.55 **Round** - A complete pass through all the prescribed test points for any given test.

2.1.56 **Queue Depth (QD) -** Interchangeably refers to the OIO/Thread produced by the Workload Generator.

2.1.57 **Secondary Workload –** a subset of one or more IO Streams that are extracted from an IO Capture that are used as an Applied Test Workload. The IO Streams may be filtered by Process ID (such as all sqlservr.exe IOs), time range (such as 8 am to noon) or by event (2 am data back-up to drive0).

2.1.58 **Slope -** As used in the definition of Steady State, shall mean the slope of the "Best Linear Fit Line."

2.1.59 **Snapshot -** A point in time copy of a defined collection of data.

2.1.60 **Software Stack (SW Stack) –** refers to the layers of software (Operating System (OS), applications, APIs, drivers and abstractions) that exist between User space and storage.

2.1.61 **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.

2.1.62 **Target Server –** The host server from which an IO Capture is taken.

2.1.63 **Test Code -** Refers to the measurement steps or test flow set forth in the test sections contained in this Specification.

2.1.64 **Test Storage –** Logical storage that is subject to a RWSW or other type of test.

2.1.65 **Transition Zone -** A performance state where the device's performance is changing as it goes from one state to another (such as from FOB to Steady State).

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

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

2.1.68 **Total OIO -** Total outstanding IO Operations specified by a test = (OIO/Thread) * (TC).

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

2.1.70 **Volatile Cache -** A cache that does not retain data through power cycles.

2.1.71 **Workload –** the amount of work, in this case IOs, to be done on a system or server

2.2 Acronyms and Abbreviations

2.2.1 **ART:** Average Response Time

2.2.2 **DAS:** Direct Attached Storage

2.2.3 **DI:** Demand Intensity (aka Total OIO)

2.2.4 **DIRTH:** Demand Intensity / Response Time Histogram

2.2.5 **DUT:** Device Under Test

2.2.6   **FOB:** Fresh Out of Box

2.2.7   **IO:** Input Output Operation

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

2.2.9   **JBOD:** Just a Bunch of Drives

2.2.10   **JBOF:** Just a Bunch of Flash

2.2.11   **LAT:** Latency

2.2.12   **LBA:** Logical Block Address

2.2.13   **LUN:** Logical Unit as in Logical Storage Unit

2.2.14   **Multi-WSAT:** Multiple IO Stream Write Saturation

2.2.15   **NAS:** Network Attached Storage

2.2.16   **OIO:** Outstanding IO

2.2.17   **QD:** Queue Depth

2.2.18   **RND:** Random

2.2.19   **R/W:** Read/Write

2.2.20   **RWSW:** Real World Storage Workload

2.2.21   **SAN:** Storage Attached Network

2.2.22   **SDS:** Software Defined Storage

2.2.23   **SEQ:** Sequential

2.2.24   **SSD:** Solid State Drive

2.2.25   **SSSI:** Solid State Storage Initiative

2.2.26   **SSS:** Solid State Storage

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

2.2.28   **SW Stack:** Software Stack

2.2.29   **SUT:** Storage Under Test

2.2.30   **TC:** Thread Count

2.2.31   **TOIO:** Total Outstanding IO

2.2.32   **TP:** Throughput

2.2.33   **WSAT:** Write Saturation

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 and data transfer rates and amounts 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 Loop 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 Loop 1
- ➢ Set R/W Mix% to 95/5   >>>>> Beginning of Loop 2
- ➢ Set Block Size to 1024 KiB
- ➢ Execute…
- ➢ Record…
- ➢ …

3 Key Test Process Concepts

The performance of solid state storage (SSS) is highly dependent on its prior usage, the pre-test state of the storage and the test parameters and settings. This section describes key SSS test methodology concepts. It is recommended to apply RWSW tests to Steady State when possible. However, it may be impractical to PURGE and run RWSW to Steady State for a LUN, JBOF, JBOD or other logical storage. In those cases, the test operator shall select an appropriate pre-conditioning regime and disclose the test preparation in the test results.

3.1 Steady State

SSS that is Fresh-Out-of-the-Box (FOB), or in an equivalent state, typically exhibits a transient period of elevated performance, which evolves to a stable performance state that is time invariant relative to the workload being applied. This state is referred to as a Steady State (Definition 2.1.61).

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

1) To ensure that a storage's initial performance (FOB or Purged) will not be reported as "typical", since this is transient behavior and not a meaningful indicator of the storage'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 2.1.61, is demonstrated and documented.

Steady State, per Definition 2.1.61, shall meet the Steady State Verification criteria as set forth in each test. Steady State reporting requirements are covered in the respective test sections.

3.2 Purge

The purpose of the Purge process, per Definition 2.1.47, is to put the storage in a state as if no writes have occurred prior to pre-conditioning and testing, and to facilitate a clear demonstration of Steady State convergence behavior.

Purge, when applied, shall be run prior to each pre-conditioning and testing cycle. If the storage under test does not support any kind of Purge method the fact that Purge was not supported/run shall 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) NVMe: FORMAT namespace
d) Vendor specific methods

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

3.3 Pre-conditioning

The goal of pre-conditioning is to facilitate convergence to Steady State during the test itself.

This Specification adopts the SSS PTS v2.0 definition of two types of pre-conditioning:

- Workload Independent Pre-conditioning (Definition 2.1.43 a); and
- Workload Dependent Pre-conditioning (Definition 2.1.43 b)

This Specification also allows the test sponsor to apply and disclose optional pre-conditioning. This optional pre-conditioning can be comprised of a user defined IO Stream or Streams, or in the case of the Replay Native test, the user of another RWSW test immediately prior to the selected test as the optional pre-conditioning. For example, the test sponsor could use the RWSW DIRTH test as a pre-condition to the Replay Native test. In this case, there would be little if any delay between the tests and the Replay Native test would not use a device PURGE or other pre-conditioning. See Section 8 Pseudo Code below.

Note: While Workload Based Pre-conditioning is not a distinct step in the test scripts (it occurs as part of running the core test loop in each test), it is critically important to achieving valid Steady State results.

3.4 ActiveRange

It is desirable to be able to test the performance characteristics of workloads that issue IOs across a wide range of the LBA space as opposed to those which issue IOs across only a narrow range. To enable this capability, this Specification defines ActiveRange (see 2.1.1).

ActiveRange can also be used to emulate the amount to which storage has been filled with prior IO activity (or Drive Fill). When applying RWSW tests to storage on a deployed system, or when testing target storage in the lab, the test operator may choose to emulate a relative state of Drive Fill by setting the ActiveRange to a value less than 100%. For example, it is common to test storage designed for Client laptops to an ActiveRange of 80% to emulate the level of Drive Fill during real-world usage.

The test scripts define required and optional settings for ActiveRange.

Figure 3-1 show two examples of ActiveRange.

ActiveRange (0:100) **ActiveRange (0:75)**
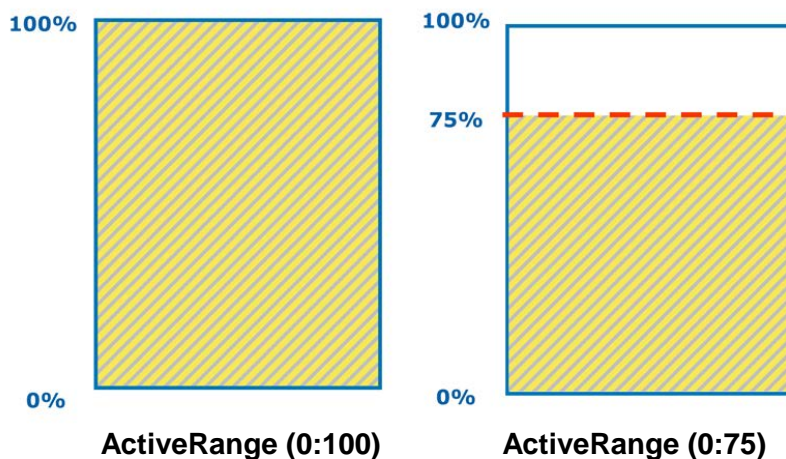
Figure 3-1. ActiveRange Diagram

3.5 Data Patterns

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

3.6 Multiple Thread Guideline

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

While the Test Operator may select a given OIO for a test, the Test Operator shall use the same Thread Count and OIO/Thread for all steps of a given test.

3.7 Caching

The Volatile Write Cache setting is user selectable and may be set by the test operator and shall be disclosed in test reporting. E.g. Write Cache Enabled (WCE) or Write Cache Disabled (WCD).

3.8 IO Capture – SW Stack Level

IO Captures shall be captured at the Block IO level. IO Captures may optionally be captured at other levels in the SW Stack, such as at the File System level. The test operator shall disclose the SW Stack level at which the IO Capture is taken.

3.9 IO Capture – Logical Units (LUN)

IO Capture tools typically capture IO Streams for all logical storage recognized at the selected SW Stack level. This Specification assumes the capture of IO Streams of a single logical storage unit or device. The test operator shall disclose the logical storage upon which the IO Capture has been conducted, such as Drive0, disk1 or sda (such designation depending upon how the OS designates storage units).

The test operator may optionally report IO Streams associated with more than one logical storage unit (LUN). The test operator may report IO Stream activity by LUN, or may report IO Stream activity by aggregating multiple LUNs into a single IO Capture set.

4 Software Tools & Reporting Requirements

This Specification is software tool and hardware agnostic. Any IO Capture tool or software tool that meets the requirements of the Specification may be used.

4.1 IO Capture Tools

There are several public and private IO capture tools available. IO Capture tools differ by Operating System(s) supported, levels in the SW Stack at which the captures are taken and the IO metrics that are catalogued. It is important to understand and disclose the level in the SW Stack where the IO Capture is taken and the IO metrics that are catalogued.

Because IO Streams are modified as they traverse the SW Stack, IO Stream content will be different at the File System and the Block IO levels.

- For example, File System level captures tend to capture IO Streams with data transfer sizes reported in bytes (as many IOs are written to cache) compared to Block IO level data transfer sizes that tend to be reported in kilobytes.

- Second, small block writes seen at the File System may be written to cache and subsequently merged with other small blocks or otherwise appended with metadata before being presented to storage at the Block IO level.

- Third, large block SEQ Reads or Writes may be fragmented into smaller concurrent RND Reads and Writes as the IO Streams move up or down the SW Stack.

4.2 Software Tools

Software tools used to create the scripts and to test target storage pursuant to this Specification shall have the ability to:

1) Act as workload stimulus generator as well as data recorder
2) Load the memory buffer with binary data of know compressibility or duplication ratio[2]
3) Issue Random (RND) and Sequential (SEQ) Block level I/O
4) Restrict LBA accesses to a particular range of available user LBA space
5) Limit "total unique LBAs used" to a specific value (aka Test Active Range)
6) Randomly distribute a number of equally sized LBA segments across the Test Active Range
7) Set R/W percentage mix % for each test step[3]
8) Set Random/Sequential IO mix % for each test step[4]
9) Set IO Transfer Size for each test step[5]
10) Set Queue Depth for each test step[6]
11) Generate and maintain multiple outstanding IO requests. Ensure that all steps in the test sequence can be executed immediately one after the other, to ensure that storage is not recovering between processing steps, unless recovery is the explicit goal of the test.
12) Provide output, or output that can be used to derive, IOPS, MB/s, response times and other specified metrics within some measurement period or with each test step

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.

Note that different software tools operate at different levels in the SW Stack. This can affect the reporting of metrics such as response times where cache may interact with the SW tool. Accordingly, it is recommended to use SW tools that operate as close to storage as possible.

---

[2] This feature is necessary for RWSW test step parameter requirements
[3] This feature is necessary for RWSW test step parameter requirements
[4] This feature is necessary for RWSW test step parameter requirements
[5] This feature is necessary for RWSW test step parameter requirements
[6] This feature is necessary for RWSW test step parameter requirements

4.3 Common Reporting Requirements

There are Common Reporting Requirements for the IO Capture, the Applied Test Workload and for the Test Results for the RWSW tests listed in this Specification. Each IO Capture, Applied Test Workload and RWSW test result shall report the following information as applicable.

The test operator shall report the required (default) and optional (user selectable) parameters, settings and results values as set forth in the Common Reporting Requirements and as specified in the In-Situ Target Server Self-Test or RWSW Test sections. However, the test operator may select the reporting format, unless specifically stated otherwise, and may select to present data in tabular, chart or plot format.

4.3.1 General

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

4.3.2 Original IO Capture

1) Date of Capture, IO Capture Tool
2) IO Capture SW Stack Level (e.g. File System or Block IO level)
3) Main Application activity of interest during IO Capture (e.g. sqlservr.exe)
4) General type/class of IO Capture (e.g. Web Portal Streaming Media)
5) Hardware System including RAM, CPU
6) Operating System, Storage Architecture, logical storage units
7) Storage Software (such as SDS, VMs, etc.)
8) Duration, Time Resolution & Number of steps in/of the IO Capture
9) Type of storage (logical, physical, SSD, HDD, etc.)
10) Storage Manufacturer, serial number, capacity

4.3.3 Applied Test Workload

1) Number of Test Segments that comprise the Applied Test Workload
2) Number, Distribution and Percentage of IO Streams in the Applied Test Workload segments
3) Queue Depth Settings of original IO Capture by step
4) Demand Intensity Settings – QD or IO throttling of test steps
5) For Replay test, the Cumulative IO Stream distribution and the number and duration of individual test steps (as it is impractical to list IO Streams for each of very many steps)[7]

4.3.4 RWSW Test

1) Name of RWSW Test
2) Applied Test Workload for Multi-WSAT, Individual Streams-WSAT or RWSW DIRTH
3) Replay Test Demand Intensity – Native, Scaled, Fixed
4) Target Test Storage Architecture – SSD, DAS, SAN, NAS, JBOD, JBOF, RAID, etc.
5) Target Test Storage - type, devices, volumes, model, SW level
6) User Capacity, Interface/Speed, Form Factor (e.g., 2.5")
7) Media Type (e.g., MLC NAND Flash)
8) Optional: Other major relevant features (e.g. optional pre-conditioning, protocol, virtualization details, etc.)

---

[7] For example, the number of steps in a 24-hour Replay test with a step resolution of 1 second will be equal to 60 seconds x 60 minutes x 24 hours, or a total of 86,400 individual test steps. Listing the IO Streams for each of the 86,400 test steps is impractical. Instead, listing of the Cumulative Workload IO Streams distribution is specified.

5   Presentation & Evaluation of IO Captures

5.1   IO Capture Process and Naming Conventions

The test operator shall run the IO Capture at the desired SW Stack level using the desired IO Capture tool. The IO statistics shall be gathered and reported per the requirements of section 4.3.2 Original IO Capture.

When naming an IO Capture, it is recommended to identify the original IO Capture by the target server architecture, primary application of interest, IO Stream count, capture duration and time interval.

5.2   Visual Presentation of the IO Capture using an IO Stream Map

An IO Capture can be visualized by creating an IO Stream Map that shows IO Streams on the Y axis with Time on the X axis. Figure 5-1 below is an example of an IO Stream Map. IO Stream Maps are an optional reporting element.
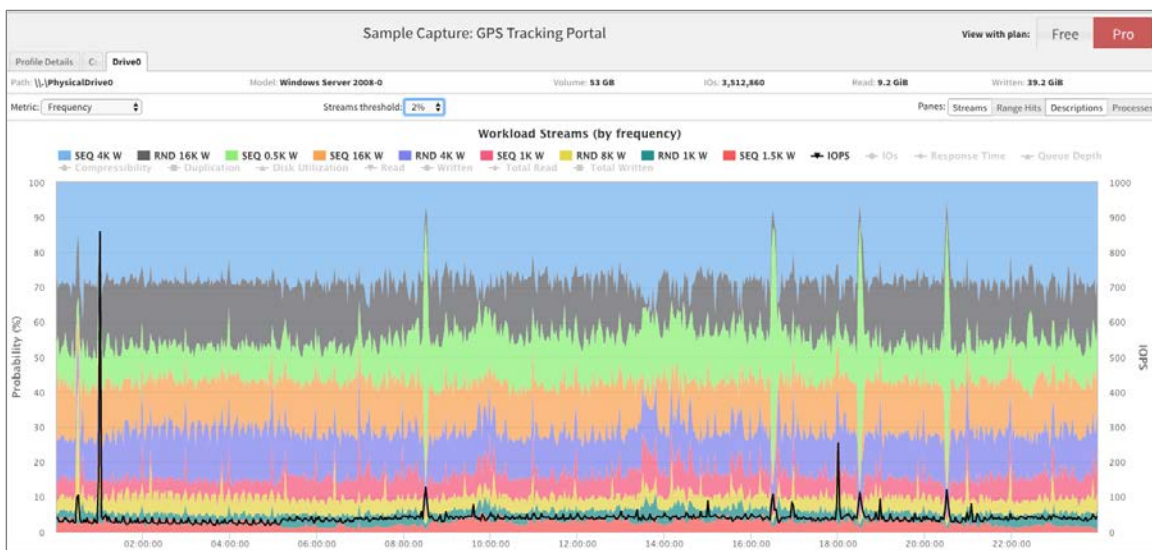

Figure 5-1. IO Stream Map

The number of IO Streams (or unique access patterns of a specific data transfer size and Read or Write IO) to be displayed are determined by the test operator and can be filtered by percent occurrence over the IO Capture duration (such as by IO Stream Map Threshold 2.1.30), by application process IOs (such as all sqlservr.exe IOs), by time (such as a time point or time range), by an activity or event (such as 1 am back-up), or by Cumulative workload (see 2.1.10).

In the example above, IO Streams are filtered to an IO Stream Threshold of 2% for the Cumulative workload on Drive0. This results in showing 9 IO Streams that occur 2% or more of the time over the 24-hour capture. Each of the 9 IO Streams is presented as a different color data series. Probability of IO Stream occurrence is on the Y axis while IOPS are shown by the dominant black line on the secondary Y axis.

Use of colored data series helps to visualize the changing combinations and percentage weight of IO Streams over the capture duration.

5.3   Listing Process IDs and Cumulative Workload List

Once an IO Capture is taken, the identified Process IDs (PIDs) shall be listed by percent and number of IOs that occur over the capture duration. PIDs are also referred to as 'Application IOs' but can include activity such as metadata, OS activity, journaling and more.

Figure 5-2. Process ID List

Figure 5-3. Cumulative Workload List

Figure 5-2. Process ID List above shows a total of 49 Process IDs (PIDs) identified by the IO Capture tool that occurred over the 24-hour capture duration.   Each process is listed in descending order by the percentage of occurrence of the process and by total IO count. Here, the dominant processes are mysqld.exe (64.1%), sqlservr.exe (16.7%) and System (15.7%).

Note: Process IDs do not list every IO associated with a given process or application. Many IO activities can be masked or hidden when performed by or within the System IOs (often after applications are loaded to system memory).

Figure 5-3. Cumulative Workload List above shows the dominant IO Streams selected at the 2% IO Stream Threshold. In this case, 9 IO Streams occur 2% or more of the time over the 24-hour capture duration and represent 78% of the total IOs.

When the Applied Test Workload is created, the relative IO Stream percentages are normalized so that the sum of the 9 IO Stream percentages equals 100%. See Creating an Applied Test Workload below.

Note: The total number of discrete IO Streams captured in a RWSW IO Capture can number in the 10s, to 100s, to 1,000s. Here, over 1,000 IO Streams (1,033) occur that generate 3,512,860 IOs. However, the dominant 9 IO Streams at the 2% IO Stream Threshold comprise 78%, or 2,755,026, of the total IOs that occur over the 24-hour capture duration.

IO Captures can be further filtered, or parsed, to extract IO Streams for periods of time, by event, by process and/or by logical storage. The following examples in this Specification are based on the GPS Navigation Portal 2% IO Stream Threshold IOs for the Cumulative workload for Drive0.

5.4 Creating an Applied Test Workload – Cumulative Workload Drive0

Once the IO Capture workload is determined, the list of IO Streams are used to create the Applied Test Workload (see Definition 2.1.2). For the 9 selected IO Capture IO Streams, the percentage occurrence of each IO Stream during the capture is listed on the left with the normalized percentage for the Applied Test Workload listed on the right (in boxes). For example, SEQ 4K W occurs 21.6% of the time in the 1,033 IO Stream capture but is normalized to equal 27.3% probability of occurrence in the 9 IO Stream Applied Test Workload.

Σ0	Drive0 Cumulative W	✖
SEQ 4K W	21.6%	27.3
RND 16K W	12.0%	15.4
SEQ 0.5K W	11.7%	14.9
SEQ 16K W	10.7%	13.7
RND 4K W	9.6%	12.2
SEQ 1K W	4.9%	6.3
RND 8K W	3.5%	4.4
RND 1K W	2.41%	3.1
SEQ 1.5K W	2.07%	2.65

Figure 5-4. Applied Test Workload

5.5 Reporting Demand Intensity (Queue Depth) of Original Capture

The Native Demand Intensity or QD (see Definition 2.1.38) shall be reported for each step of the original capture. Demand Intensity (aka Outstanding IO or OIO) is also referred to as the QD. The test operator may set the QD to Native (reproducing the observed QD for each step), Static (a fixed value such as T4Q32 as in the Multi-WSAT test), or to Scaled (some multiplier of the Native QD) as the RWSW test may require or permit. See RWSW test sections that follow.
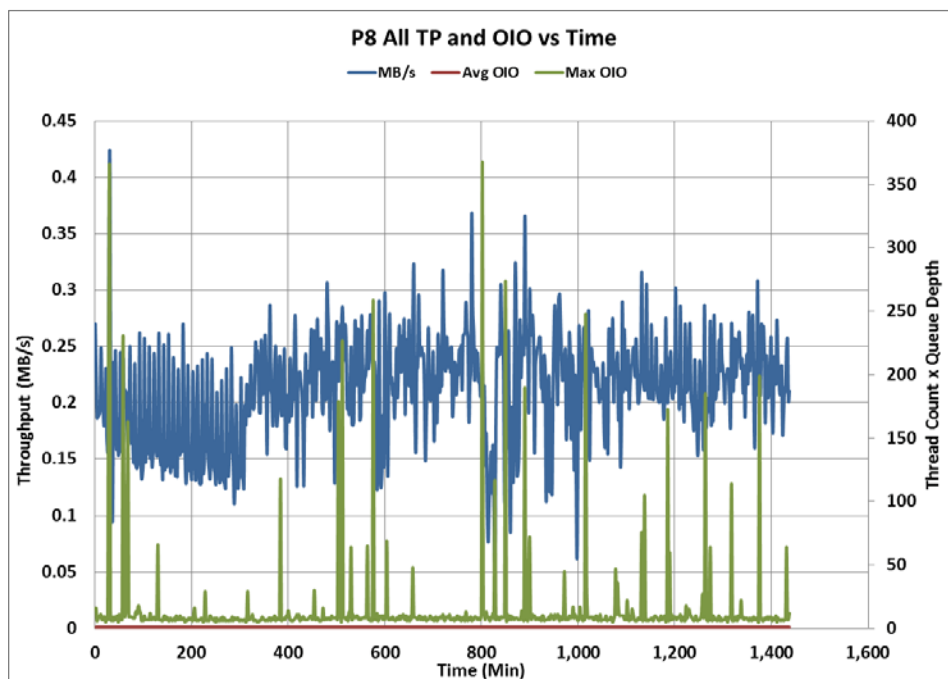

Figure 5-5. Throughput and OIO of Original Capture IOs

6 Target Server Self-Test

6.1 Target Server Self-Test Descriptive Note

General Purpose:

In-situ performance, or Target Server Self-Test, presents the performance of the target server during the IO Capture process for the selected IO Streams of interest. In-situ Target Server Self-Test is not an actual test but rather the compilation of performance measurements based on IO metrics taken during the IO Capture. In-situ target server performance can be analyzed by presenting the IO Stream combinations, QDs and IO metrics associated with the selected IO Capture IO Streams.

Note:

There is no Test Flow for In-situ Target Server Self-Test. Target Server Self-Test extracts IO Streams and metrics defined by the Applied Test Workload from the original IO Capture data. Any combination or subset of IO Stream metrics can be extracted and presented to the extent that the IO Capture tool provides such IO Stream metrics. Measurements are reported as the average value over the duration of each test or capture step.

Test Flow:

1. **Select the desired IO Capture**
2. **Create the Applied Test Workload based on PIDs, IO Streams, Time, Events or Storage**
2. **Tabulate IO Streams, QDs, Idle Times and associated IO Metrics**
3. **Create plots, graphs and figures as specified**

Test Results:

The test results presented for the Target Server Self-Test are similar to test results presented for the Replay-Native test and RWSW tests that follow. This allows the test operator and reader to easily compare in-situ Target Server Self-Test performance with the RWSW tests applied to test storage.

Test Interpretation:

The reader or test operator should take note of the IOPS, Bandwidth, Response Times and QDs that occur during the IO Capture as well as those metrics associated with specific IO Streams, applications, events, time periods or logical storage unit(s).

The relative performance shown in-situ Target Server Self-Test reports will typically be lower than lab testing of storage using the same Replay-Native test workloads. This is largely due to IO throttling that occurs on the target server. The number of IOs and associated performance on the target server is limited by the application IOs, QDs and disk (or storage) utilization that occur during the IO Capture whereas lab testing of Datacenter storage applies IOs in a limitless fashion.

Analysis can be made of both Cumulative workloads as well as workload subsets (such as sqlservr.exe IOs) extracted from the IO Capture. Native Queue Depths can be used for setting Demand Intensity parameters in RWSW tests. IO burst, disk utilization, compression ratio, duplication ratio and other metrics can also assist in the analysis and validation of software stack and storage optimizations.

6.2 Target Server Self-Test Pseudo Code

There is no Pseudo Code for In-situ Target Server Self-Test.

6.3 Test Specific Reporting for Target Server Self-Test

Figures 6.3.1 through 6.3.9 list the reporting requirements specific to the Target Server Self-Test. The test operator may report required data in tabular or plot format. Reporting requirements common to all tests are documented in Section 4.3. The Target Server Self-Test reports that

follow are based on the Applied Test Workload in Section 5.1 IO Stream Map and 5.4 Applied Test Workload above.

6.3.1 IO Streams Distribution

The Test Operator shall report the Applied Test Workload component IO Streams distribution and the IO Stream percentage of occurrence.

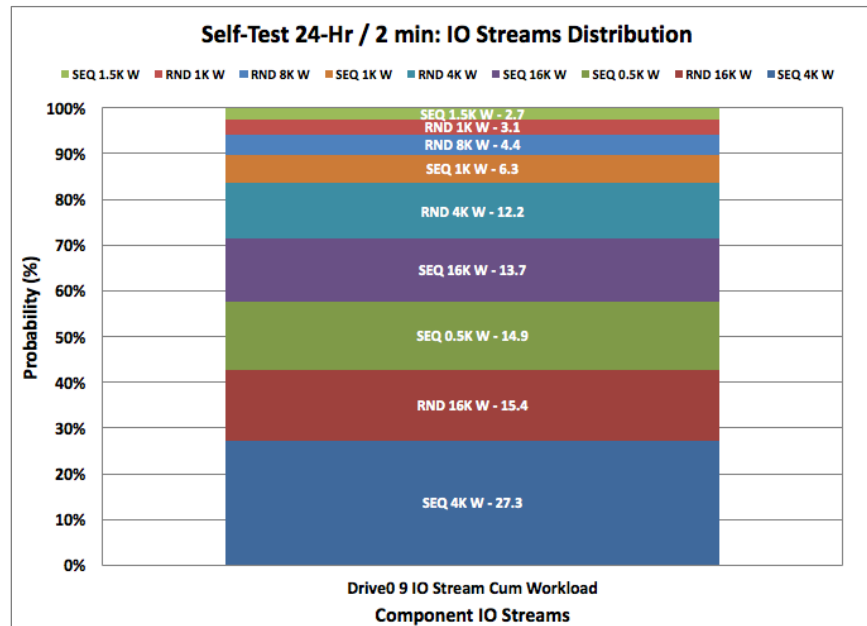**Figure 6-1. Self-Test 24-Hr/2 min: IO Streams Distribution**

6.3.2 IO Streams Map by Quantity of IOs

The test operator may report IO Streams, IOs and IOPS by IO Capture step over Time. The IO Stream Map by Quantity of IOs below shows the changing combination of IO Streams (colored data series), IO count (primary Y axis) and IOPS (blue dot and secondary Y axis) for each IO Capture step (time on x axis).
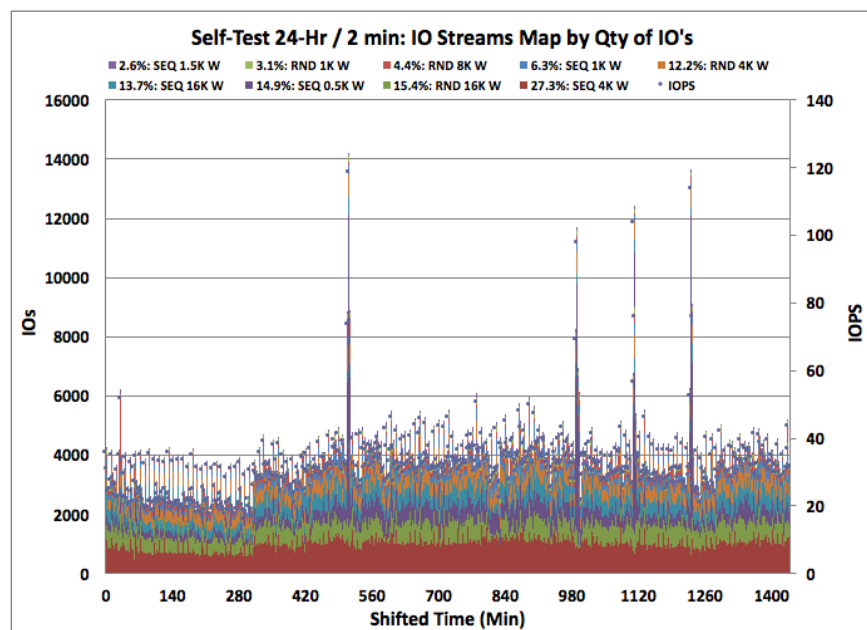
Figure 6-2. IO Streams Map by Quantity of IOs

6.3.3 Probability of IO Streams by Quantity of IOs

The test operator may report the Probability of IO Streams by Quantity of IOs for each IO Capture step over time. This shows the probability of occurrence percent for each IO Stream at each step.
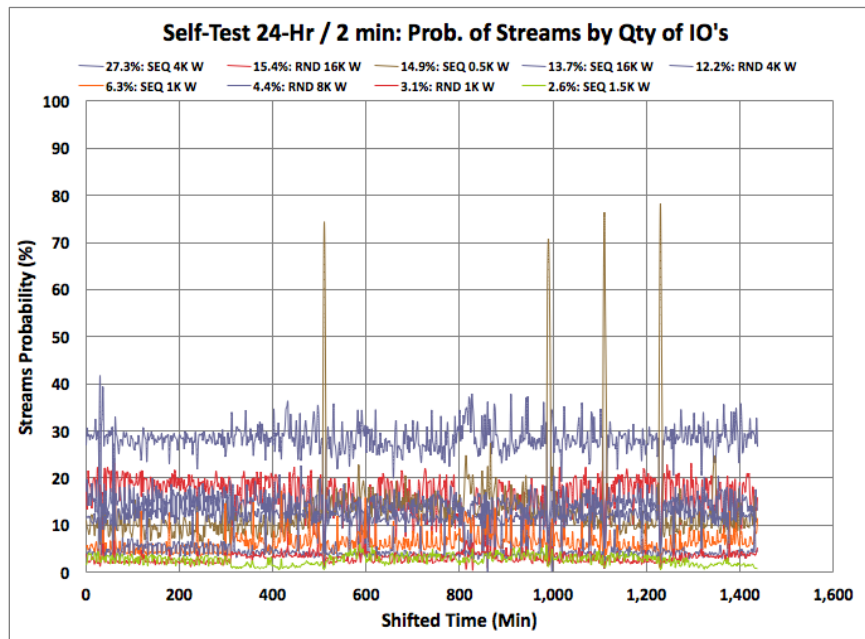
Figure 6-3. Probability of IO Streams by Quantity of IOs

6.3.4 Throughput & Queue Depths v Time: 24-Hr Plot

The test operator shall report Throughput (TP) & Queue Depths (QD) over the 24-hour capture period. TP shall be reported in MB/s. Average and Maximum QDs shall be reported.
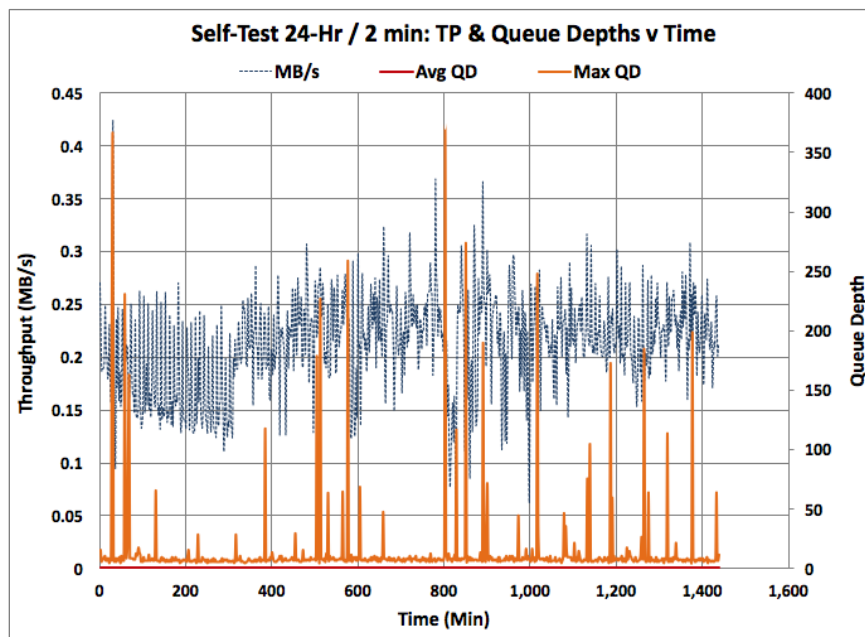
Figure 6-4. Self-Test 24-Hr/2 min: TP & Queue Depth v Time

6.3.5 IOPS v Time: 24-Hr Plot

The test operator shall report IOPS v Time for the 24-hour capture period. IOPS shall be averaged over each IO Capture step with each of the IO Capture steps plotted on the x-axis.
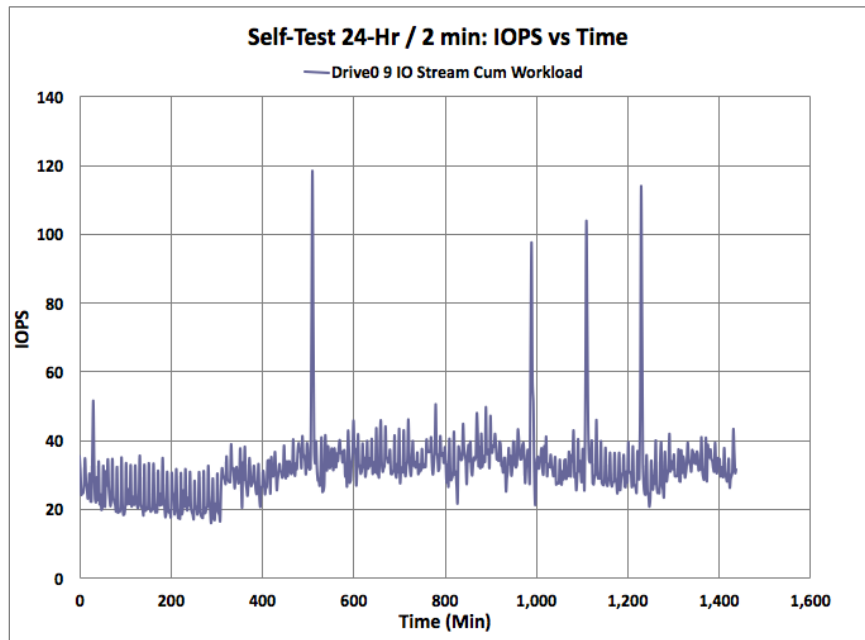
Figure 6-5. Self-Test 24-Hr/2 Min: IOPS v Time 24-Hr

6.3.6 Throughput v Time: 24-Hr Plot

The test operator shall report Throughput (TP) in MB/s over the 24-hour capture period. TP shall be averaged over each IO Capture step with each of the IO Capture steps plotted on the x-axis.
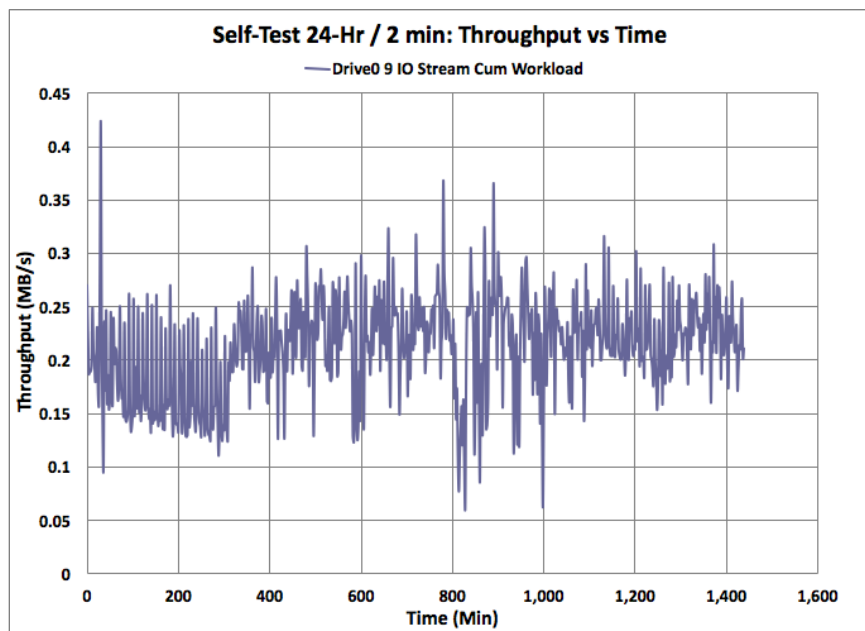
Figure 6-6. Self-Test 24-Hr/2 Min: Throughput v Time 24-Hr

6.3.7 Latency v Time: 24-Hr Plot

The test operator shall report Latency (Response Times) v Time for the 24-hour capture period. The test operator shall report Average and Maximum Response Times in mSec.
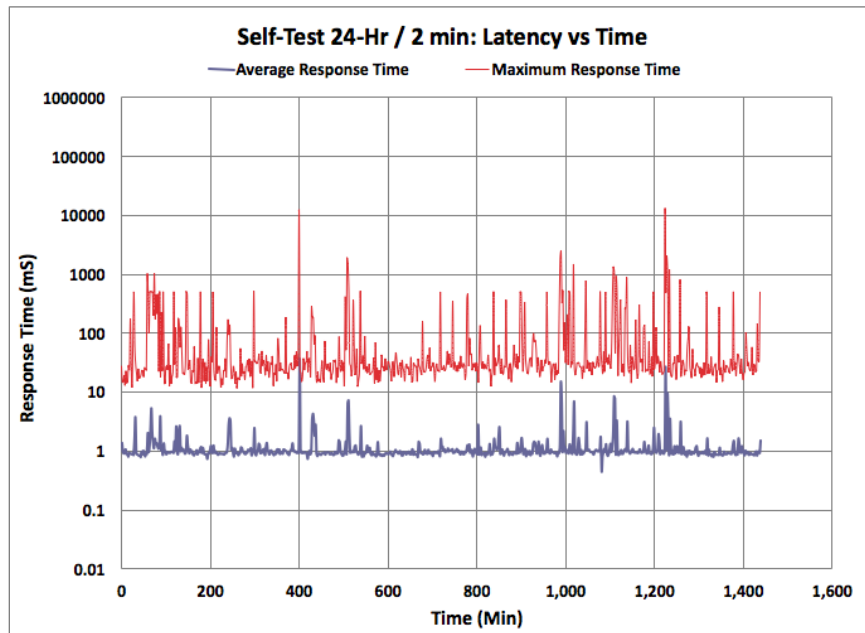
Figure 6-7. Self-Test 24-Hr/2 Min: Latency v Time 24-Hr

6.3.8 IOPS & ART: 24-Hr Average

The test operator shall report the average IOPS & ART v Time for the 24-hour capture period.
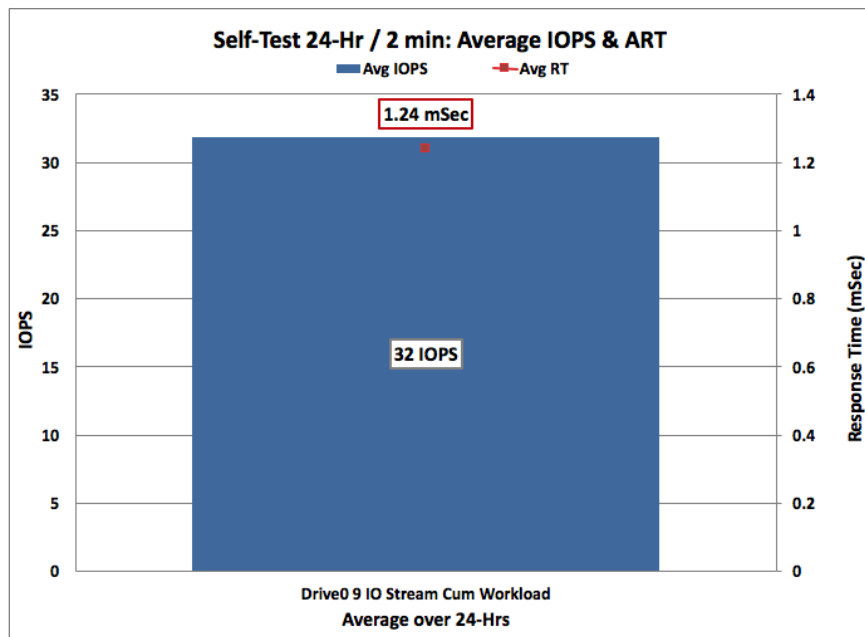
Figure 6-8. Self-Test 24-Hr/2 Min: Average IOPS & ART

6.3.9 Throughput & Ave QD: 24-Hr Average

The test operator shall report the average Throughput (TP) & QD for the 24-hour capture period.

Figure 6-9. Self-Test 24-Hr/2 Min: Average TP & QD

Note: 24-Hr average values reflect periods of low disk utilization, low application IO count and low Demand Intensity (QD). Compare Self-Test values to Replay-Native results that follow.

7   Replay-Native Test

7.1   Replay-Native Test Descriptive Note

General Purpose:

The Replay-Native test reproduces each IO Capture step combination of IO Streams, QDs and Idle times that occur during the original IO Capture to create a replay test workload. The Replay-Native test allows the test sponsor to evaluate test storage using the same workload IO Streams and settings as were observed on the target server during the IO Capture.

Note:

Target server Self-Test results are typically lower than lab testing of test storage using Replay-Native test workloads. This is because Target Server Self-Test results are subject to the IO throttling and pe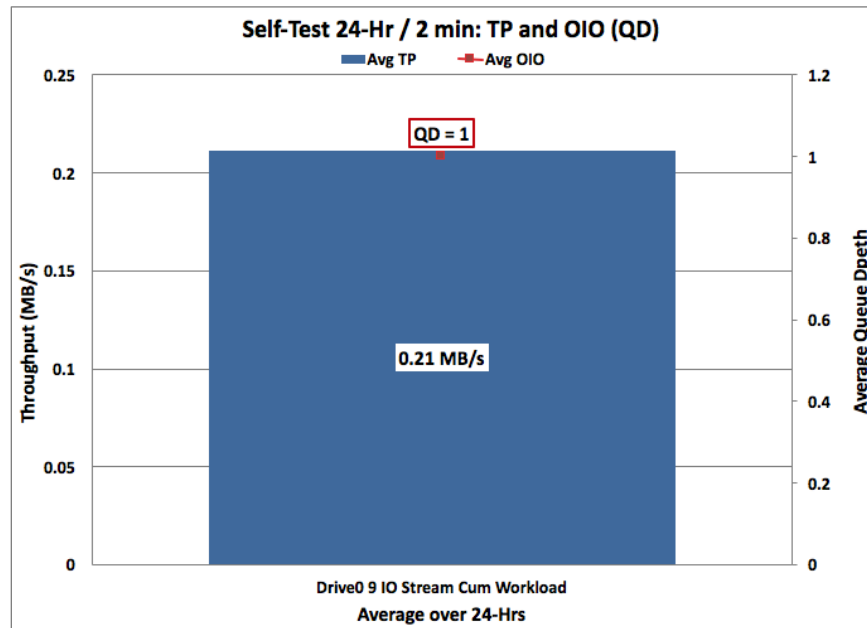riods of low disk utilization and Demand Intensity that occur on the target server during the IO Capture while lab testing of Replay-Native workloads do not limit storage IOs.

The test operator may choose to increase Demand Intensity by scaling the QD since the original IO Capture QD is averaged over the IO Capture step (and thus includes periods of low disk utilization). Typical native average QD values can be less than QD=4. Replay test QD settings can be increased to ensure saturation of the test storage for full performance evaluation.

Test Flow:

1. **Set Parameters & Conditions**
2. **Purge** – optional
3. **Pre-conditioning** – optional
   a. Workload independent PC - writing twice the User Capacity with SEQ 128K W
   b. Workload dependent PC – apply Cumulative Replay Workload 1 min Rounds separated by 30 minutes of in-between Round writes OR
   c. Use of the Section 9 RWSW DIRTH test and Replay Native PURGE disabled
4. **Steady State** – required. Run Cum Replay Workload until 5 Round Steady State is met
5. **Run Replay workload** – run IO Combinations, QDs and Idle times in the same sequence and duration as the original IO Capture.

Test Results:

The test results presented for the Replay-Native test are similar to the test results presented for the Self-Test. This allows the test operator and reader to easily compare in-situ Self-Test performance with the RWSW tests that are applied to test storage.

Each step of the IO Capture is reproduced to apply the same combinations of IO Streams, QDs and Idle times as observed during the original IO Capture. Demand Intensity is set as 'native' meaning that QD are equal to the observed average QD of each IO Capture step. The test operator may optionally change Demand Intensity settings to a 'fixed' value or to a 'scaled' value where the observed native QD is multiplied by a scaling factor.

Test Interpretation:

The Replay-Native test assesses how storage responds to the same RWSW as observed during the IO Captures. Target Server Self-Test workloads reflect application IO throttling. Use of native (average) QD settings may be insufficient to generate full storage performance. The replay of idle times can help the test sponsor evaluate garbage collection and other flash management algorithms while analysis of IO bursts can help in overall storage optimization.

Note: The Replay-Native test reproduces each IO Capture step IO Stream combination. Each step can have any number of IO Streams. Individual step IO Stream content will be different than the Cumulative 9 IO Stream Workload which averages the IO Streams across the entire capture. The Cumulative Workload 9 IO Stream distribution is used to present Replay-Native IO Stream distribution because of the difficulty in listing IO Stream combinations for every step of a Replay-Native test that has many, many steps (and different IO Stream combinations).

7.2   Replay-Native Pseudo Code

```
For RWSW Test, default AR=100.  Optional AR=User Selection.
  1 Prepare the Array A of [test step elements] Set test parameters and record
    for later reporting.
 1.1  Each element corresponds to specific time interval of captured data and
      has the following properties:
      1.1.1     Time Stamp
      1.1.2     Duration
      1.1.3     Thread Count (TC)
      1.1.4     Queue Depth (QD)
      1.1.5     Array S of [0 or many] streams, each having the following
                properties:
                1.1.5.1    RND or SEQ access flag
                1.1.5.2    Read or Write flag
                1.1.5.3    Block Size
                1.1.5.4    Probability of Occurrence
  2 Prepare Applied Test Workload
 2.1  Select IO Capture IO Streams for Cumulative Workload.
 2.2  Applied Test Workload is defined as the set of IO streams (or periods of
      no IO Streams aka idle periods), consisting of IO streams of the capture,
      which are filtered by IO Stream Threshold, time range, process ID, event,
      logical storage or other User defined criteria.
 2.3  IO Stream Threshold – recommended=3% or higher IO occurrence during
      capture
  3 Purge the storage=optional. Run steps 4 and 5 below OR use the Section 9
    RWSW DIRTH test as pre-conditioning with Replay Native PURGE=Disable.
    (Note: ActiveRange Amount and other Test Parameters are not applicable to
    Purge steps 4 & 5 below; any values can be used and none need to be
    reported.)
  4 Workload Independent Pre-conditioning
 4.1  Set and record parameters for later reporting
      4.1.1 Volatile Write cache: enabled (WCE)
      4.1.2 Thread Count: TC=1.
      4.1.3 OIO/Thread: Test Operator Choice* (recommended QD=32)
      4.1.4 Data Pattern: Required = Random, Optional = Test Operator Choice
 4.2  Run SEQ WIPC – Write 2X User Capacity with User selected (128KiB SEQ
      writes or SEQ 1024KiB writes) to the entire ActiveRange (AR=0,100)
 4.3  Run Workload Dependent Pre-conditioning and Test Stimulus
  5 Workload Dependent Pre-conditioning
 5.1  Set parameters and record for later reporting
      5.1.1 Volatile Write cache: enabled (WCE)
      5.1.2 Thread Count: Same as in step 4.1 above.
      5.1.3 OIO/Thread: Same as in step 4.1 above.
      5.1.4 Data Pattern: Required = Random, Optional = Test Operator Choice
      5.1.5 Compressibility Ratio = User selectable, must disclose
            compression tool and compression value
      5.1.6 Duplication Ratio = User selectable, must disclose duplication
            tool and duplication value
 5.2  Run the following until Steady State is reached, or maximum of 25
      Rounds
      5.2.1 Using Applied Test Workload (the set of IO streams of Cumulative
            Workload):
            5.2.1.1  Execute the workload for 1 minute
```

```
                    5.2.1.2  Record Ave MB/s
                    5.2.1.3  Use Ave MB/s to detect Steady State.
            5.2.2  If Steady State is not reached by Round x=25, then the Test
                   Operator may 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.
            5.2.3  Note that the accesses shall be continuous and use the entire
                   ActiveRange between test steps.
   6  Replay-Native Steps [Test Stimulus]
      6.1  Set parameters and record for later reporting
            6.1.1  Volatile Write cache: enabled (WCE)
            6.1.2  Thread Count: Same as in step 2.1 above.
            6.1.3  OIO/Thread: Same as in step 2.1 above.
            6.1.4  Data Pattern: Required = Random, Optional = Test Operator Choice
      6.2  For each element of array A:
            6.2.1  Thread Count: value from step element
            6.2.2  Queue Depth: value from step element
            6.2.3  Duration: value from step element
            6.2.4  Run the workload consisting of set of IO streams S
            6.2.5  Record IOPS, MB/s, RTs and other required metrics
   7  Process and plot the accumulated Rounds data

End (For ActiveRange) loop
```

7.3   Test Specific Reporting for Replay-Native Test

Figures 7.3.1 through 7.3.12 list the reporting requirements specific to the Replay-Native. Note that the test operator may select to report required data in tabular or plot format.

Reporting requirements common to all tests are documented in Section 4.3. The Replay-Native reports that follow are based on the Applied Test Workload in Section 5.1 IO Stream Map and Section 5.4 Applied Test Workload above.

7.3.1 Purge Report

Purge is optional. If the storage is Purged, the Test Operator shall report the method used to run the Purge operation. If Section 9 RWSW DIRTH test is used as an optional pre-conditioning, the Replay Native Purge setting shall be set to 'disabled'.

Note: The Section 9 RWSW DIRTH test is recommended as an optional pre-conditioning for the Replay Native test. This is because the target storage capacity may be such that pre-conditioning may be impractical or imprecise due to the storage architecture and/or total capacity of the storage. Use of the RWSW DIRTH test ensures a repeatble duration pre-condition with the captured workload applied across a range of OIO until Steady State is reached.

7.3.2 Steady State Measurement

Pre-conditioning is optional. Steady State is required. The test operator shall run the stimulus for the capacity or time set forth in the pseudo code Section 0 above OR until Steady State is achieved by calculating a five Round average as defined in 2.1.61 using one-minute test periods separated by 30 minutes of stimulus.
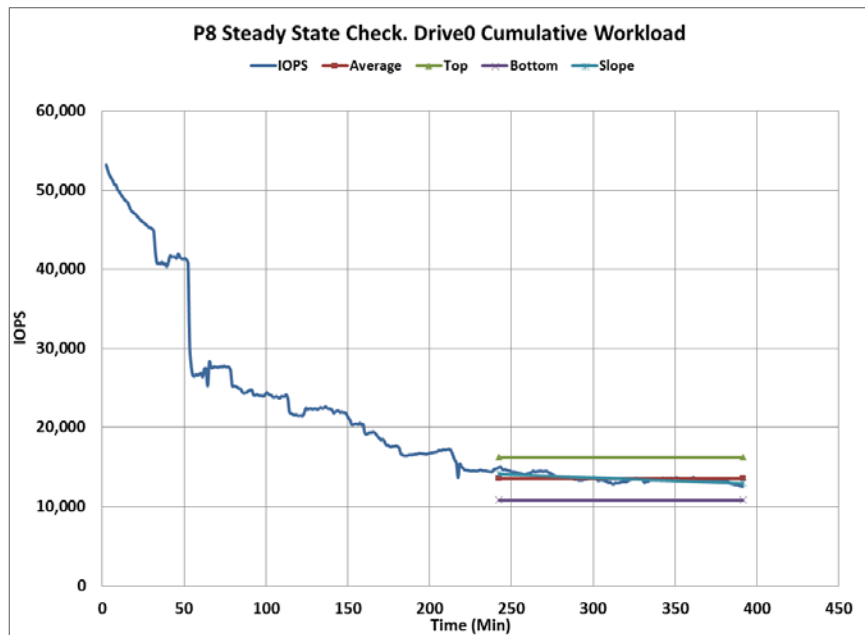
Figure 7-1. Steady State Check - Cum Workload Drive0

7.3.3 IO Streams Distribution

The Test Operator shall report the Applied Test Workload Cumulative Workload. Note that each test step will have a unique combination of IO Streams different from the Cumulative Workload. The Cumulative Workload IO Stream distribution is used because it is impractical to list the IO Stream combinations for each of the numerous Replay-Native test steps.
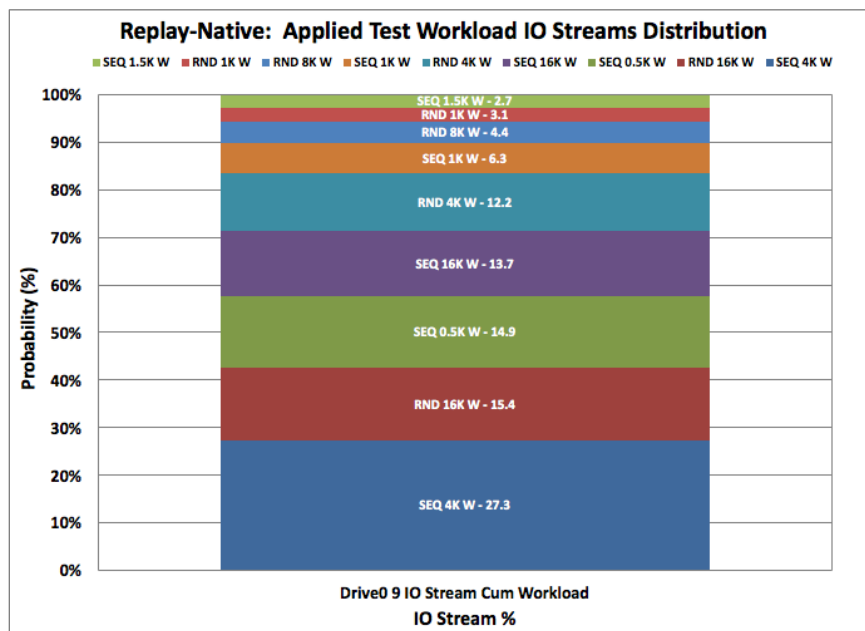

Figure 7-2. Applied Test Workload: IO Streams Distribution

7.3.4 IO Streams Map by Quantity of IOs

The test operator may report IO Streams, IOs and IOPS by IO Capture step over Time. The IO Stream Map by Quantity of IOs below shows the changing combination of IO Streams, IO count and IOPS for each IO Capture step. Compare Figure 7-3 below with the IO Stream Map by Quantity of IOs for the Self-Test in Figure 6.3.2 above.
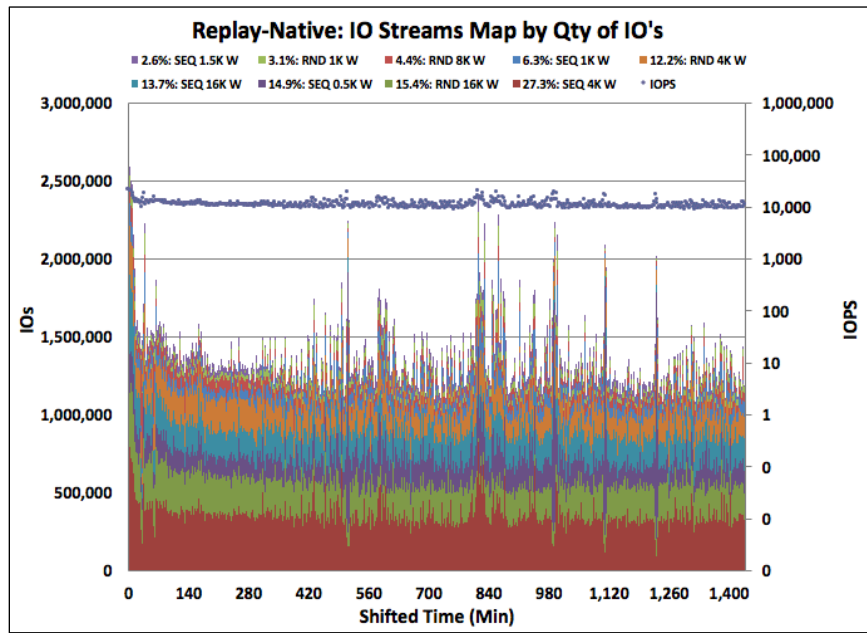
Figure 7-3. IO Streams Map by Quantity of IOs

7.3.5 Probability of IO Streams by Quantity of IOs

The test operator may report the Probability of IO Streams by Quantity of IOs for each IO Capture step over time. This shows the probability of occurrence percent for each IO Stream at each step.
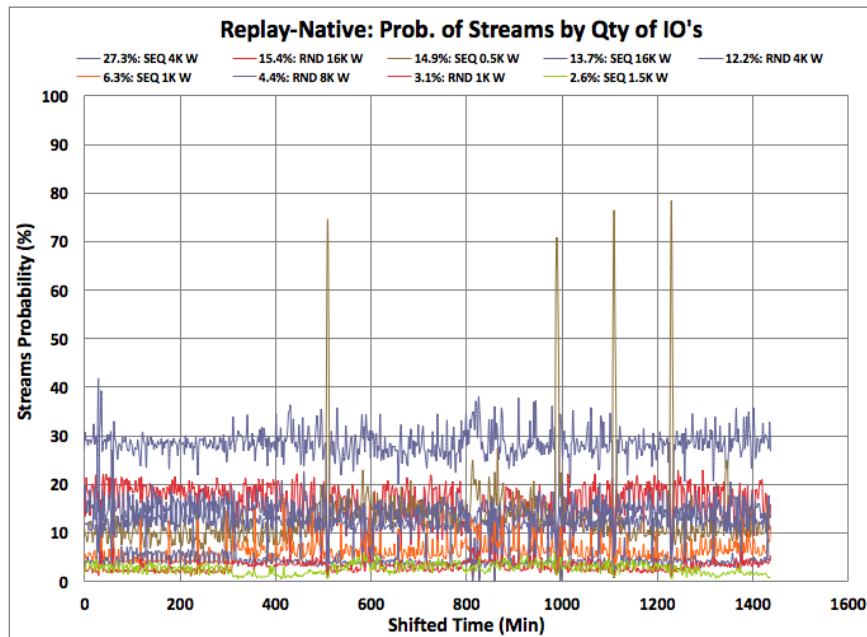

Figure 7-4. Probability of IO Streams by Quantity of IOs

7.3.6 Throughput & Queue Depths v Time: 24-Hr Plot

The test operator shall report Throughput (TP) & Queue Depths (QD) v Time for each test step.
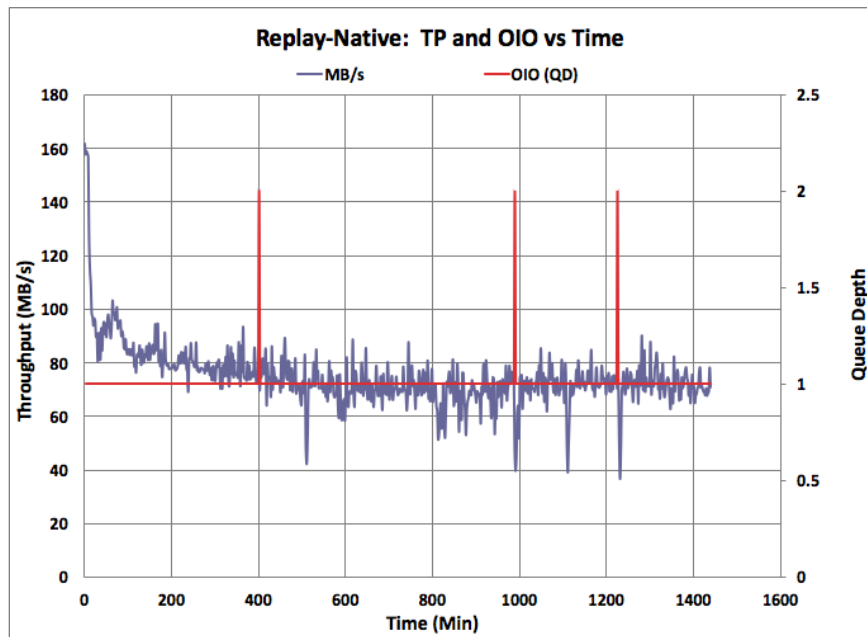
Figure 7-5. TP & OIO (Average QD from IO Capture)

Note: Native QD test settings reflect periods of low disk utilization and low application IO count. Replay-Native QD settings are therefore low (QD=1-2) and may not present sufficient Demand Intensity to saturate the test storage.

7.3.7 IOPS v Time: 24-Hr Plot

The test operator shall report IOPS v Time for the 24-hour capture period by plotting the average IOPS for each test step v Time. Note that each step is comprised of a unique combination of IO Streams that are reproduced from the original IO Capture.
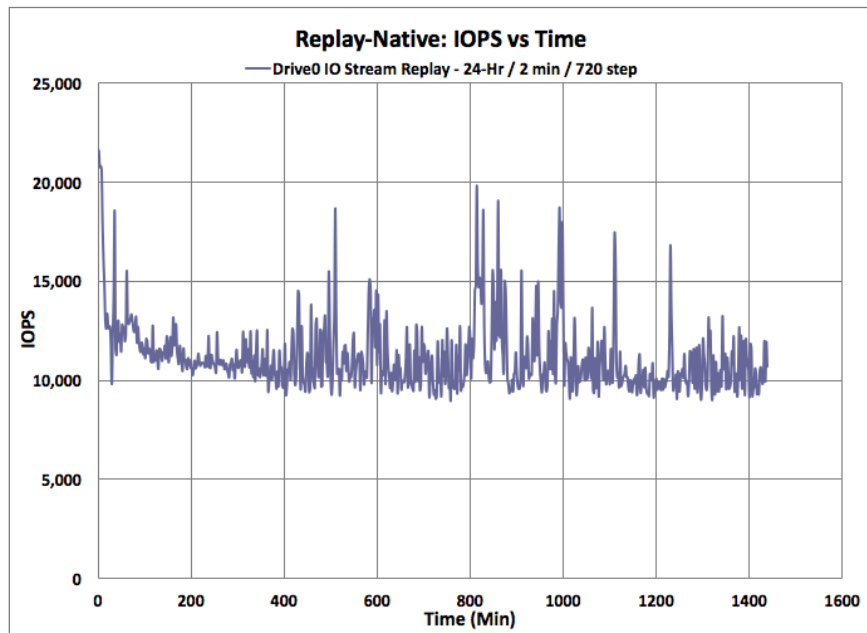

Figure 7-6. Replay-Native: IOPS v Time

7.3.8 Throughput v Time: 24-Hr Plot

The test operator shall report Throughput (TP) v Time for the 24-hour capture period. TP shall be reported in MB/sec.
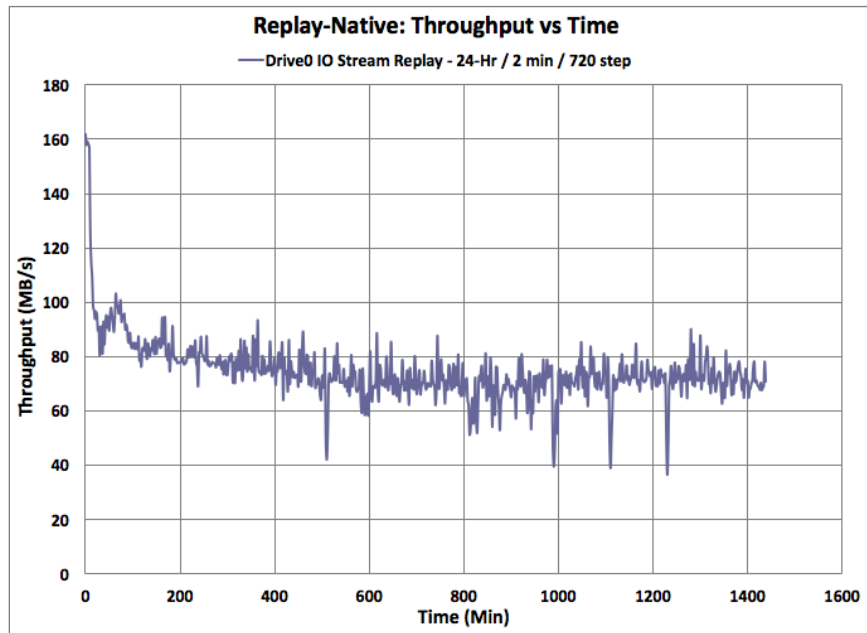
Figure 7-7. Replay-Native: TP v Time

7.3.9 Latency v Time: 24-Hr Plot

The test operator shall report Latency (Response Times) v Time for the 24-hour capture period. The test operator shall report Average, 5 9's and Maximum Response Times in mSec. 5 9's Response Time is also referred to Response Time Quality of Service (RT QoS).
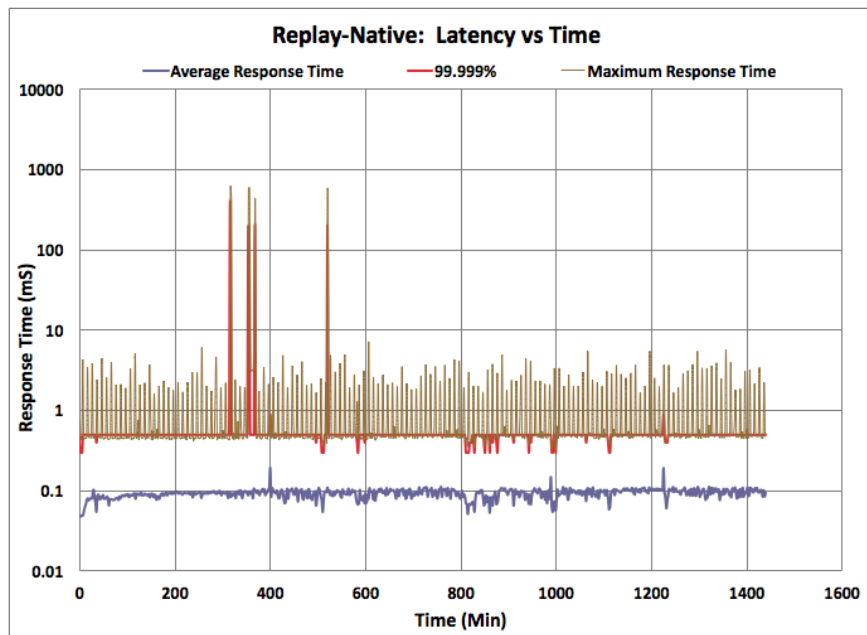
Figure 7-8. Replay-Native: Latency v Time

7.3.10 IOPS & Power v Time: 24-Hr Plot

The test operator may optionally report IOPS & Power (in mW) v Time for the 24-hour capture period. Storage Power Measurement requires suitable power measurement software and hardware and the ability to record storage power consumption for every IO command.
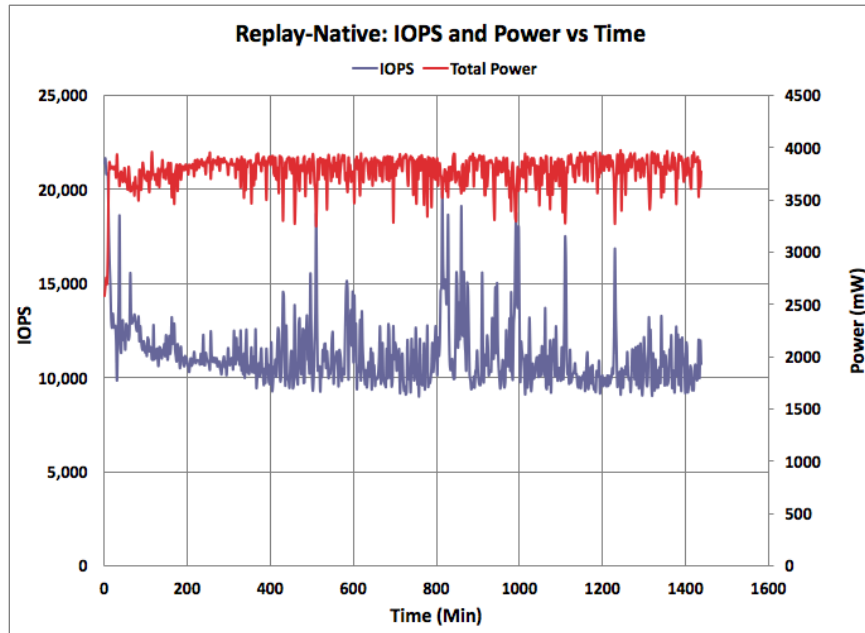

Figure 7-9. Replay-Native: IOPS & Power v Time

7.3.11 IOPS & Response Times: 24-Hr Average

The test operator shall report the average IOPS & Average, 5 9's, and Maximum Response Times in mSec averaged over the 24-hour capture period.
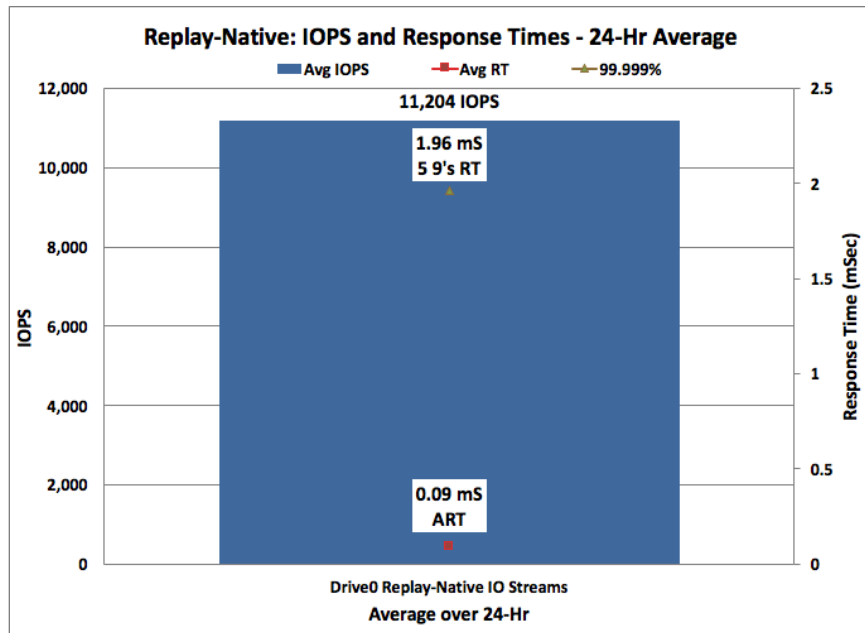

Figure 7-10. IOPS & Response Times - Average over 24-Hr

7.3.12 Throughput & Ave QD: 24-Hr Average

The test operator shall report the average Throughput (TP) in MB/s for the 24-hour capture period. The test operator may report Power Consumption in mW averaged over the 24-hour capture period if IO command power measurement is supported by the test software and hardware.
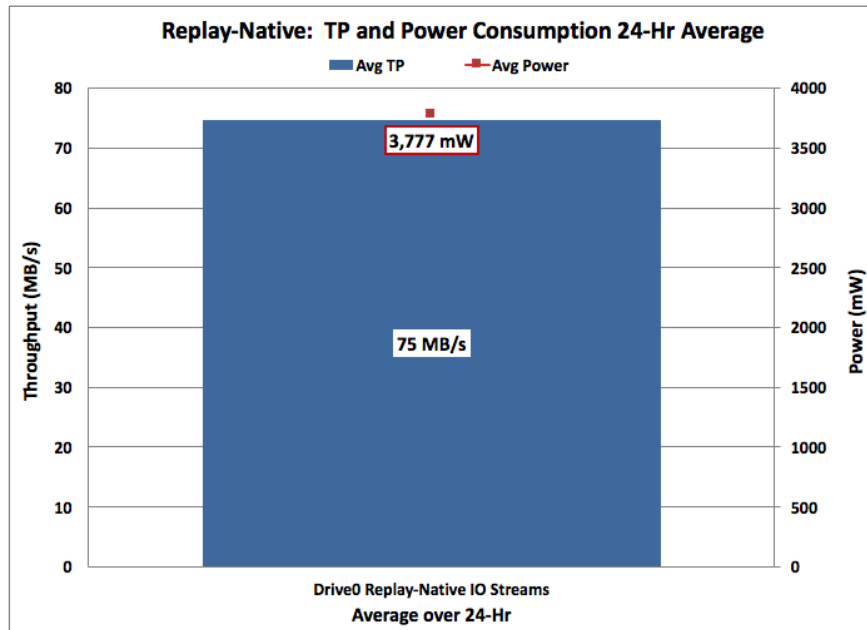

Figure 7-11. TP & Power - Average over 24-Hr

8   Multi-WSAT Test

8.1   Multi-WSAT Test Descriptive Note

General Purpose:

The Multi-WSAT test applies a multiple IO Stream workload to Steady State to provide a 'single number' RWSW performance value. This 'blend' or 'composite' of IO Streams is comprised of the IO Streams defined in the Applied Test Workload. This composite IO Stream workload provides a fixed and constant workload for performance comparison among test storage.

Note:

Multi-WSAT creates a fixed and constant workload from the Applied Test Workload IO Streams. In this test, the 9 IO Stream composite from Section 5.4 Applied Test Workload is applied for each Multi-WSAT test step in the same probability of occurrence as the original IO Capture. See **Figure 8-2. Applied Test Workload: IO Streams Distribution**.

Test Flow:

1. **Set Parameters & Conditions**
2. **Purge** – optional
3. **Pre-conditioning** – optional
   a. Workload independent PC - writing twice the User Capacity with SEQ 128K W
   b. Workload dependent PC – apply Applied Test Workload 1 min Rounds
      separated by 30 minutes of in-between Round writes
4. **Steady State** – required
5. **Run the test stimulus to Steady State –** Run the 9 IO Stream Applied Test Workload until the 5 Round Steady State criteria is met
6. **Process & Plot data** – compile and generate results plots, figures and tables as specified

Test Results:

Multi-WSAT shows IOPS, TP and Latency performance both as a single Steady State value as well as performance evolution over time.

Test Interpretation:

The Multi-WSAT tests shows how the Applied Test Workload Composite IO Stream workload performance evolves over time, Steady State convergence, and Steady State performance.

8.2   Multi-WSAT Pseudo Code

```
Basic Test Flow:
   For (ActiveRange=100 or other specified values)
       1. Purge the Storage
           1.1 Purge is optional for arrays but required for devices
           1.2 Note: Test Operator may use any values for ActiveRange and Test
               parameters for this step; no parameter reporting is required.
       2. Run the Workload Independent Pre-conditioning
           2.1 SEQ 128KiB Writes for twice user capacity.
           2.2 Note: Test Operator shall use specified ActiveRange ("For ActiveRange
               ="), but may choose, and report, other Test Parameter values to
               optimize this step
       3. Run the Test Stimulus (includes Workload Based Pre-conditioning).
           3.1 Set and record test parameters:
               3.1.1 Device volatile write cache = Enabled/Disabled – as specified
               3.1.2 Thread Count/Queue Depth: as specified (default 4/32, Native or
                     Scaled)
           3.2 Run the Test Stimulus (aggregated composite workload consisting of
               several RND/BS/RW components (IO Streams)) until 4X User Capacity is
               written, 8 hours, or five Round Steady State (per PTS 2.0) is reached,
               whichever occurs first.
```

```
      4. Record IOPS, MB/s, Response Times, and other required metrics.
      5. Process and plot the accumulated data.
   End "For ActiveRange"
```

8.3 Test Specific Reporting for Multi-WSAT Test

Figures 8.3.1 through 8.3.9 list the reporting requirements specific to the Multi-WSAT test. Note that the test operator may select to report required data in tabular or plot format.

Reporting requirements common to all tests are documented in Section 4.3. The Multi-WSAT reports that follow are based on the Applied Test Workload of Demo No. 6 'GPS Navigation Portal - 24-Hr/2min 9 IO Stream – Drive0 Cumulative Workload.'

8.3.1 Purge Report

Purge is optional. If the storage is Purged, the Test Operator shall report the method used to run the Purge operation.

8.3.2 Steady State Measurement

Pre-conditioning is optional. Steady State is required. The test operator shall run the stimulus for the capacity or time set forth in the pseudo code 8.2 above OR until Steady State is achieved by calculating a five Round average as defined in 2.1.61 using one-minute test periods separated by 30 minutes of stimulus.
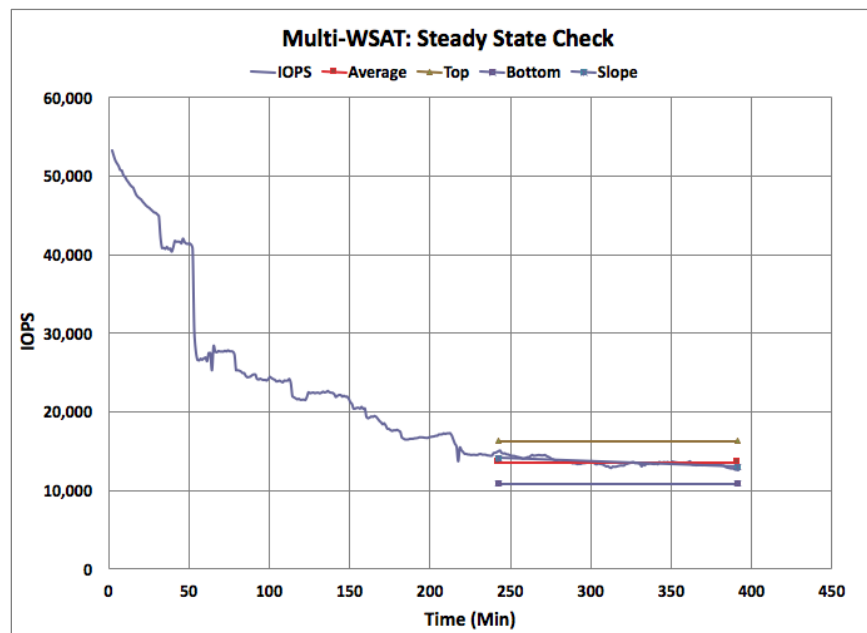

Figure 8-1. Steady State Check - Cum Workload 9 IO Stream Drive0

8.3.3 IO Streams Distribution

The Test Operator shall report the Applied Test Workload Cumulative Workload. Note that each test step will have the same fixed composite IO Stream Workload.
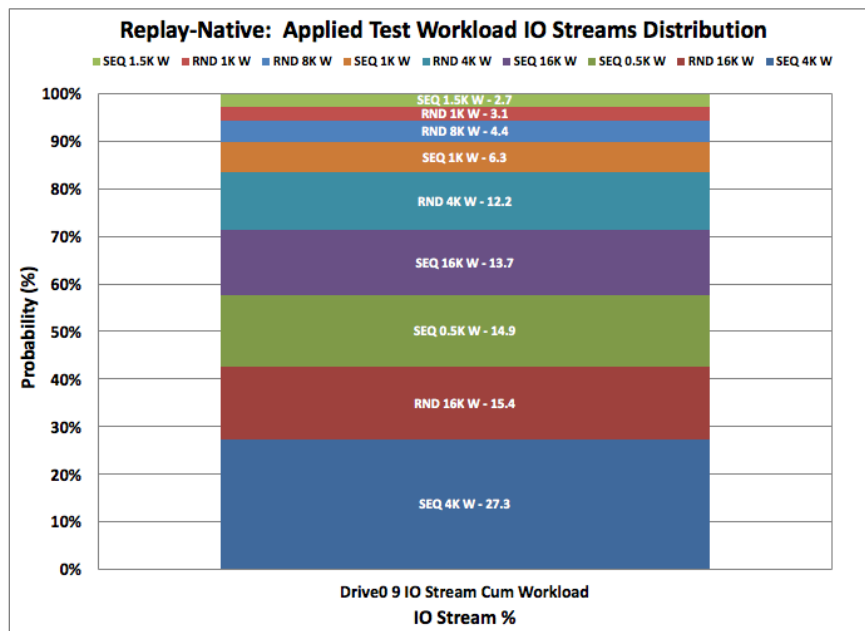

Figure 8-2. Applied Test Workload: IO Streams Distribution

8.3.4 IOPS v Time

The test operator shall report IOPS v Time. Note that each test step is comprised of the same Applied Test Workload 9 IO Stream combination.
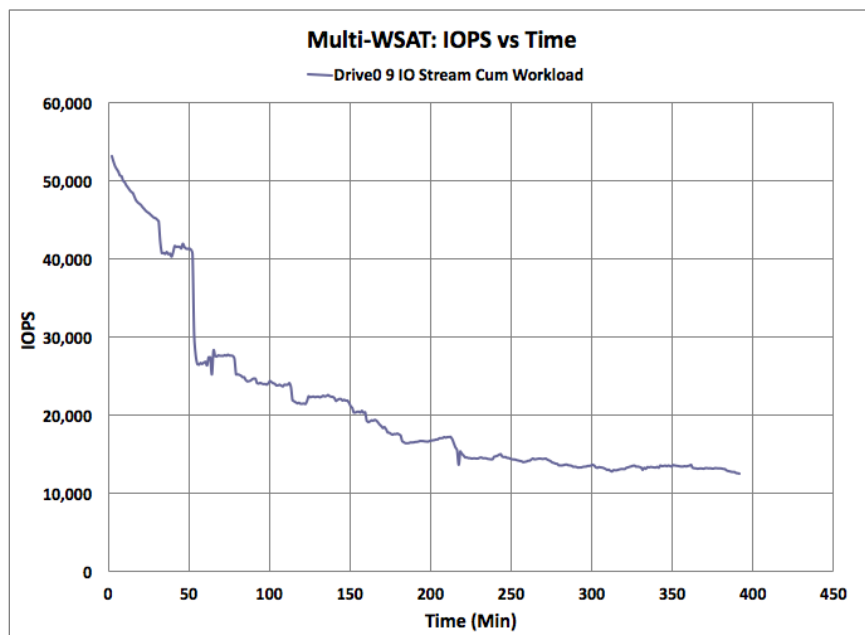

Figure 8-3. Multi-WSAT: IOPS v Time

8.3.5 Throughput v Time

The test operator shall report Throughput (TP) in MB/s v Time.
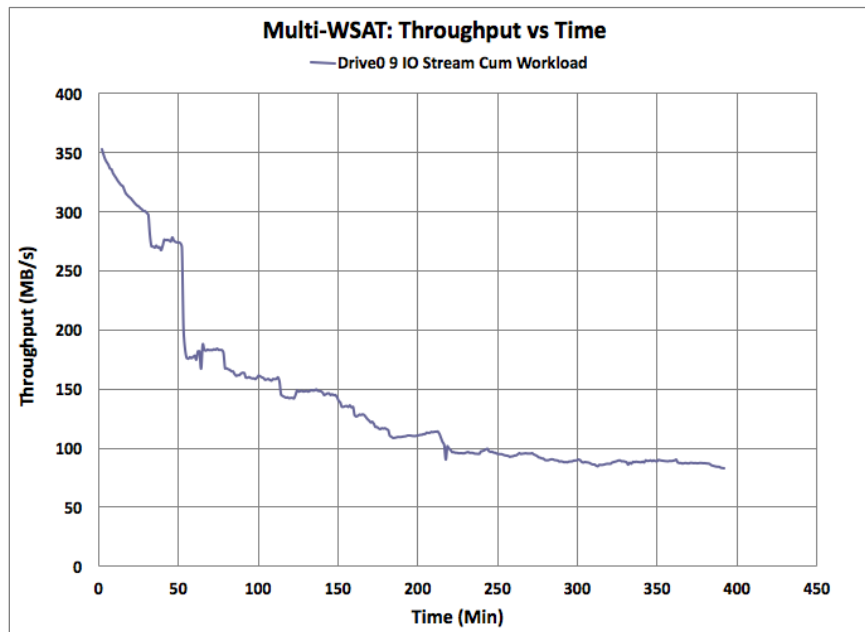

Figure 8-4. Multi-WSAT: Throughput v Time

8.3.6 Latency v Time

The test operator shall report Latency (Response Times) v Time. The test operator shall report Average, 5 9's and Maximum Response Times in mSec.
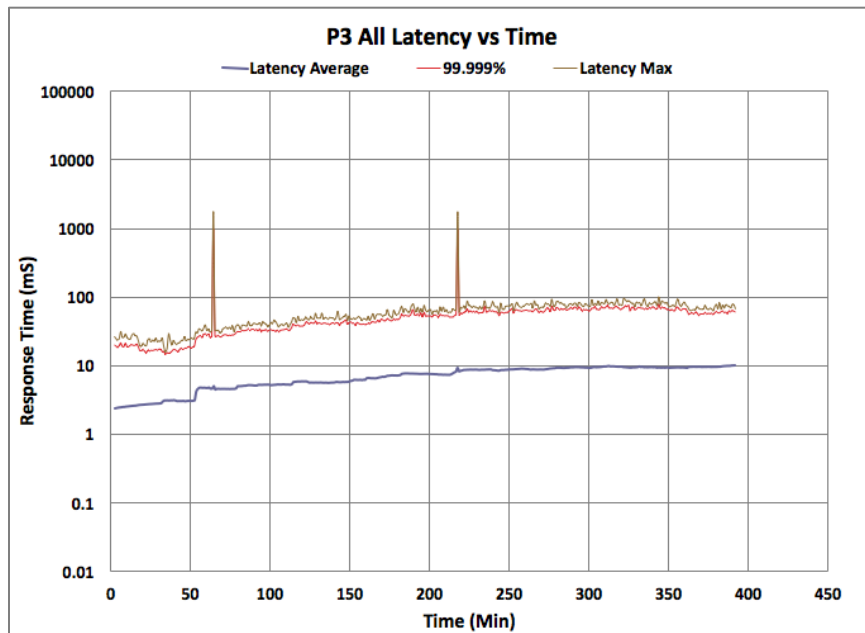

Figure 8-5. Multi-WSAT: Latency v Time

8.3.7 IOPS & Response Times: Steady State Value

The test operator shall report the average IOPS & Average, 5 9's, and Maximum Response Times in mSec averaged over the 24-hour capture period.
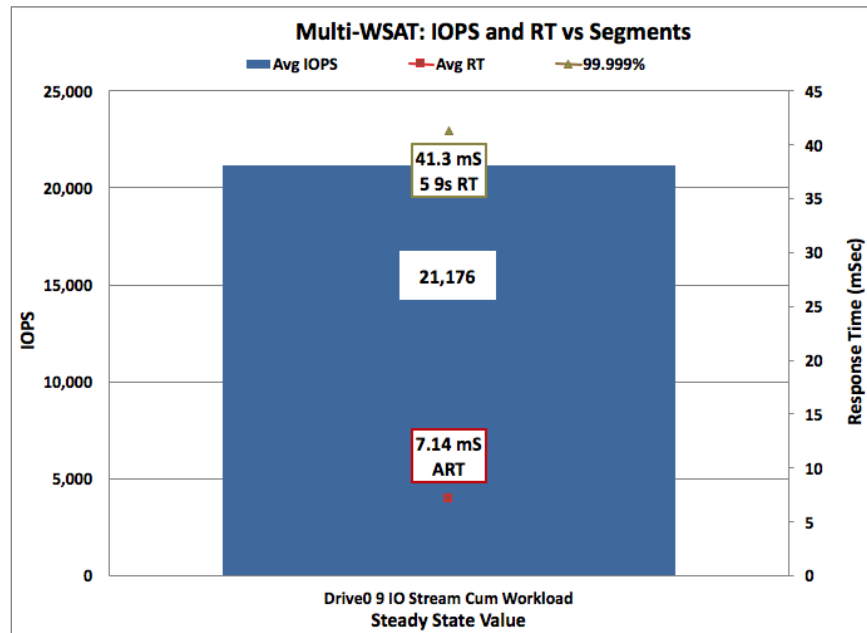

Figure 8-6. Multi-WSAT: IOPS & Response Times – Steady State Value

8.3.8 Throughput & Power Consumption: Steady State Value

The test operator shall report the average Throughput (TP) in MB/s. The test operator may report Power Consumption in mW averaged over the 24-hour capture period if power measurement for every IO command is supported by the test software and hardware.
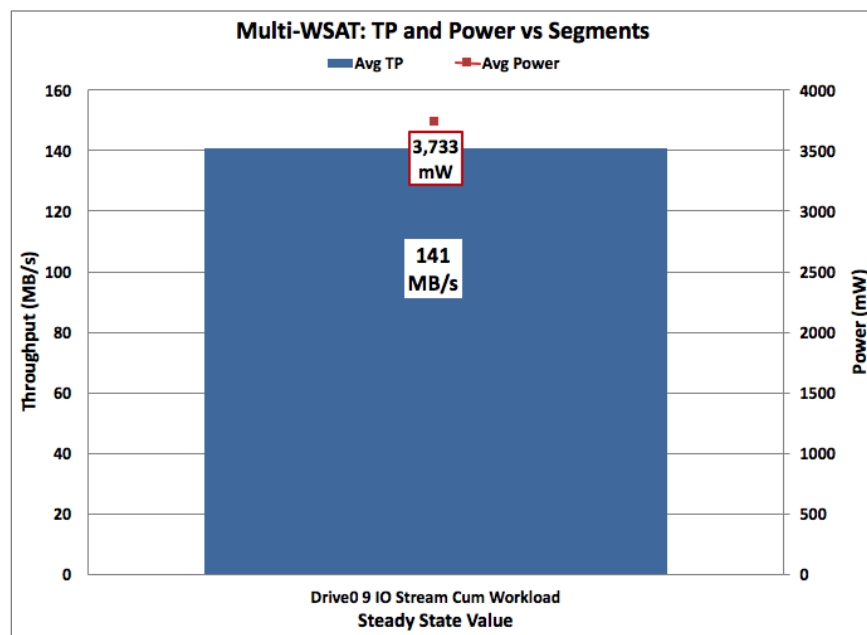

Figure 8-7. Throughput & Power Consumption v Time - Steady State Value

8.3.9 IOPS & Total Power v Time

The test operator may optionally report IOPS & Power (in mW) v Time for the Multi-WSAT test period. Storage Power Measurement requires suitable power measurement software and hardware and the ability to record storage power consumption for every IO command.
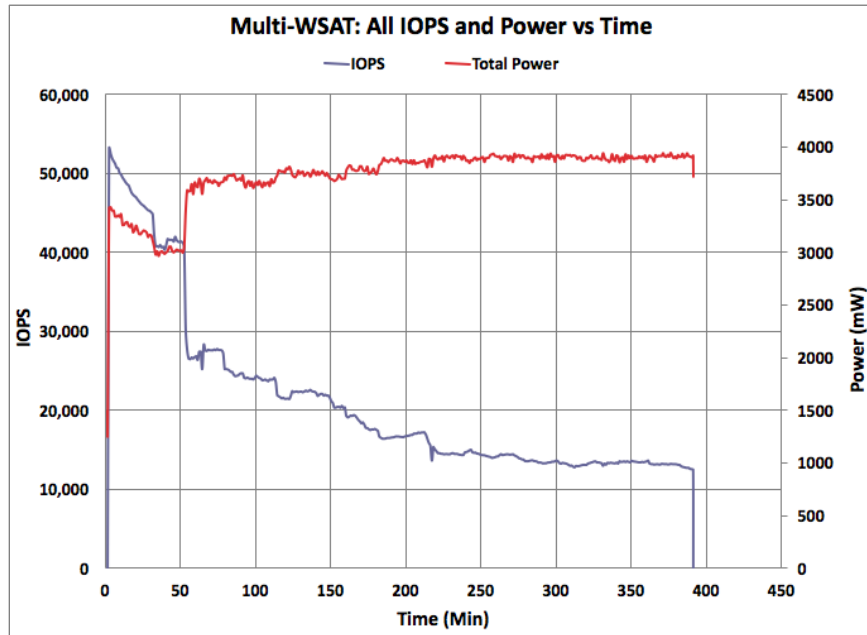

Figure 8-8. Replay-Native: IOPS & Power v Time

9   Individual Streams-WSAT Test

9.1   Individual Streams-WSAT Test Descriptive Note

General Purpose:

The Individual Streams-WSAT test applies each component (individual) IO Stream workload to Steady State. This allows the test operator to compare individual IO Stream WSAT values to single access pattern corner case benchmark, or manufacturer specification, test results.

Note:

Each of the individual IO Stream components defined in the Applied Test Workload is run as a separate WSAT test to Steady State. Each WSAT segment is run serially with a host idle period separating adjacent WSAT segments.

Test Flow:

1. **Set Parameters & Conditions**
2. **Purge** – optional
3. **Pre-conditioning** – optional
   a. Workload independent PC - writing twice the User Capacity with SEQ 128K W
   b. Workload dependent PC – run the Applied Test Workload for 1 min Rounds
      separated by 30 minutes of in-between Round writes
4. **Steady State** – required
5. **Run the test stimulus to Steady State –** Run each independent IO Stream as a separate WSAT segment workload until 5 Round Steady State is met
6. **Insert Idle times** – insert host idle time between Individual Stream-WSAT segments
7. **Process & Plot data** – compile and generate results plots, figures and tables as specified

Test Results:

Individual Streams-WSAT shows IOPS, TP and Latency performance both as a single Steady State value as well as IO Stream performance evolution over time for each individual IO Stream.

Test Interpretation:

The Individual Streams-WSAT test presents Steady State performance for each of the component Applied Test Workload IO Streams. Differential WSAT segment performance can influence overall RWSW performance. For example, SQL Server workloads typically contain a significant percentage of SEQ 0.5K Writes. There can be a large performance variance among different SSDs for SEQ 0.5K Writes which can significantly impact ordinal ranking to RWSWs that contain SEQ 0.5K Writes.

9.2   Individual Streams-WSAT Pseudo Code

```
Basic Test Flow:
   For (ActiveRange=100 or other specified values)
      1. Purge the Storage
         1.1 Purge is optional for arrays but required for devices
         1.2 Note: Test Operator may use any values for ActiveRange and Test
             parameters for this step; no parameter reporting is required.
      2. Run the Workload Independent Pre-conditioning
         2.1 SEQ 128KiB Writes for twice user capacity.
         2.2 Note: Test Operator shall use specified ActiveRange ("For ActiveRange
             ="), but may choose, and report, other Test Parameter values to
             optimize this step
      3. Run the Test Stimulus (includes Workload Based Pre-conditioning).
         3.1 Set and record test parameters:
            3.1.1 Device volatile write cache = Enabled/Disabled – as specified
            3.1.2 Thread Count/Queue Depth: as specified (default 4/32, Native or
                  Scaled)
```

```
        3.2 Run the Test Stimulus (for each individual RND/BS/RW component (IO
            Stream)) until 4X User Capacity is written, 8 hours, or five Round
            Steady State (per PTS 2.0) is reached, whichever occurs first.
    4. Record IOPS, Response Times, etc.
    5. Process and plot the accumulated data.
End "For ActiveRange"
```

9.3   Test Specific Reporting for Individual Streams-WSAT Test

Figures 9.3.2 through 9.3.8 list the reporting requirements specific to the Individual Streams-WSAT test. Note that the test operator may select to report required data in tabular or plot format.

Reporting requirements common to all tests are documented in Section 4.3. The Individual Streams -WSAT reports that follow are based on the Section 5.4 Applied Test Workload.

9.3.1 Purge Report

Purge is optional. If the storage is Purged, the Test Operator shall report the method used to run the Purge operation.

9.3.2 Steady State Measurement

Pre-conditioning is optional. Steady State is required. The test operator shall run the stimulus for the capacity or time set forth in the pseudo code Section 9.2 above OR until Steady State is achieved by calculating a five Round average as defined in 2.1.61 using one-minute test periods separated by 30 minutes of stimulus.

Steady State Measurement Windows are presented below for each of the 9 IO Streams.
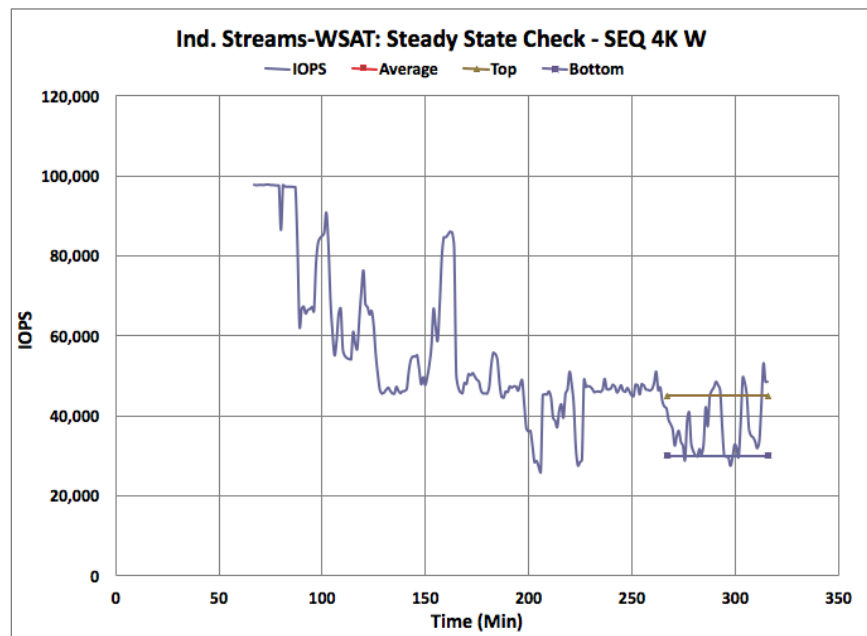

9.3.2.1   Steady State Check – SEQ 4K W

Figure 9-1. Ind. Streams-WSAT: Steady State Check - SEQ 4K W

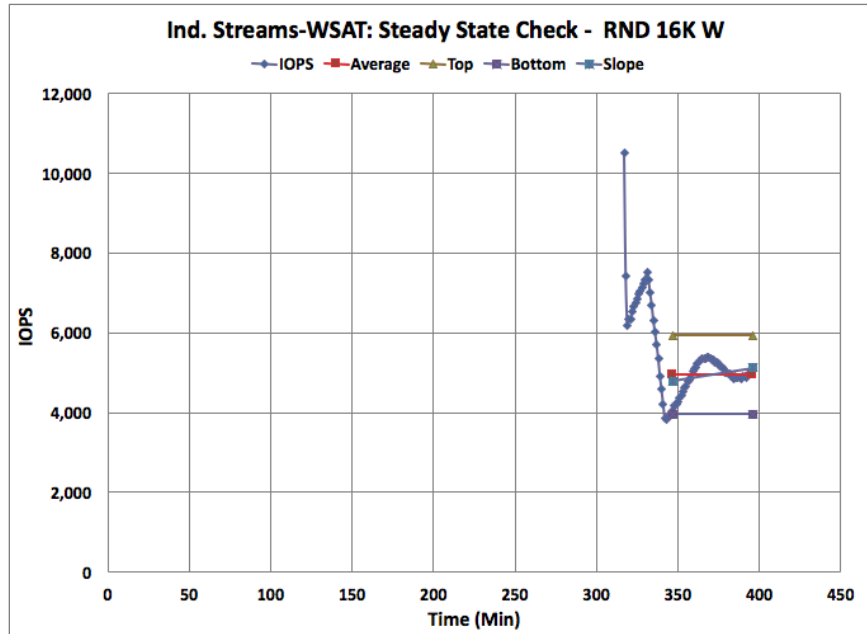### 9.3.2.2 Steady State Check – RND 16K W

Figure 9-2. Ind. Streams-WSAT: Steady State Check – RND 16K W
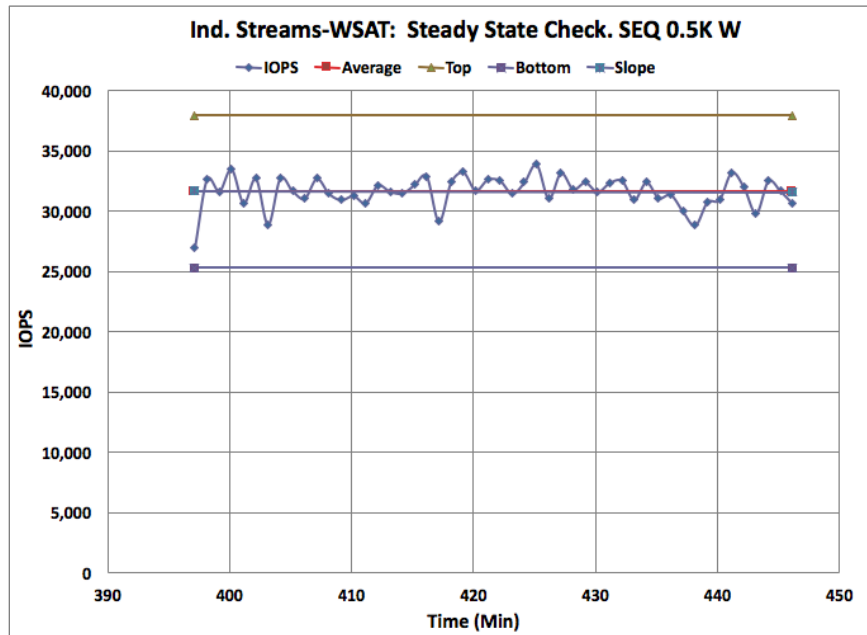
9.3.2.3 Steady State Check – SEQ 0.5K W

Figure 9-3. Ind. Streams-WSAT: Steady State Check – SEQ 0.5K W

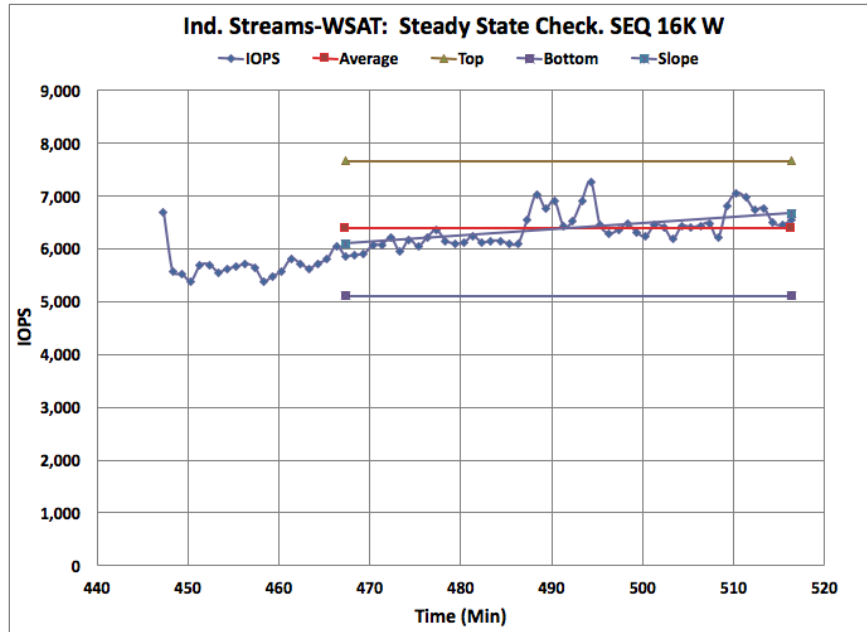### 9.3.2.4 Steady State Check – SEQ 16K W

Figure 9-4. Ind. Streams-WSAT: Steady State Check – SEQ 16K W
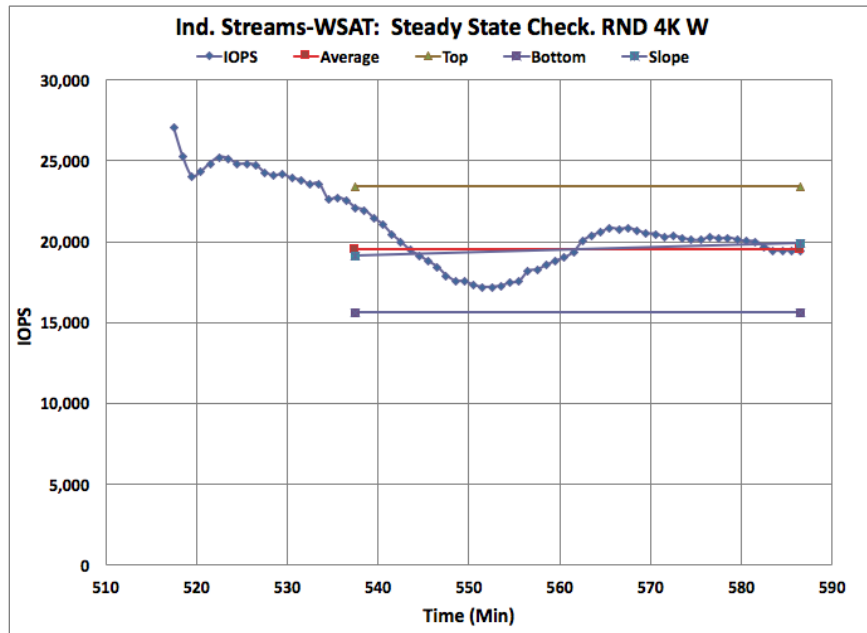
9.3.2.5 Steady State Check – RND 4K W

Figure 9-5. Ind. Streams-WSAT: Steady State Check – RND 4K W

9.3.2.6 Steady State Check – SEQ 1K W
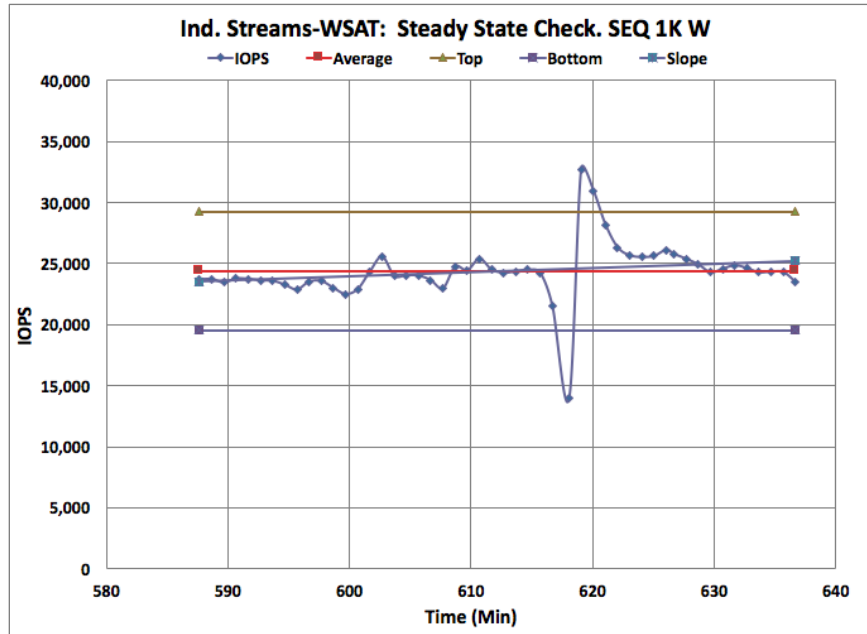

Figure 9-6. Ind. Streams-WSAT: Steady State Check – SEQ 1K W
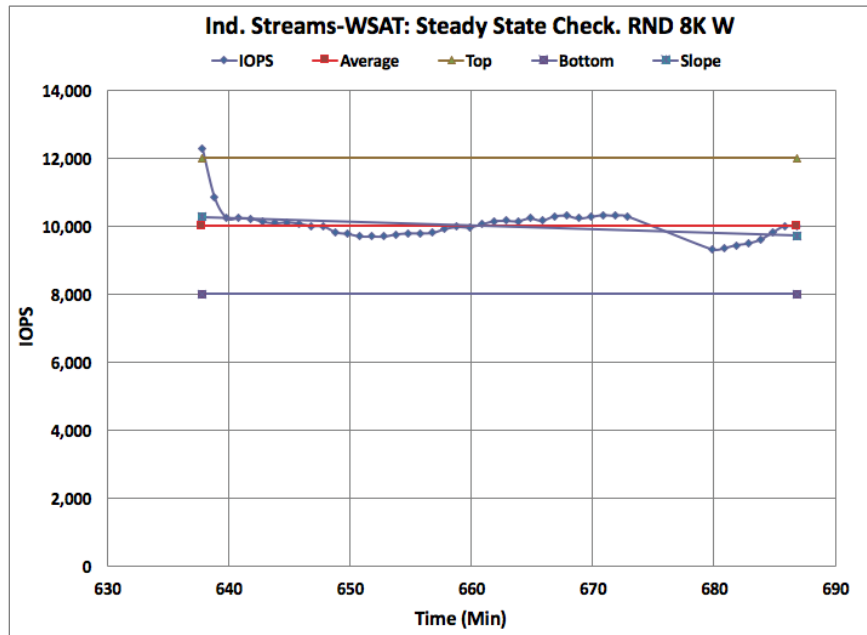
9.3.2.7 Steady State Check – RND 8K W

Figure 9-7. Ind. Streams-WSAT: Steady State Check – RND 8K W

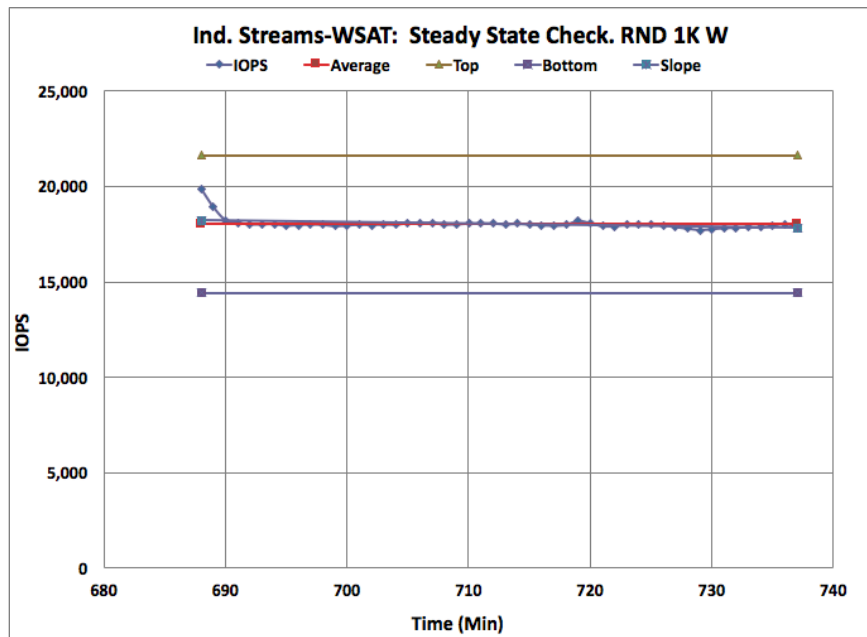### 9.3.2.8 Steady State Check – RND 1K W

Figure 9-8. Ind. Streams-WSAT: Steady State Check – RND 1K W

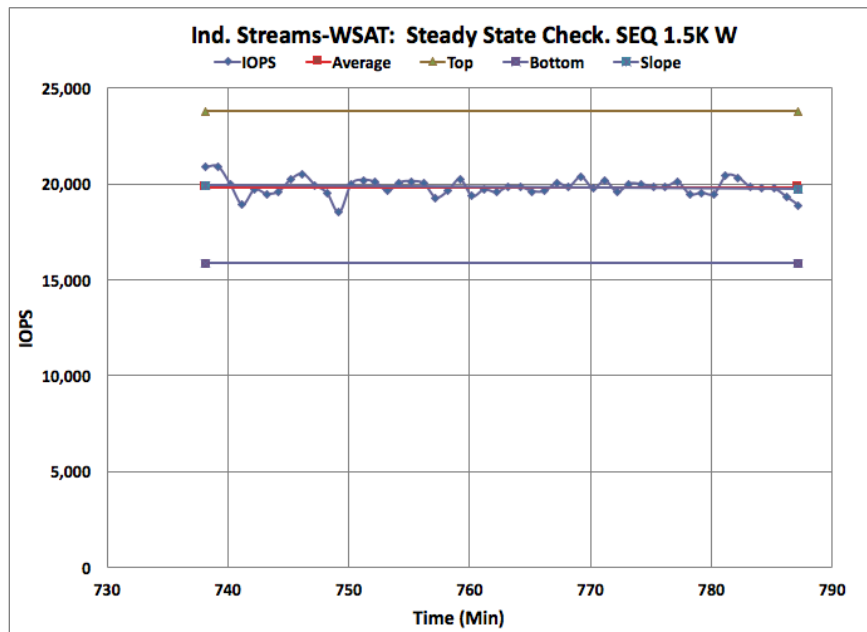### 9.3.2.9   Steady State Check – SEQ 1.5K W

Figure 9-9. Ind. Streams-WSAT: Steady State Check – SEQ 1.5K W

9.3.3 IO Streams Distribution

The Test Operator shall report the Applied Test Workload Cumulative Workload. Note that each test step will have only one of IO Streams listed in the Cumulative Workload.
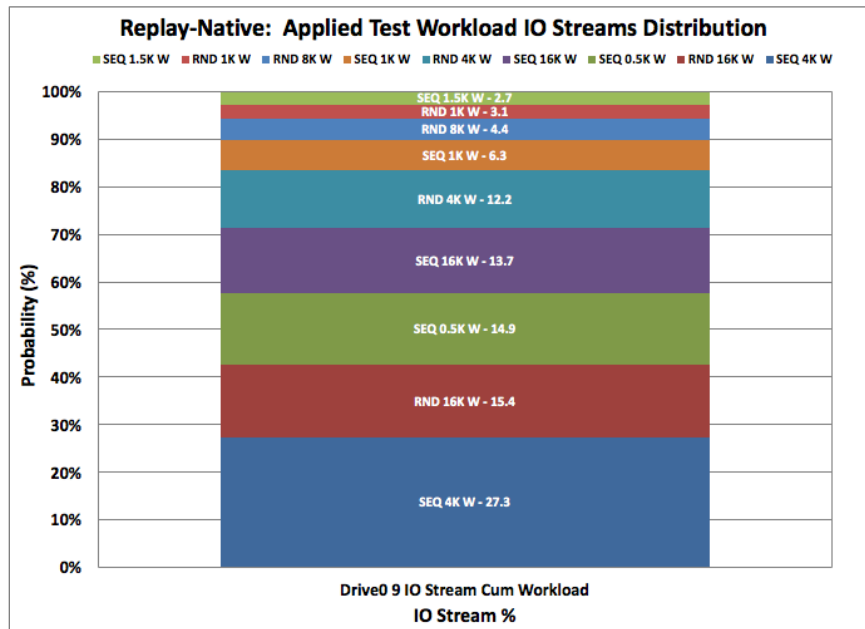
Figure 9-10. Applied Test Workload: IO Streams Distribution

9.3.4 IOPS & Response Times: Steady State Values

The test operator shall report the Steady State IOPS & Average Response Times in mSec for the Applied Test Workload IO Stream segments.
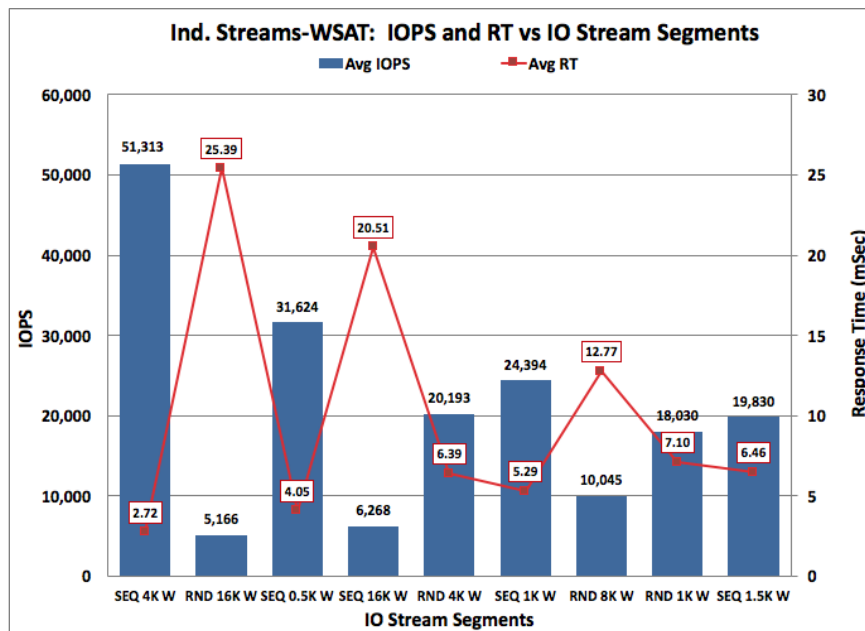
Figure 9-11. Multi-WSAT: IOPS & Response Times – Steady State Values

9.3.5 Throughput & Power Consumption: Steady State Values

The test operator shall report the Steady State Throughput (TP) in MB/s. The test operator may report Steady State Power Consumption in mW if power measurement for every IO command is supported by the test software and hardware.
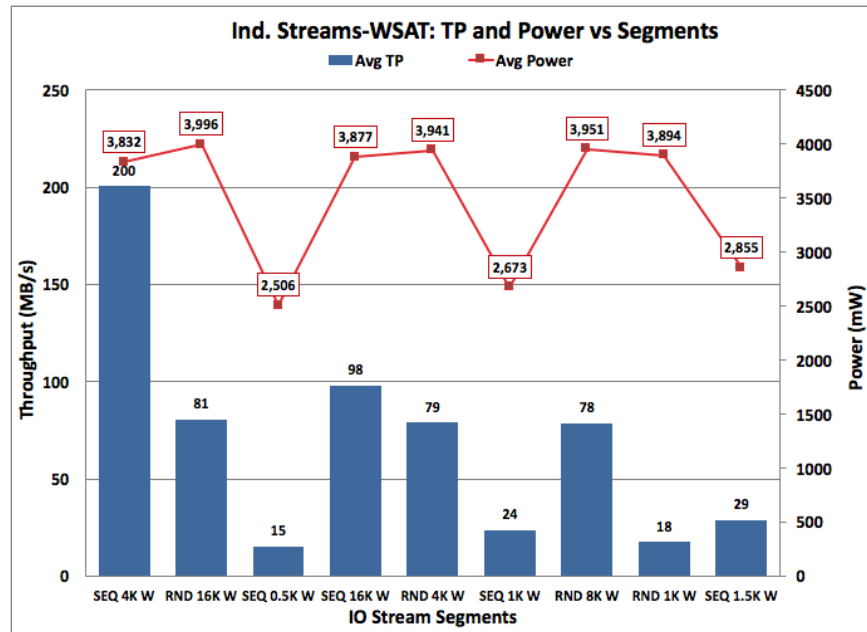

Figure 9-12. Throughput & Power Consumption v Time - Steady State Values

9.3.6 IOPS v Time

The test operator shall report IOPS v Time for each IO Stream segment. Segments show IOPS convergence to Steady State for each IO Stream.
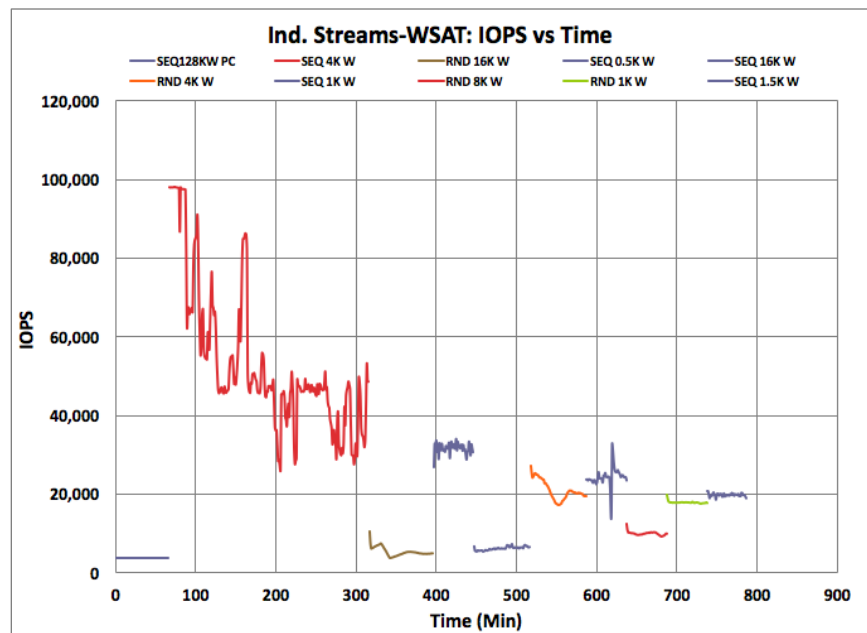
Figure 9-13. Individual Streams-WSAT: IOPS v Time

9.3.7 Throughput v Time

The test operator shall report Throughput (TP) in MB/s v Time for each IO Stream segment. Segments show TP convergence to Steady State for each IO Stream.
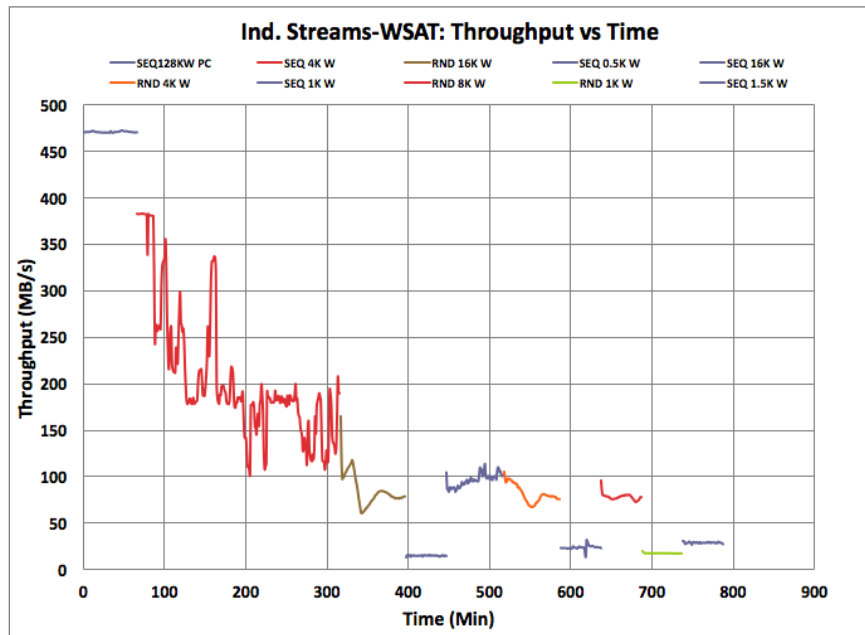

Figure 9-14. Individual Streams-WSAT: Throughput v Time

9.3.8 Latency v Time

The test operator shall report Latency (Response Times) v Time for each IO Stream segment. The test operator shall report Average, 5 9's and Maximum Response Times in mSec.
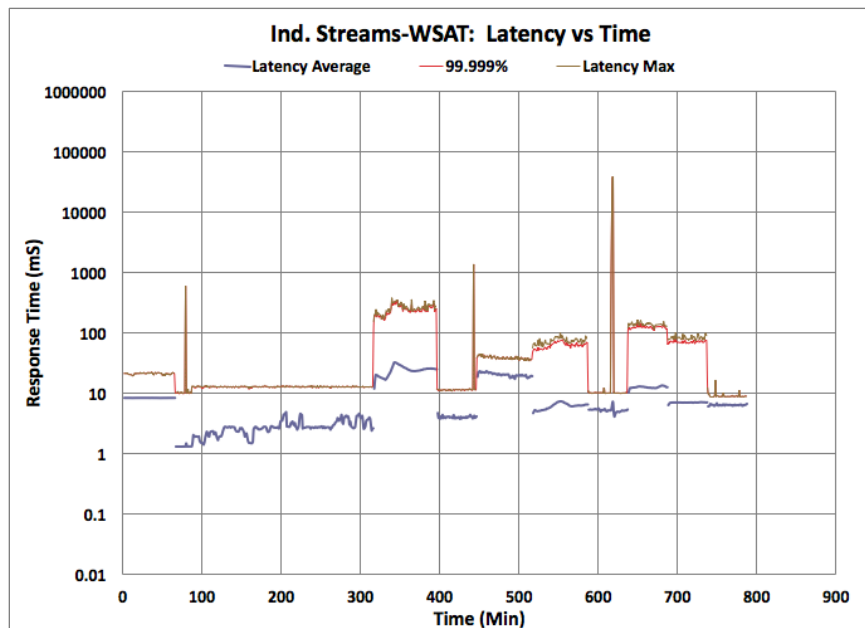

Figure 9-15. Individual Streams -WSAT: Latency v Time

10 RWSW Demand Intensity / Response Time Histogram

10.1 RWSW DIRTH Test Descriptive Note:

General Purpose:

The RWSW DIRTH test applies the composite Applied Test Workload IO Stream workload to Steady State and then varies the OIO across a range of 1 User to 1,024 Users. This allows the test operator to evaluate the test storage range of performance and Response Time Saturation.

Note:

The Applied Test Workload is the same workload used in the Multi-WSAT test.

Test Flow:

1. **Set Parameters & Conditions**
2. **Purge** – optional
3. **Pre-conditioning** – optional
       a. Workload independent PC - writing twice the User Capacity with SEQ 128K W
       b. Workload dependent PC – run the Applied Test Workload for 1 min Rounds
          separated by 30 minutes of in-between Round writes
4. **Steady State** – required
5. **Run the test stimulus to Steady State –** Run the Applied Test Workload while varying the Total Outstanding IOs by applying an outer loop of High to Low Thread Count (TC) by an inner loop of High to Low Queue Depth (QD) with the application of an inter loop Pre-Write between each TOIO loop until Steady State, as defined, is reached for the TOIO tracking variable.
6. **Total OIO (TOIO) tracking variable** – run the TOIO loop in descending order of TC/QD (from 1,024 to 1) in one-minute steps. Use T32Q32 IOPS as the Steady State tracking variable.
7. **Set Min, Mid, Max and Max Prime OIO** - Select a MAX IOPS point representing an operating point where the IOPS is maximum while achieving a reasonable ART; select a MIN IOPS point where TC=1 and QD=1; and select a minimum of 1 additional MID IOPS point(s), using the (Thread Count, OIO/Thread) operating points obtained during the test run such that their IOPS values lie between and equally divides the IOPS value between MinIOPS and MaxIOPS; and select the Max Prime OIO TC/QD combination where Maximum IOPS are observed without regard to response times.
9. **Process & Plot data** – compile and generate results plots, figures and tables as specified.

Test Results:

RWSW DIRTH shows Demand Variation (IOPS as a function of Thread Count and Queue Depth), Demand Intensity (Average Response Times as a function of increasing OIO and IOPS), Confidence Level Plot Compare (Response Time Quality of Service (5 9s RT), IOPS and specific TC/QD settings for min, mid, max and max prime IOPS), and IOPS and Bandwidth & ART v Total OIO (showing ART, IOPS and MB/s saturation as OIO increases).

Test Interpretation:

RWSW DIRTH performance v OIO shows the range of performance and RT saturation of the test storage. This helps the test operator to estimate test storage performance to the Applied Test Workload at various TOIO as well as to observe performance and saturation at the QD range observed in the original IO Capture on the target server.

10.2 RWSW DIRTH Pseudo Code

```
Basic Test Flow:
   For (ActiveRange=100, optional ActiveRange=Test Operator Choice, Test Workload
   = Applied Test Workload (aggregated composite workload consisting of several
   RND/BS/RW components (IO Streams))
       1. Purge the Storage. (optional for arrays, required for devices)
          1.1  Note: Test Operator may use any values for ActiveRange and Test
               parameters for this step; no parameter reporting is required.
```

2. **Pre-conditioning using the Test Workload but with RW Mix=0% (writes)**
    2.1 Set test parameters and record for later reporting
        2.1.1 Volatile device write cache = Enabled (WCE)
        2.1.2 Thread Count: 4
        2.1.3 Queue Depth: 32
        2.1.4 Data Pattern: Required = Random, Optional = Test Operator
               Choice

    2.2 Run Test Workload, using the required ActiveRange=100% or the
         corresponding desired optional ActiveRange.
        2.2.1 Record elapsed time, IOPS, Average Response Time (ART) and
               Maximum Response Time (MRT) every 1 minute.
        2.2.2 Using the first 1 Minute IOPS, along with subsequent 1 Minute
               IOPS results that are 30 Minutes apart, run Access Pattern
               until Steady State is reached, or until the maximum number of
               Rounds=25 has been reached.
3. **Run the Test Workload while varying demand settings**:
    3.1 Set test parameters and record for later reporting
        3.1.1 Volatile device write cache = Enabled (WCE)
        3.1.2 Data Pattern: Same as Pre-conditioning
        3.1.3 Vary TC using TC=[32,16,8,4,2,1]
        3.1.4 Vary QD using QD=[32,16,8,4,2,1]

    3.2 Apply Inter-Round Pre-Write
        3.2.1 Run the Applied Test Workload, using TC=32 and QD=32 for a
               minimum of 5 minutes and a maximum of either 30 minutes or 10%
               of the User Capacity, whichever occurring first.
        3.2.2 Record elapsed time, IOPS, ART, MRT and Percentage CPU
               Utilization by System (SYS_CPU) every 1 Minute.

    3.3 Apply One Round of the Applied Test Workload:
        3.3.1 Run the Applied Test Workload for 1 Minute at each TC/QD
               combination, in the order of decreasing TOIO from 1024 (32x32)
               to 1, using all of the TC/QD combinations that can be generated
               from TC and QD values. When multiple TC/QD combinations give
               rise to equal TOIO values, apply TC/QD combination with the
               higher TC first.
        3.3.2 Record elapsed time, IOPS, ART and MRT and Percentage CPU
               Utilization by System (CPU_SYS).
        3.3.3 Repeat the test loop described 3.3.1babove until Steady State
               is reached, using IOPS values for TC=32, QD=32 and Block Size
               and R/W Mix as specified in the Applied Test Workload as the
               tracking variable, or until the maximum number of Rounds=25 has
               been reached.
4. **Determine MaxIOPS, MinIOPS,1 MidIOPS operating point and MaxIOPS prime:**
    4.1 A MaxIOPS point shall be chosen from the (Thread Count, OIO/Thread)
         operating points, such that:
        4.1.1 The MaxIOPS point should be chosen to represent the operating
               point where the IOPS are highest while achieving a reasonable
               ART.
        4.1.2 The ART for such MaxIOPS point shall be below 5 mS.
    4.2 The MinIOPS point is defined to be the operating point where Thread
         Count=1 and OIO/Thread=1.
    4.3 Choose 1 additional MidIOPS point(s), using the (Thread Count,
         OIO/Thread) operating points obtained during the test run such that
         their IOPS values lie between and, as much as possible, equally
         divides the IOPS value between MinIOPS and MaxIOPS.
    4.4 Choose a MaxIOPS prime point where the highest IOPS are observed.
5. **Response Time Histograms at MaxIOPS:**
    5.1 Select a (Thread Count, Queue Depth) operating point that yields
         maximum IOPS using the lowest number of Total Outstanding IO
         (TOIO=Thread Count x Queue Depth)
    5.2 Run Pre-Writes

      5.2.1 Execute the Applied Test Workload for 60 minutes. Log elapsed
            time, IOPS, ART and MRT every 1 minute.
    5.3 Execute Applied Test Workload for 10 minutes. Capture all individual
       IO command completion times such that a response time histogram
       showing count versus time can be constructed. The maximum time value
       used in the capture shall be greater or equal to the MRT encountered
       during the 10-minute capture.
6. Response Time Histograms at MinIOPS:
    6.1 Select a (Thread Count=1, Queue Depth=1) operating point
    6.2 Run Pre-Writes
      6.2.1 Execute Applied Test Workload for 60 minutes. Log elapsed time,
          IOPS, ART and MRT every 1 minute
    6.3 Execute Applied Test Workload for 10 minutes. Capture all individual
       IO command completion times such that a response time histogram
       showing count versus time can be constructed. The maximum time value
       used in the capture shall be greater or equal to the MRT encountered
       during the 10-minute capture.
7. Response Time Histogram at one or more chosen MidIOPS operating points:
    7.1 Select a (Thread Count, Queue Depth) operating point that yields an
       IOPS result that lies approximately halfway between Maximum IOPS in
       (6) above, and the Minimum IOPS in (7) above.
    7.2 Run Pre-Writes
      7.2.1 Execute Applied Test Workload for 60 minutes. Log elapsed time,
          IOPS, ART and MRT every 1 minute.
    7.3 Execute Applied Test Workload for 10 minutes. Capture all individual
       IO command completion times such that a response time histogram
       showing count versus time can be constructed. The maximum time value
       used in the capture shall be greater or equal to the MRT encountered
       during the 10-minute capture.
8. Process and plot the accumulated data per report guidelines in next
   section.
End "For ActiveRange"

10.3 Test Specific Reporting for RWSW DIRTH Test

Figures 10.3.3.1 through 10.3.3.12 list the reporting requirements specific to the RWSW DIRTH test. Note that the test operator may select to report required data in tabular or plot format.

Reporting requirements common to all tests are documented in Section 4.3. The RWSW DIRTH test reports that follow are based on the Section 5.4 Applied Test Workload.

10.3.1 Purge Report

Purge is optional. If the storage is Purged, the Test Operator shall report the method used to run the Purge operation.

10.3.2 Steady State Measurement

Pre-conditioning is optional. Steady State is required. The test operator shall run the stimulus for the capacity or time set forth in the pseudo code Section 10.2 above OR until Steady State is achieved by calculating a five Round average as defined in 2.1.61 using one-minute test periods separated by 30 minutes of stimulus.

10.3.3 Measurement Report

The Test Operator shall generate Measurement Plots as specified in the following sub sections.

10.3.3.1 Applied Test Workload IO Streams Distribution

The Test Operator shall report the Applied Test Workload Cumulative Workload.
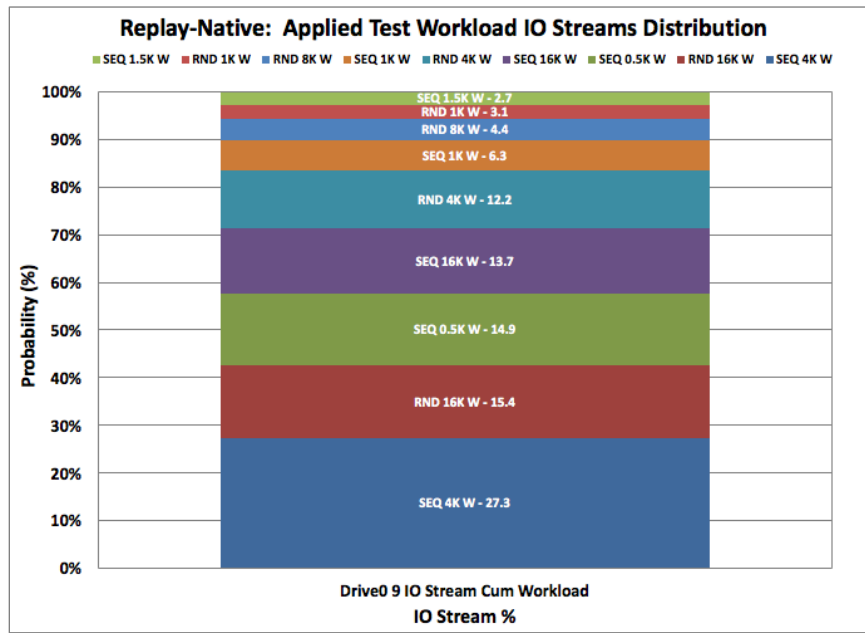
Figure 10-1. Applied Test Workload: IO Streams Distribution

10.3.3.2 Pre-conditioning

When Pre-conditioning is conducted, results shall be reported as IOPS v Time plot.
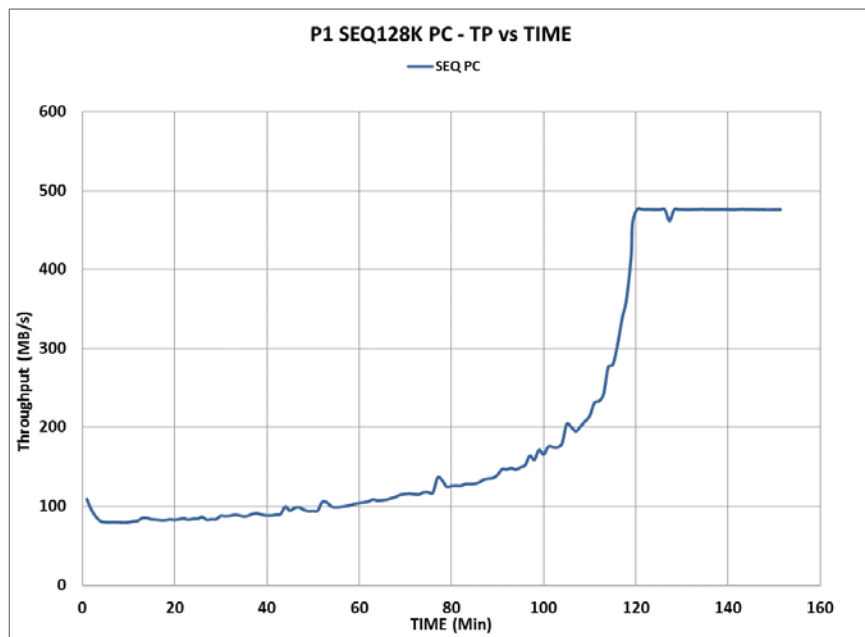

Figure 10-2. Applied Test Workload: IO Streams Distribution

10.3.3.3 IOPS v Time All Data

The Test Operator shall report IOPS v Time for All Data showing IOPS for each Thread Count and Queue Depth of the Applied Test Workload Total OIO test loops.
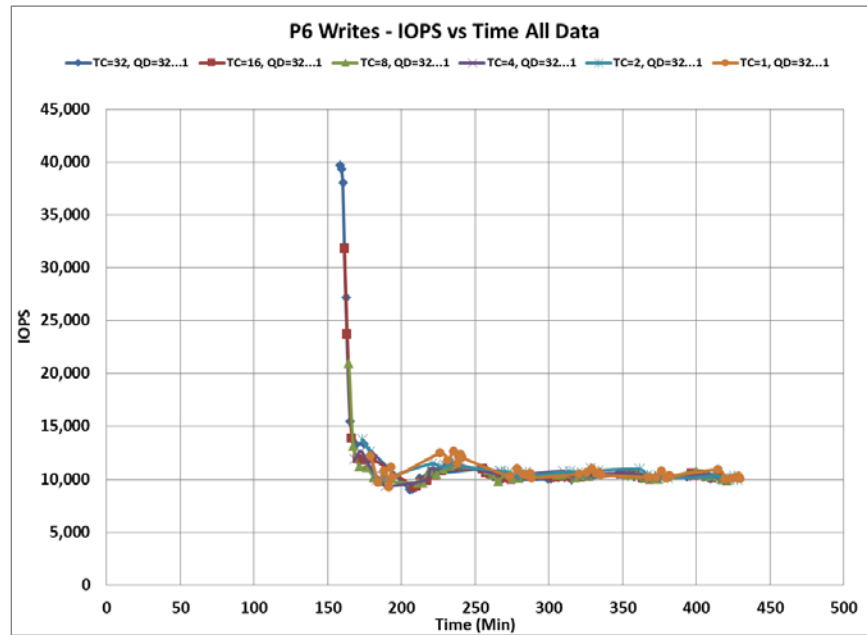

Figure 10-3. DIRTH: IOPS v Time All Data

10.3.3.4 Steady State Check – T32Q32

The Test Operator shall report the Steady State Measurement window at T32Q32.
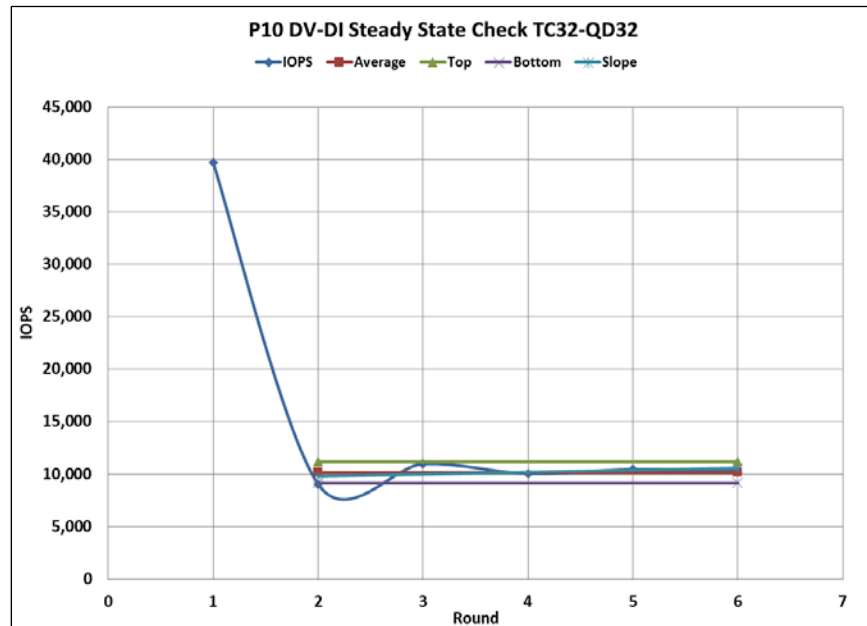
Figure 10-4. DIRTH: Steady State Check T32Q32

10.3.3.5 Demand Variation

The Test Operator shall report Demand Variation by presenting a plot of IOPS v Thread count and Queue Depth where each data series is a Thread Count with QD as the x axis points.
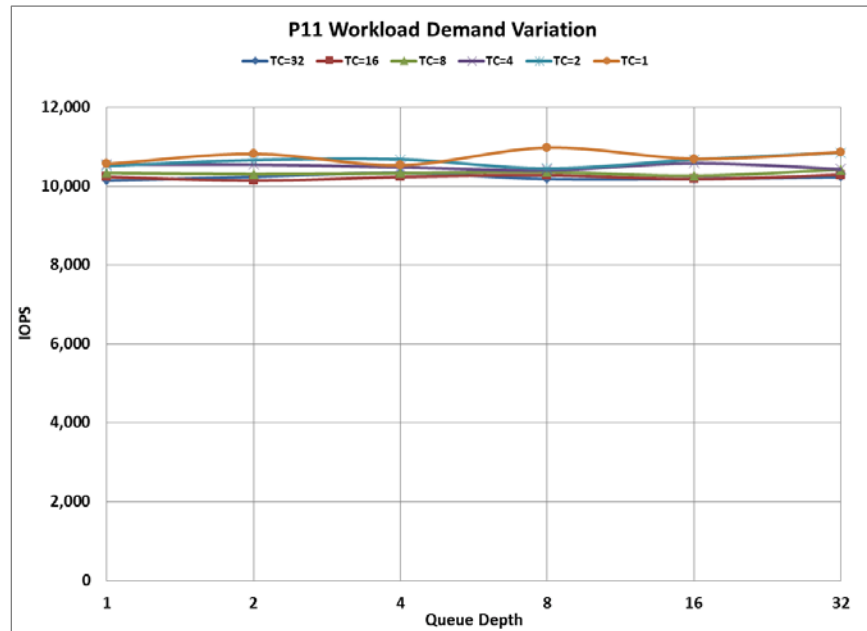
Figure 10-5. DIRTH: Demand Variation

10.3.3.6 Demand Intensity

The Test Operator shall report Demand Intensity by presenting a plot of ART v IOPS where each data series is a Thread Count and QD with IOPS along the x axis.
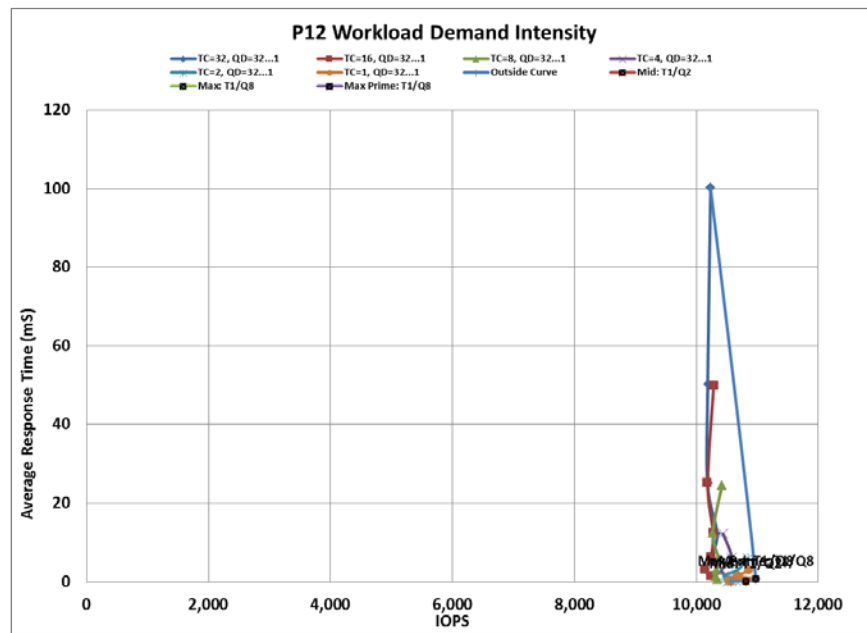

Figure 10-6. DIRTH: Demand Intensity

10.3.3.7 Response Time Histogram – MinIOPS Point

The Test Operator shall report the Response Time Histogram for the MinIOPS Point.
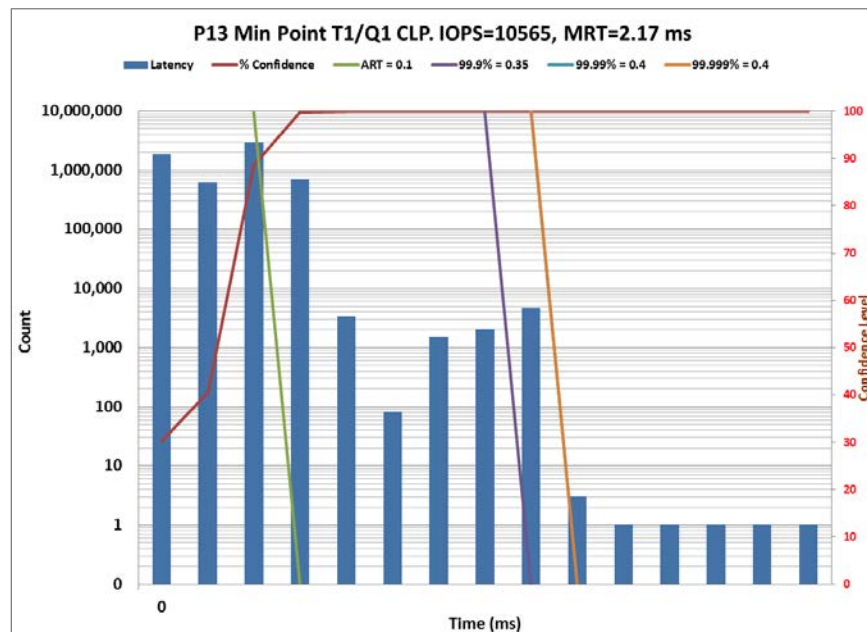
Figure 10-7. DIRTH: Response Time Histogram – MinIOPS Point

10.3.3.8 Response Time Histogram – MidIOPS Point

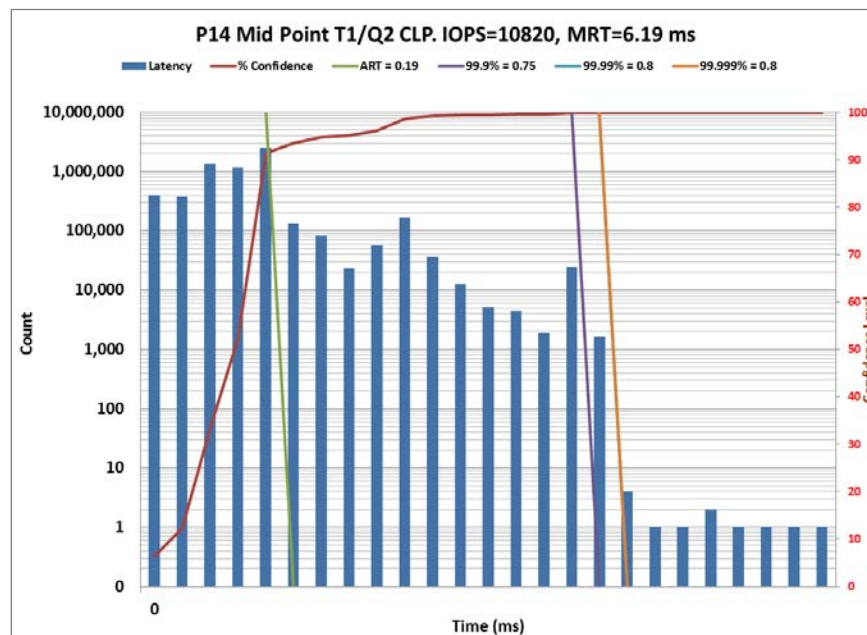The Test Operator shall report the Response Time Histogram for the MidIOPS Point.

Figure 10-8. DIRTH: Response Time Histogram – MidIOPS Point

10.3.3.9 Response Time Histogram – MaxIOPS Point

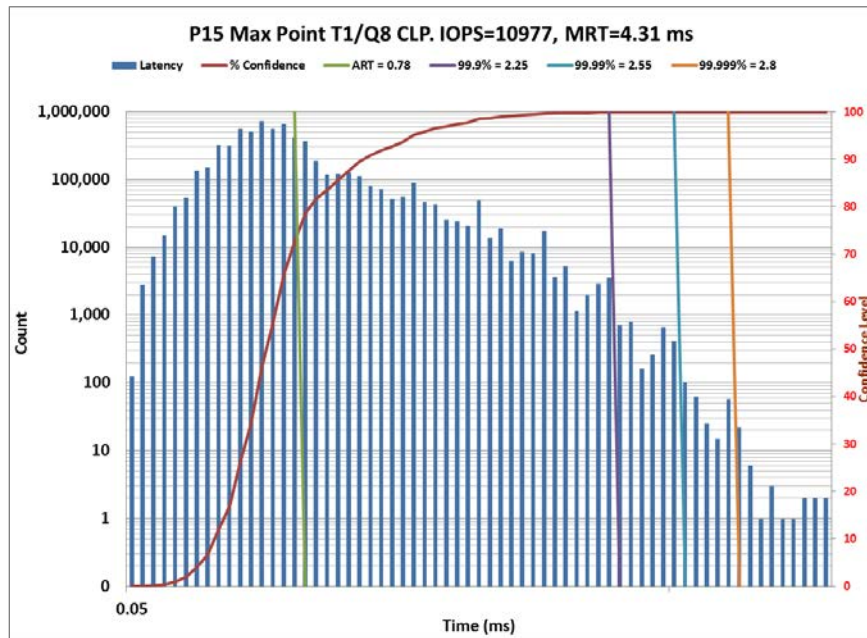The Test Operator shall report the Response Time Histogram for the MaxIOPS Point.

Figure 10-9. DIRTH: Response Time Histogram – MaxIOPS Point

10.3.3.10 Confidence Level Plot Compare – Min, Mid, Max & Max Prime

The Test Operator shall report the comparative Response Time Histograms for the MinIOPS, MidIOPS, MaxIOPS and MaxIOPS Prime points. Note that the data series for MaxIOPS Prime may be omitted when the TOIO values are the same as the MaxIOPS point.
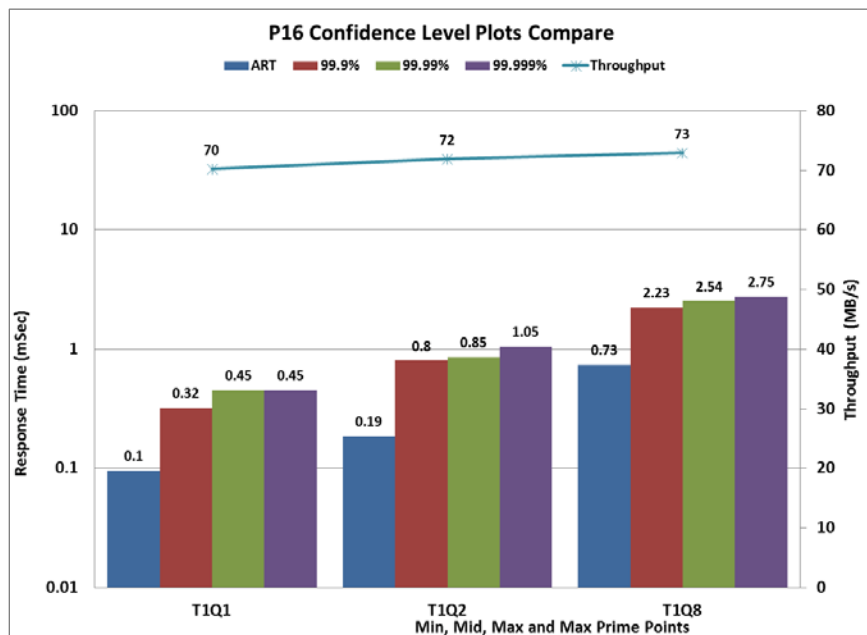

Figure 10-10. DIRTH: Confidence Level Plot Compare

10.3.3.11　ART, 5 9s RT & Bandwidth v Total OIO

The Test Operator shall report the ART, 5 9s Response Times and Bandwidth v Total OIO.
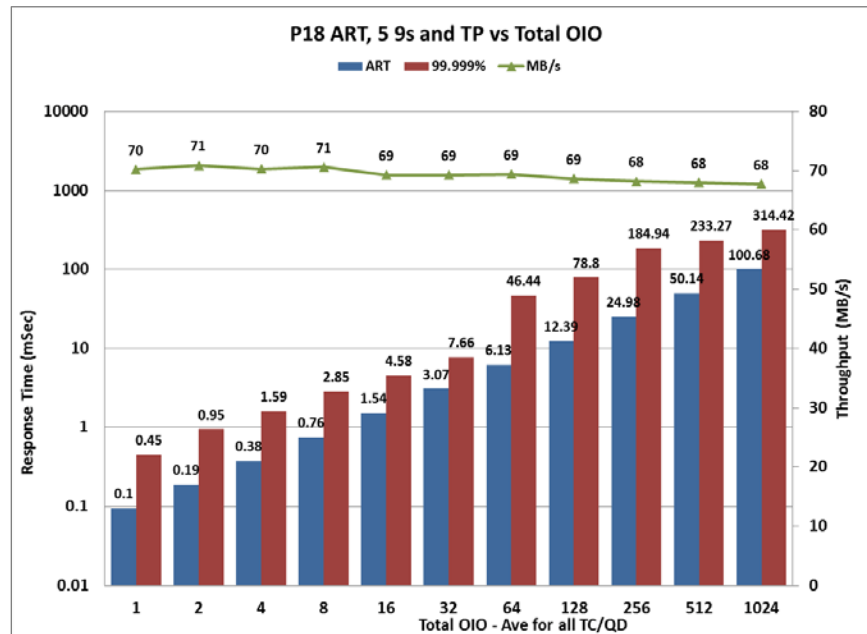

Figure 10-11. DIRTH: ART, 5 9s RT & Bandwidth v Total OIO

10.3.3.12　CPU System Usage % & IOPS v Total OIO

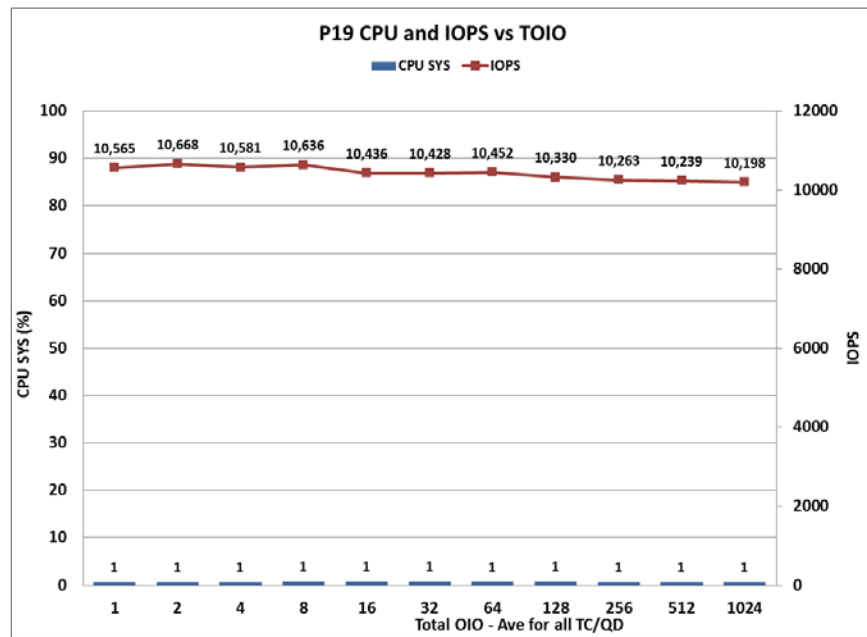The Test Operator shall report CPU System Usage % and IOPS v Total OIO.

Figure 10-12. DIRTH: CPU Sys Usage % & IOPS v Total OIO