Persistent Memory (PM) Storage Performance Test Specification (PTS)

Version 1.0

ABSTRACT: This specification describes best practices for Persistent Memory Storage Performance Test and sets forth a performance test methodology, PM storage platform set up, test settings, synthetic benchmark workloads, real-world application workloads and test results reporting format. It is intended to provide accurate, repeatable and reliable comparison of Block IO and In-Memory byte addressable test results used in traditional and PM aware applications under various PM Storage configurations.

SNIA Standard

June 17, 2023

USAGE

DISCLAIMER

The information contained in this publication is subject to change without notice. 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. 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
0.00	April 22-2019	Eden Kim	• Initial Draft Outline; Comment to Preamble by Eden Kim, Chuck Paridon, Tom West, Eduardo Berrocal
0.01	April 24, 2019	Eden Kim	• First Draft Content by Eden Kim
0.02	May 20, 2019	Eden Kim	• Solid State Storage (S3) TWG Concall Review • Section 1.1 Preamble Edit for External Release • Section 1.3 Background update with PMEM mode TC x QD=1
0.03	June 15, 2019	Eden Kim	• Update Background Section 1.3 • Insert Fig. 1-1 3 Access Mode Chart by Eduardo Berrocal • Update Fig. 1-2 PM Storage Demand Intensity Curves
0.04	June 18, 2020	Eden Kim	• Incorporate content from PM PTS White Paper v0.7.4 • Add new Figure 1-1 Storage Hierarchy from White Paper
0.05	August 15, 2020	Eden Kim	• Updated content from PM PTS WP • Added Background and Reporting Requirements Section
0.06	September 15, 2020	Eden Kim	• Updated Background section • Updated Test Sequences
0.07	October 1, 2020	Eden Kim	• Updated Data in Test Sections 6, 7, 8, 9 and 10
0.08	October 13, 2020	Eden Kim	• Incorporated Eduardo comments • Grammar and formatting in definitions section 2 • NB: Discussion on test settings p 19 for mmap, msync, non temp W page and definition/examples on p 32 section 3.3.2 and 3.3.3.1 IO Access modes that is incorporated in Test Sections 6,7,8,9,10 • Added Annex A: Sample PTS Report Header
0.08.2	Oct. 19, 2020	Eden Kim	• S3 TWG Concall review • Code set up for mmap, msync, non Temp W – Eden to Eduardo offline • IOPS variability – Eduardo to run pseudo code to validate • Defintions: interleaved from Keith; psync from Eduardo • Section 3.2.1 – background discussion NUMA from Keith • Section 3.2.1 – background discussion re: logical storage space, local or remote server, app direct v NUMA from Eduardo/Keith • STOP at Section 3.4
0.08.3	Oct. 26, 2020	Eden Kim	• Accepted changes up to Section 3.4 • definitions added: Psync, pread, pwrite, queue depth, test storage amount. • Definitions needed: interleaved • Added NUMA back into section 3.1.1 settings; added firmware version • Settings required - section numbers globally changed to bullets • Modified 3.2 to be “Test Storage Configuration” • Section 3.3 - Edited PM Settings to PM File Stack Configuration • Placeholder outline of PM File Stack Configuration – to be filled out • Section 3.3.2 – edited to memcopy, memcopy memsync, memcopy clwb, and non-temporal writes • Stopped at Section 3.4
0.08.4	Nov. 3, 2020	Eden Kim	• Group discussion on Eduardo text for Section 3.3.1 from v0.08.3 • Eduardo & Keith to re organize and draft 3.3.1 and sync with Fig. 1-3 • Eden to update Fig 1-3 Storage Access Modes per Keith/Eduardo

			• Eden & Eduardo to run new IO Engines to validate PM test sections
0.08.5	Nov. 9, 2020	Eden Kim	• Replaced Figure 1-3 – Simplified Storage Access Modes • Group Discussion & Edits for Section 1.3.2 Fig. 1-3 • Added definitions for: mode; raw access • Homework: Review Eduardo's re-draft of 3.3.1 with Keith's comments Harmonize 3.3.1 and Figure 3-1 – IO Access Modes with Section 1.3.2 Storage Access Mode introduction and 4 mode diagram
0.08.6	Nov. 16, 2020	Eden Kim	• Need Definition for 'Interleaved' section 2.1.22 • Updated Test Settings Section 1.4.2 item 8. In-Memory IO Access Engines: blockio sync, memcopy nosync, memcopy Msync, memcopy clwb and memcopy non temporal. • Inserted Section 1.4.2 item 7. Namespace: sector, fsdax, devdax, raw • Inserted Section 3.2 PM Interleaving • Inserted Fig. 3-1 Interleaved and Fig. 3-2 Non-Interleaved • Updated Section 3.3 Background • Inserted Updated Fig. 3-3 PM File Stack Configuration Modes • Next Section to discuss: 3.3.32Test Settings
0.08.7	Nov. 30, 2020	EK KO EB	• KO/EB – Section 3.3 edits: • Interleaved definition 2.1 • Added placeholder definitions for PM namespace-sector, PM namespace-fsdax, PM namespace-devdax, PM namespace-raw, IO Engine-blockio_sync; IO Engine-memcopy_nosync; IO Engine-memcopy_msync; IO Engine-memcopy_CLWB; IO Engine-memcopy_non-temporal-writes • Added 3.2.3 Remote v Local Server • Edited 3.3.2 PM File Stack Modes • Edited 3.3.3 PM Namespace Settings • Stopped before 3.3.4 IO Engines
0.08.8	Dec. 07, 2020	Eden Kim	• Definitions for Namespaces • Defintions for IO Engines • Definition for IO Stiumulus Generator • Reviewed Section 3.3.4 • Homework – review Test Methodologies Sections
0.09.1	Dec. 14, 2020	Eden Kim	• Added definitions for PM File Stack and PM File Stack Modes • Edited 1.3.2 Storage Access Modes to be "programmatic configuration" and 3.3.2 to be PM File Stack Configuration Modes to be "logical configuration" • Replaced memcopy_msync with memcopy_OSsync • Updated section 1.4.2 Test Settings • Updated Figure 1-7 PM Storage DI Curves: memcopy_CLWB, memcopy_OSsync, memcopy_Non-Temporal Writes • Homework – review 3.4 & 3.5: please edit and comment word doc
0.09.2	Dec. 18, 2020	Chuck Paridon	• Edits to Section 3.4-3.7
0.09.3	Dec. 21, 2020	Eden Kim	• Group review of Sections 3.4 – 3.7 • Added Annex B – PM RTP – Tom West to update and return • General Discussion re: write atomicity, efficacy of workload stimulus, vendor neutrality vis a vis Intel specific commands and availability of software tools, IO engines, etc. • Need to discuss with Eduardo: Test Validation with Calypso; Write Atomicity topic; Intel Specific PMEM v SNIA vendor neutrality • Next Action: Week of Jan. 4 – test validation • Next Concall: Jan. 11 – see v0.09.4 redline and clean to be posted
0.09.4	Jan. 11, 2020	Group	• Annex B – PM RTP – Eden & Keith to rev for multiple suppliers • Section 1.3.2 Eduardo to find terminology for "memcopy 8 byte atomicity" • Sections 1.3.7-8 Keith to reveiew for contextual clarity/consisitency with Section 1.3.2

0.09.5	Jan. 25, 2021	Eden Kim	• Annex B – Table 1 PM RTP Servers (EK) • Annex B – Table 2 Performance Related Software & Components • Section 1.3.2 – Failure Atomic Store Size (EB) • Section 1.3.7 – Rewording consistent with 1.3.2 (KO)
0.09.6	Feb. 08, 2021	Eden Kim	• 1.3.2, 1.3.7 & Annex B Table review • Remove Non-PM PTS related Definitions • Discussion re: Test Validation Calypso & Intel
0.09.6 clean	Apr. 24, 2021	Eden Kim	• Accepted all changes from v0.09.6 redline
0.09.7	Apr. 25, 2021	Edem Kim	• Update to Figures 1-3 Storage Access Modes • Replaced 3-1 PM File Stack Configurations with new 3-1 Storage • Access Modes – Software Perspective. • Added new Figure 1-7 PM Storage Access Modes – Data Path Perspective • Replaced 1-7 with new 1-8 DI Curves: IO Engine Modes • New Figure 3-3 PM Linux Stack Configurations • New Figure 3-4 PM IO Engine Modes • Placed “Descriptive & Tech Notes – Informative” after Section 1.3.1 • Place “Tech Note – Data Path” after Section 1.3.6 • Reduced PM Tests to four: TC/QD Sweep; Replay; Ind Streams & • Synthetic Corner Case. Updated first pass of pseudo code. Added • Global test settings and Global PC. • Updated TOC and Table of Figures with accepted changes
0.09.8	Apr. 26, 2021	Eden Kim	• Solid State Storage (SSS) TWG call review of 0.09.7 • Updated 1.3.2 Storage Access Modes & Fig 1-3 • Discussion & Edit 1.3.3. Block v Byte Access • Editing IO Engine Diagram ppt v 0.9.2
0.09.8.1	May 3, 2021	Eden Kim	• New section 1.3.3 Block IO v PM Memory Mapped (byte) Access • Modes: discussed Fig 1-3 columns 2,3,4 • New Section 1.3.4 Block v Byte Access – Hardware view: to be discussed next meeting with updated Fig 1-4 & 1-5
0.09.8.3	May 17, 2021	Eden Kim	• 1.3.3 New title: PM Block IO (sector) v PM Direct Access (fsdax & devdax) • 1.3.4 Title: Traditional Block v Byte Access: Hardware View
0.09.8.4	May 24, 2021	Eden Kim	• Section 1.3.4 Storage Access Modes HW: R Path/W Path
0.09.8.5	June 7, 2021	Eden Kim	• Section 1.3.3 – New first sentence describing PM PTS data paths • Section 1.3.4 – New Diagram Figure 1-4 & accompanying text
0.09.8.6	June 21, 2021	Eden Kim	• Section 1.3.3.1 – updated synchronous v asynchronous IO & TC • Section 1.3.3.1 – updated sector v base sector size atomicity • Section 1.3.4 – updated Figure 1-4 Storage Access Mode: HW View
0.09.8.7	June 28, 2021	Eden Kim	• Deleted old 1.3.5 1.3.6 on Block v Byte Access • Section 1.3.7 now 1.3.5 PM Access: Data Path Perspective • Updated 1.3.6 Block Access w/ sector atomicity • Update 1.3.7 Block Access w/o sector atomicity • Next Mtg: 1.3.8 Direct Access; Tech Note on Data Path

0.09.8.8	July 19, 2021	Eden Kim	• Normative & Other References – Section 1.8: updated • Section 1.3.5-7 deleted with Notes added by Note Number • Section 1 is now 1.3.2 Storage Access Modes; 1.3.3 PM Block IO v PM Direct Access; 1.3.4 Trad BlockIO v PM Access: Hardware View; • Notes are Note 1. Sector Atomicity; Note 2. PM Thread Count; Note 3. Data Path • Next Section 1.4 PM Storage and Tests specifications: 1.4.1 Best Practices; 1.4.2 Test Settings; 1.4.3 Test Methodologies; 1.4.4 Demand Intensity Curves
0.09.9.0 Clean	July 26, 2021	Eden Kim	• Update Background Sections 1.3 and 1.4 • Clean copy for group review
0.09.9.1	Sept. 20, 2021 Sept. 27, 2021	Eden Kim	• Note 5: Data Path edit • Section 1.4 – 1.8 reviewed (PM Storage Test Best Practices, Scope, Not in Scope, Disclaimer, Norm Ref)
0.09.9.2	Oct. 18, 2021	Eden Kim	• Start 2.0 Definitions & 3.0 Key PM Test Process Concepts & Reporting •
0.09.9.3	Oct. 25, 2021	Eden Kim	• Section 3.3
0.09.9.4	Dec. 13, 2021	Eden Kim	• Section 5 IO Capture Tools updated • Note 6: IO Capture Tool Metrics Algorithms - added
0.09.9.5	Jan. 30, 2022	Eden Kim	• Clean version from 0.09.9.4 • Update to Test Flow & Tests
0.09.9.6	Feb. 28, 2022	Eden Kim	• Eduardo comments to Section 1.4 • Tom West comments Section 1.4.1
0.09.9.7 clean	March 07, 2022	Eden Kim	• Clean Version • New text for review: Section 1.4 – 5.3 • Updated TOC, List of Figures • Added List of Notes
0.09.9.8	Mar 07, 2022	Eden Kim	• Group review of clean version

0.09.9.8.3	Mar. 21, 2022	Eden Kim	• Definitions in RED
			• Next Meeting:
			  ○ PM Best Practices paragraph in section 1.4.1 by group
			  ○ Annex B language; draft of CMSI PM RTP page by Eden

0.09.10.1	Aug. 15, 2022	Eden Kim	• Update Sections 3.4, 3.5 • Update Sections 6, 7, 8 • Added Notes 7, 8, 9 • Added Annex B: PM PTS Report Header
0.09.10.4	August 26, 2022	Eden Kim	• Sections 6,7.8 and Appendix • Final Edits before S3 vote for release to TC

Contributors

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

Company	Contributor
Calypso	Eden Kim
dcx	Chuck Paridon
HPE	Keith Orsak
Hyper I/O	Tom West
Intel	Eduardo Berrocal Andy Rudoff
Samsung	Bill Martin Mike Allison
SK Hynix	Santosh Kumar
theDecisionPlace	Jim Fister

Intended Audience

This document is intended for use by individuals and companies engaged in the development of this Specification and in validating the tests and procedures incorporated herein. After approvals and release to the public, this Specification is intended for use by individuals and companies engaged in the design, development, qualification, manufacture, test, acceptance and failure analysis of Persistent Memory devices and systems and sub systems incorporating PM storage as well as in the development, optimization and deployment of PM aware applications.

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 two 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.

Contents

List of Figures

List of Notes

1 Introduction

1.1 Preamble

This Persistent Memory (PM) Storage Performance Test Specification (PTS) is intended to define, create and validate best practices, test methodologies, settings, tests and reporting requirements for the comparative performance evaluation of byte addressable and block IO Persistent Memory Storage using synthetic benchmark and real-world application workloads.

In this PM PTS, the tests and methodologies defined are intended to apply generally to PM Storage devices with attention given to PM storage embodiments such as 3D Cross Point Persistent Memory Modules (PMEM) and DRAM NVDIMM Type N/P (NVDIMM-N/P) modules. It is anticipated that additional requirements for other types of PM Storage and interconnect – such as MRAM, ReRAM, Spin Torque, CXL and others – will be added at a future date.

For the purposes of this PTS, PM storage refers generally to storage that resides on the memory channel bus and shares the attributes of very fast response times (very low latency), data persistence (non-volatile), cache coherence and access as a data or compute storage area (block IO or byte addressable storage). While primarily intended to serve as in-memory load-store storage, this PM PTS will address PM storage access both via traditional block IO paths (to be suitable for PM Storage use in existing applications) as well as byte addressable memory channel lanes (for use in PM aware application designs).

PM Storage can be distinguished from traditional NAND Flash based Solid State Storage (SSS) – such as defined and specified in the SNIA SSS PTS – by the mode of information access (block IO or byte addressable), underlying storage medium, hardware/software platform and settings necessary for the operation of the specified PM Storage. More specifically, PM Storage does not typically share performance hysteresis (such as write history, Fresh-Out-of-the-Box (FOB) and transition to steady state) observed with NAND Flash based storage. The NAND Flash Storage requirements for pre-conditioning and the effects of write history are therefore not applicable to PM Storage performance (see exceptions for initial warm-up requirements for first time – or FOB - use of NVDIMMs and other PM Storage).

This PM PTS is a SNIA Technical Position and thus shares requirements to be vendor neutral and product agnostic. However, as PM Storage is a recently emerging class of storage, there may be few, or single, embodiments of specific PM Storage technologies and/or products. Accordingly, every effort is made to ensure that the identification, definition, treatment of and specification for such PM Storage products adhere to the fundamental principles of vendor neutrality and product agnosticism.

This PM PTS also sets forth a recommended PM Reference Test Platform (PM RTP) for use with PM Storage. The PM RTP is intended to normalize PM performance measurements and sets forth the minimum hardware and software requirements for the proper usage of the identified PM Storage and associated best practices and settings for conducting, obtaining and reporting PM Storage performance.

Readers and industry members are encouraged to participate in the further SNIA Solid State Storage (SSS) TWG works and can contact the TWG at its website portal at http://www.snia.org/feedback/.

1.2 Purpose

Manufacturers, developers and consumers of PM Storage need to be able to consistently and reliably test, evaluate and optimize the performance of Persistent Memory (PM) Storage under a range of platform set up criteria, test settings and workloads. This Specification defines a set of PM storage test best practices, settings and methodologies intended to enable comparative testing of PM Storage in Enterprise (see 02.1.18) and Client systems (see 02.1.9).

Note 1. Emphasis on PM Modules & NVDIMM-N/P. While the tests defined in this specification could be applied to PM Storage based on any technology (ReRAM, MRAM, Spin Torque, etc.), the emphasis in this specification is oriented towards Persistent Memory Modules based on 3D Cross Point technology and NVDIMM-N/P based on DRAM DIMM memory technology.

1.3 Background

1.3.1. Storage Hierarchy

Persistent Memory storage is broadly defined as high speed, low latency, non-volatile, cache coherent storage that sits directly on the memory bus (see Definitions 02.1.62). This PM storage tier resides below the faster and more expensive Main Memory tier and above the slower, but less expensive and higher capacity, NAND Flash SSD storage tier. See Figure 1-1 Storage Hierarchy and Figure 1-2 Storage Tiers below.

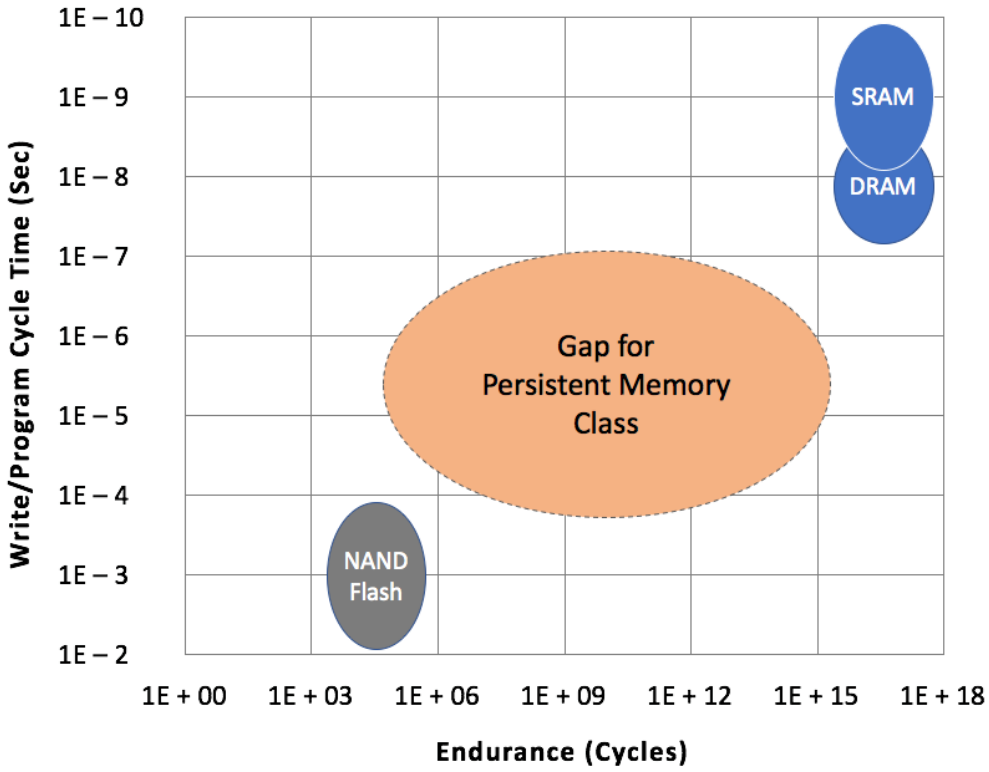


Figure 1-1 Storage Hierarchy

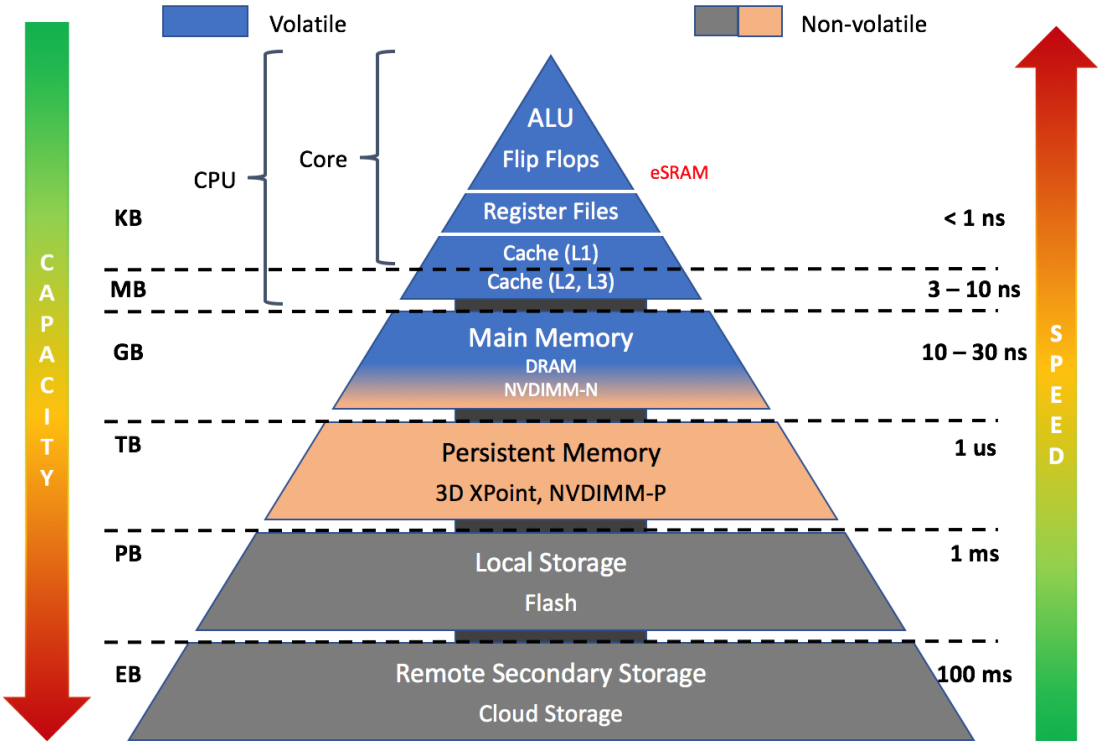


Figure 1-2. Storage Tiers

The recommended best practices and settings set forth in this PM PTS v1.0 are intended to apply, in general, to all PM Storage, but is targeted, in particular, at 3D Cross Point technology used in DCPMM and DRAM storage technology used in NVDIMM-N/P (and subsequent NVDIMM storage iterations).

Note 2. Descriptive & Technical Notes are Informative only. Notes are placed throughout the text to provide context and background, to explain the test rationale, and to provide examples of possible interpretations of test results. These notes are informative only and are set forth as “Descriptive Note” for the reader’s convenience.

1.3.2. Storage Access Modes

PM Storage can be addressed as both traditional block IO storage as well as direct access byte addressable storage (See Figure 1-3 Storage Access Modes below).

1. Traditional Block IO. Traditional Block IO access in column 1 below is shown only to reference legacy block IO storage and is not part of this PM PTS specification.
2. PM Block IO (**sector mode**). R/W IOs associated with existing applications can be applied to PM as a traditional block IO device by using PM Block IO sector mode in column 2. In this case, Reads and Writes are converted to cache line IO Load/Store operations by a PM Aware driver but appear as traditional block IO Reads and Writes to the application/user. The PM Aware driver ensures sector atomicity of a pre-determined sector size (Base Atomicity Store Size of 512 or 4K Bytes) by use of a Block Translation Table (BTT).
3. PM Direct Access (**fsdax mode**). Fsdax mode shown in column 3 supports dax (direct access) enabled file systems (such as XFS and EXT4 in Linux, or NTFS in Windows). These allow applications to directly access PM data through memory mapping. Because there is no BTT, there is a limited Base Atomicity Store Size (which is 8 Bytes in X86 architectures) in fsdax mode for Read/Write. See Note 3.
4. PM Direct Access (**devdax mode**). Devdax shown in column 4 is intended for situations where a file system (e.g., fsdax) cannot be used. While similar to fsdax, devdax mode does not support a file system nor Read/Write IOs. See Note 3.

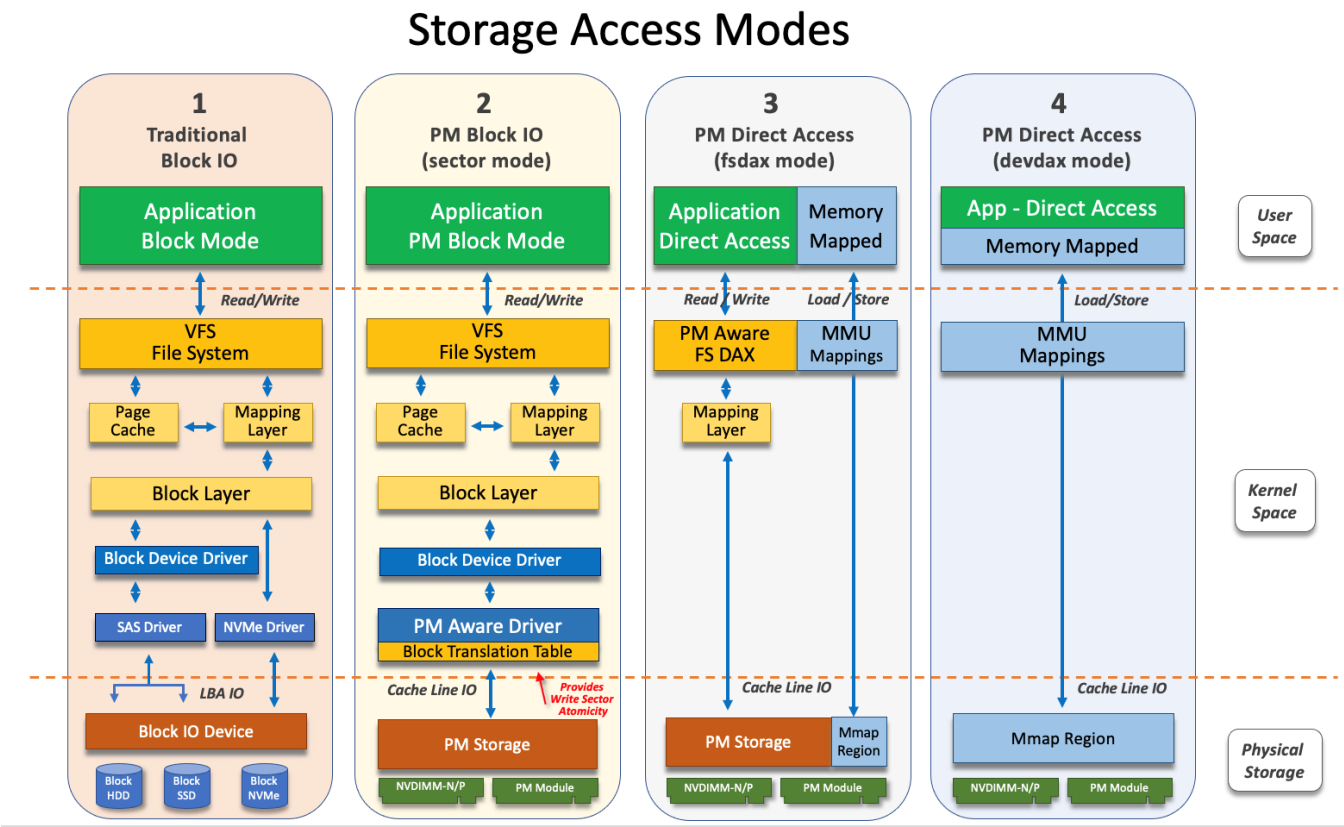


Figure 1-3. Storage Access Modes

Note 3. PM Direct Access and Data Integrity. PM Direct Access does not protect data from torn writes at the sector level in the event of a crash or power failure. Since all IO to PM is done

at cache line granularity (typically 64 Bytes), all IO operations are converted by the driver into memory copies. This means that an application's data can be corrupted due to x86 architectures that only guarantee a small Base Atomicity Store Size (such as 8 Bytes).

1.3.3. PM Block IO v PM Direct Access

Figure 1-3 illustrates the data paths discussed in this PM PTS specification: PM Block IO Access (Access Modes 2 & 3 – block IO data paths) and PM Direct Access (Access Modes 3 & 4 - memory mapped data paths). Note that PM storage performance can be significantly affected by workload Thread Count – see Note 4.

1.3.3.1. PM Block IO Access (sector & fsdax) Mode

PM Block IO (sector mode) and PM Direct Access (fsdax mode) both support legacy Read-Writes. In PM Block IO sector mode, legacy Read-Writes are converted into cache line Load/Store operations by the PM Aware driver. In Traditional Block IO access (Fig 1-3), IO requests to a storage device must be made using a device driver and an IO buffer. In PM Block IO (fsdax mode), Read-Writes are converted into cache line Load/Store operations directly by the file system.

1.3.3.2. PM Direct Access (fsdax & devdax) Mode

In PM Direct Access (through memory mapping), application Load/Store operations directly access PM by bypassing the file system, block layer and page cache which greatly reduces overhead (such as CPU context switching) and improves performance.

Note 4. PM Thread Count. Large Thread Count workloads can overwhelm system resources and result in anomalous performance. While legacy Block IO workloads may include high TC and high QD, PM Aware drivers will translate QDs into Threads. This can result in large Thread Count workloads, where the number of Threads exceeds the available physical or hyperthreaded cores in the system.

1.3.4. Traditional Block IO v PM Access – Hardware View

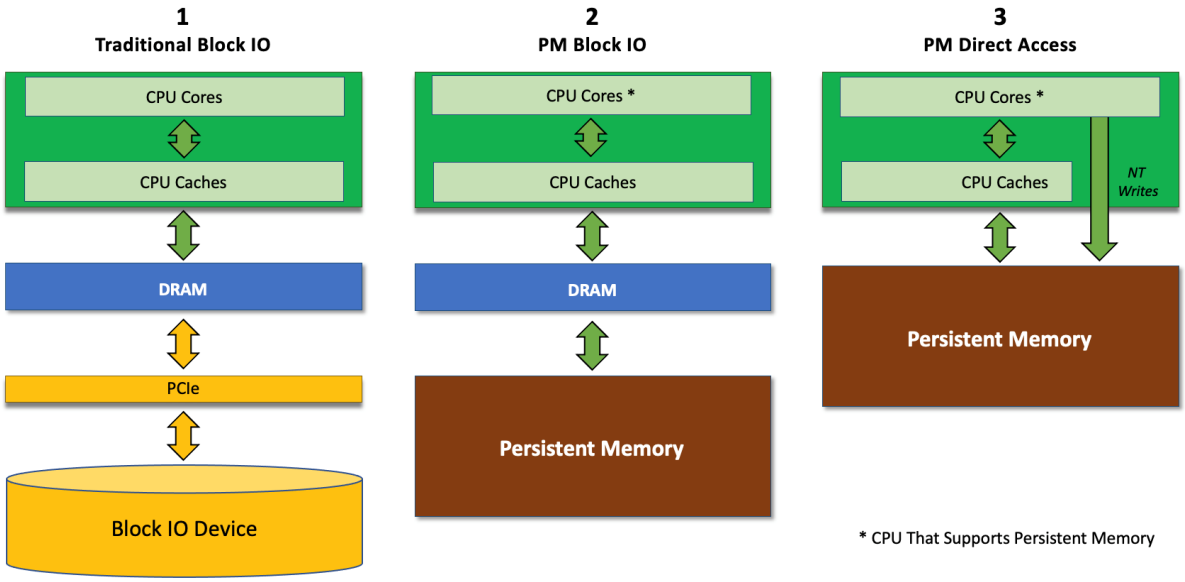


Figure 1-4. Storage Access Modes: Hardware View

In Column 1 Traditional Block IO, IOs are queued from the CPU to the device. The device executes IOs by performing Direct Memory Access (DMA) operations to and from DRAM across the PCIe or other storage bus.

In Column 2 PM Block IO, the PCIe bus is removed from the data path. The CPU executes IOs across the memory bus by executing memory copies from DRAM to PM and from PM to DRAM.

In Column 3 PM Direct Access, the CPU directly executes IOs using Load/Store operations (from CPU to PM and from PM to CPU). IO Writes (Stores) can bypass CPU caches when using Non-Temporal Writes. See Note 5.

Note 5. Data Path. The Data Path can impact performance depending on the use case. Smaller IO Writes (with temporal locality) can benefit from using CPU caches. Larger IO Writes (with low temporal locality) can benefit from bypassing the CPU caches and writing directly to PM storage. The following presents data paths and operations used between application space and PM storage devices in PM Direct Access modes**.**

1. All IO Activity (Reads and Writes) is initiated through an application running in User space
2. Write IOs (memory stores) are typically written to the associated CPU caches. Write IOs are then moved from the CPU cache via one of 2 methods:
    I. Kernel space: Msync system call
    II. User space: CLWB flushing instruction. Cache line write backs should be combined with an SFENCE instruction for synchronization.
3. Write IOs can also bypass CPU caches by issuing Non Temporal Write instructions. In this case an SFENCE (sync) instruction must be issued to ensure data persistence.
4. Read IOs are satisfied from the CPU caches if present and valid. Otherwise, they are moved to the CPU caches via a cache fault resulting in a Load operation from the PM device.
5. Note that in all instances the IO quantum to or from the PM device is a cache line.

1.4 PM Storage Test Practices

1.4.1. Best Practices

PM Storage has characteristics that differ from NAND-based Solid State Storage (SSS). Importantly, PM Storage does not display write hysteresis (write history) common to NAND Flash based SSS. Accordingly, PM based best practices do not specify a Device PURGE, pre-conditioning, steady state or block size sequencing. Instead, PM PTS tests recommend a short or optional warm-up and shorter duration IO measurements.

PM Best Practices include key considerations such as:

- PM Aware - ensure that the platform configuration and server set up, test settings, methodologies and workloads used are PM Aware.
- Thread Count and CPU Cores – All PM IOs generate synchronous memcopies and spawn an individual thread for each IO. Care should be taken to ensure that Total OIO does not exceed available CPU Cores and result in performance bottlenecks.

- QD & Sector Mode – In traditional Block IO (Figure 1-3 Sector Mode) IOs are changed into individual threads at the PM Aware driver level. Consequently, high QDs can result in high Thread Count which can saturate the available cores.
- PMEM configuration – Configure PM aware bios and storage modules appropriately (inter-leaving, regions, namespaces, etc.).
- Page Size – Configure page size amount. Page size boundaries must align to the namespace boundaries.
- Power – Performance Settings – CPU power state should be set to its highest value.

1.4.2. Test Settings

Measuring the performance of PM Storage is greatly influenced by platform settings, storage server set up, IO access and IO engine parameters, workload selection and test settings. Accordingly, much emphasis is placed on defining, setting and reporting the following:

1. **Platform Configuration**
    a. Motherboard
    b. CPU
    c. DRAM configuration
    d. Non-Uniform Memory Access (NUMA)
    e. ADR BIOS
    f. local or remote server
2. **Storage Server Set Up**
    a. OS & kernel version
    b. BIOS version
    c. PM firmware version
    d. Page Size Memory setting
    e. PM Test Module Configuration (number, type, capacity)
    f. Interleaved PM Region or non-interleaved PM Regions
    g. Type of PM Region used: PMEM or BLK
3. **IO Access & Engine Parameters**
    a. Namespace configured: Storage amount and type of namespace; type configures the IO Access Mode: (PM Block IO (sector), PM Direct Access (fsdax), PM Direct Access (devdax))
    b. File System Selection (XFS or EXT4 in Linux; NTFS in Windows)
    c. IO Engine (blockio_sync, memcopy_nosync, memcopy_OSsync, memcopy_CLWB and memcopy_Non-Temporal Writes)
4. **Workload Selection**
    a. Use Case
    b. Synthetic
    c. Real World Application
5. **Test Settings:**
    a. Workload Definition: IO Streams, Access Patterns, Thread Count (Demand Intensity) and Step Resolution
    b. Test Flow: warm up, workload IO Stream(s), Demand Intensity Sweep
    c. IO Steps: workload IO Streams(s), thread count, duration
    d. Reporting Requirements: metrics, set up, settings

1.4.3. Test Methodologies

The methodologies defined in this PM Performance Test Specification (PM PTS) attempt to create consistent test settings and test procedures such that performance tests conducted will be repeatable and consistent.

It is intended that the reader may make a fair and reasonable assessment of the resultant performance measurements and be able to use such test data for the optimization of PM aware applications, storage software and systems.

Please Section 3.5 Test Methodologies, Test Flow and Interpretations below.

1.4.4. Overview of Common Test Flow

The PM PTS tests (Individual Streams, Replay and Thread Count-Queue Depth Sweep) use the same general steps and flow, described in Figure 4-1. Test-specific parameter settings, reports, and other requirements are documented in the test sections themselves.

Begin “Basic Test Flow”

For (PM Total Storage Amount = the PM configuration and amount).

1) Set Global Platform and PM Server Settings

NUMA, Page Size Memory, Test Modules Configuration, Region configuration (Interleaving or Non-Interleaving, PMEM or BLK), Namespace Configuration (IO Access Mode), File System Configuration (FS creation and mount options)

2) Select IO Engine and IO Access Mode

a) PM Block IO (sector), PM Direct Access (fsdax), PM Direct Access (devdax).

b) blockio_sync, memcopy_nosync, memcopy_OSsync, memcopy_CLWB and memcopy_Non-Temporal Writes.

3) Select PTS Test & Workload

a) Individual Streams, Replay or Thread Count-Queue Depth Sweep test

b) IO Stream(s) – Individual, Sequence of Combinations, Fixed Composite.

4) Set Test Settings

a) Set Test Parameters (OIO in QD and/or Thread Count, warm-up, IO duration, etc.) as specified in the test script.

b) Run test flow and accumulate/record data as specified in the test.

5) Report Specified Data

a) The Test Operator shall report for each test:
   i) Platform and Server Set up
   ii) IO Engine, IO Access Mode, File System
   iii) Plots for warm up, test rounds, and Demand Intensity Curves

b) The Test Operator shall report specified data in SNIA Report Format:
   i) Report Headers shall contain information specified in Annex A)
   ii) Test Data shall be reported as specified in PM Test Sections herein.

End “For PM Total Storage Amount”

The Test Operator may re-run the entire “For PM Total Storage Amount” with alternate test parameters (or User Selected), which may be optional or required, depending on the test. Any User Selected test parameters or settings must be disclosed in all test data reporting.

End “Basic Test Flow”

Figure 1-5. Basic Test Flow

1.4.5. 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…
- …

1.5 Scope

1) Target Storage Server Set Up
2) PM Storage Configuration
3) Byte Addressable & Block IO Access Modes
4) TC/QD Sweep Test Methodology
5) Real World Workload Test Methodology
6) Synthetic Application Workloads
7) Real World Storage Workloads
8) PM Storage Tests
9) PM Performance Test Reporting
10) PM Performance Test Best Practices

1.6 Not in Scope

1) Certification/Validation procedures for this specification
2) Device reliability, availability, or data integrity

1.7 Disclaimer

Use of any third party or proprietary hardware or software does not imply nor infer SNIA or SSS TWG endorsement of the same. Reference to any such test or measurement software, stimulus tools, software programs or PM Storage products is strictly limited to the specific use and purpose as set forth in this Specification.

1.8 Normative References

1.8.1. Approved references

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

- SNIA SSS PTS – Solid State Storage Performance Test Specification version 2.0.2 (www.snia.org/pts)
- SNIA RWSW PTS – Real World Storage Workload Performance Test Specification for Datacenter Storage v1.0.7 (www.snia.org/rwsw).
- NVM Programming Model v1.2 June 19, 2017

1.8.2. References under development

- None in this version

1.8.3. Other Informative references

- www.TestMyWorkload.com - SNIA CMSI Real World Workload reference captures & free IO Capture tools
- CMSI Reference Real World Workloads – Reference Real World Workloads & Table
- www.iotta.snia.org - SNIA IOTTA IO Trace Repository for IO Trace Captures
- Introduction to Persistent Memory Performance Test Specification 2020
- Real World Workloads – A Primer - August 2022

2 Definitions, symbols, abbreviations, and conventions

2.1 Definitions

2.1.1 **Applied Test Workload:** When used with Real World Storage Workload testing applies to 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%. See RWSW PTS v1.0.7

2.1.2 **Base Atomicity Store Size:** the number of bytes guaranteed by a driver, such as a PM driver, to maintain atomicity.

2.1.3 **Block IO:** The level of abstraction in the host server used by logical and physical volumes responsible for storing or retrieving specified blocks of data. A block is a sequence of bits or bytes with a fixed length such as 512 bytes, 4K, 8K, 16K, 32K, etc. bytes. IOs are done in block granularity in traditional file systems. In DAX FS (see below), operations at the block level are split into multiple operations at cache line granularity (typically 64bytes per cache line).

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

2.1.5 **Byte Addressable:** The capability provided by a storage media to perform a data transfer by specifying a starting location with a byte address rather than a block address.

2.1.6 **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.

2.1.7 **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. See RWSW PTS v1.0.7

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

2.1.9 **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). See RWSW PTS v1.0.7

2.1.10 **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.

2.1.11 **CXL:** Compute Express Link™ (**CXL™**) is an industry-supported Cache-Coherent Interconnect for Processors, Memory Expansion and Accelerators

2.1.12 **DAX:** Direct Access Method, or DAX, is direct I/O specially optimized for RAM. DAX refers to the case where the media can be accessed directly from the CPU through loads and stores instructions without the need of any intermediate IO protocol (such as PCIe), or data caching in DRAM (such as generally in file systems). Media that supports DAX connect directly to the memory controller.

2.1.13 **DAX File System:** A file system supporting DAX allows applications to access directly from user space without the need of context switching to the OS to run the file system. This is done through the mechanism of memory mapped files. Once a file is memory mapped to its address space, an application then can access the data using loads and stores instructions, as well as persist stores using flushing instructions, direction without any file system involvement. It is also possible to use regular IO read()/write() POSIX calls to a DAX FS, but those calls are internally transformed to loads and stores via memcpy().

2.1.14 **Demand Intensity:** Demand Intensity equals the total number of Outstanding IO (OIO).

2.1.15 **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.

2.1.16 **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).

2.1.17 **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

2.1.18 **Flush:** synchronization of data from volatile to non-volatile memory whereby writes still pending in the caches (CPU or Memory caches) are written through to media.

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

2.1.20 **Fsync:** Fsync is a Linux system call. Fsync synchronizes file content between cache and storage devices.

2.1.21 **Interleaved:** as in “interleaved modules” refers to configuring PM modules as an aggregated logical space where data is striped across memory channels within a single CPU socket. This memory space is presented to the OS as a single region (which can contain multiple namespaces therein). Note that interleaved PM modules cannot be interleaved across CPU sockets.

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

2.1.23 **IO Access Mode:** Refers to the type of IO access that PM Storage may utilize to read or write or to load-store data in a PM Storage device. Examples of IO Access Modes include memcopy_OSsync, memcopy_CLWB, memcopy_Non-Temporal W which can be used in conjunction with a direct access file system (such as DAX FS) for byte addressable load-store directly to the PM Storage device/media.

2.1.24 **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. See RWSW PTS v1.0.7

2.1.25 **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. See RWSW PTS v1.0.7

2.1.26 **IO Stimulus Generator:** IO Stimulus Generators, aka workload generators, create data access patterns that have specific characteristics such as random or non-random data patterns, random or sequential access, asynchronous or synchronous, read/write, data transfer sizes, etc.

2.1.27 **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 of a given Demand Intensity (or Queue Depth). 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. See RWSW PTS v1.0.7

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

2.1.29 **Memcopy:** a function from the standard libc to do general copy from memory addresses.

2.1.30 **Mode:** As in "sector mode," mode refers to an operational model for a given paradigm. Note that the use of sector in this context should not be confused with sectors as used in hard disk drive storage where sector is defined in the SNIA Dictionary.

2.1.31 **MRAM:** A type of PM Storage which stores data in magnetic domains.

2.1.32 **Msync:** an IO Access Mode whereby changes made to the in-core copy of a file that was mapped to memory using mmap(2) are flushed back to disk (committed to media).

2.1.33 **Namespace –** A Namespace is an amount of PM of a given Region that is made available to the Operating System. Note that Namespaces must be configured using a selected access mode (e.g., sector, fsdax, devdax, etc.) See Figure 1-3.

2.1.34 **Non-Temporal Writes:** an IO Access Mode whereby stores from registers to memory are made without interfering or influencing memory cache. i.e., where stores by-pass any intermediate caches and go directly to media.

2.1.35 **NVDIMM:** A computer RAM DIMM that retains data even when electrical power is removed.

2.1.36 **NVDIMM-N/P:** NVDIMM based persistent memory storage types defined by [JEDEC?]

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

2.1.38 **Outstanding IO (OIO):** The number of IO operations issued by a host, or hosts, awaiting completion. Total OIO is the product of Thread Count multiplied by Queue Depth.

2.1.39 **OSsync:** an IO Access Mode whereby changes made to the in-core copy of a file that was mapped to memory using mmap(2) are flushed back to disk (committed to media).

2.1.40 **Memory Page:** Fixed-length contiguous block of virtual memory corresponding to the smallest unit of data for the memory management of a memory device in a virtual memory Operating System. The virtual address of a memory page is translated to the corresponding physical address by the use of a page table.

2.1.41 **Page Size:** The size of the memory pages the Operating System will use for a particular memory device.

2.1.42 **Persistent Memory:** A method or apparatus for efficiently storing data structures that has the characteristics of being directly connected to the memory bus, has data persistence and has very low response time behavior (or latency).

2.1.43 **Persistent Memory Storage:** A storage device, sub system or system that utilizes Persistent Memory as traditional storage to access, store and retrieve files or data objects. PM Storage typically may operate in traditional Block IO Read-Write Access Mode or Byte Addressable Load-Store IO Access Mode.

2.1.44 **PM IO Engine Mode.** PM IO Engine Mode refers to the manner in which the IO stimulus is applied to the target PM storage. This can be done in various ways including blockio_sync, memcopy_OSSync, memcopy_CLWB and memcopy_Non Temporal Writes.

2.1.45 **Pread:** A POSIX system call to perform a Read as a synchronous operation.

2.1.46 **Process ID (PID):** A PID represents the unique execution of a program(s). See RWSW PTS v1.0.7.

2.1.47 **Psync:** An IO engine that performs pread and pwrite system calls.

2.1.48 **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. See SSS PTS v2.0.2

2.1.49 **Pwrite:** A POSIX system call to perform a Write as a synchronous operation.

2.1.50 **Queue Depth:** The number of IOs in a given queue.

2.1.51 **Raw Access:** In relation to storage device, IO requests are submitted directly to the storage device without use of a file system.

2.1.52 **ReRAM:** A type of PM Storage or device whereby data access and recording is accomplished by changing resistance across a di-electric solid state material, often referred to as a memristor.

2.1.53 **Real-World Workload:** IOs, IO Streams or workloads derived from or captured on a deployed server during actual use. See RWSW PTS v1.0.7

2.1.54 **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. See RWSW PTS v1.0.7.

2.1.55 **Replay Test:** A test that reproduces the sequence and combination of IO Streams and QDs for each step of the IO Capture. See RWSW PTS v1.0.7.

2.1.56 **Region:** A Region is a grouping of one or more NVDIMMs related to a specific CPU socket. The Region can be interleaved or non-interleaved.

2.1.57 **Region – Interleaved:** An Interleaved Region spans all of the NVDIMMs associated with a given CPU socket**.**

2.1.58 **Region – Non-interleaved:** Non-interleaved Regions expose individual NVDIMMs as available storage for a given CPU socket.

2.1.59 **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). See RWSW PTS v1.0.7

2.1.60 **Sfence:** x86 Sfence (Store Fence) instruction. SFENCE performs a serializing operation on all store-to-memory instructions that were issued prior the SFENCE instruction. This serializing operation guarantees that every store instruction that precedes in program order the SFENCE instruction is globally visible before any store instruction that follows the SFENCE instruction is globally visible. Non-temporal Stores to PM, as well as flushed CPU cache lines to PM, made globally visible by an SFENCE instruction are also guaranteed to be persisted. (i.e., not in transit buffers or in CPU caches).

2.1.61 **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.

2.1.62 **Spin Torque:** A type of PM Storage or Device that accomplishes data access and storage by running a current through a thick magnetic layer (usually called the “fixed layer”) to create a spin-polarized current which can then be directed into a second, thinner magnetic layer (the “free layer”) whereby angular momentum can be transferred to this layer to change magnetic orientation.

2.1.63 **Target Server:** The host server from which an IO Capture is taken.

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

2.1.65 **Test Storage Amount:** the amount of PM storage capacity accessible for test

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

2.1.70 **Write Back:** IO stack architecture that allows data stored in a cache to be written to back-up storage asynchronously.

2.1.71 **Write Through:** IO stack architecture that allows data stored in a cache to be written to back-up storage synchronously.

2.2 Acronyms and Abbreviations

2.2.1 **ART:** Average Response Time

2.2.2 **CLWB: Cache Line Write Back**

2.2.3 **CMSI:** Compute, Memory & Storage Initiative of the SNIA

2.2.4 **CXL:** Compute Express Link™ (**CXL™**) is an industry-supported Cache-Coherent Interconnect for Processors, Memory Expansion and Accelerators.

2.2.5 **DAX:** Direct Access as in Direct Access File System

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

2.2.7 **DIRTH:** Demand Intensity / Response Time Histogram (aka Thread Count/Queue Depth sweep test)

2.2.8 **DUT:** Device Under Test

2.2.9 **FOB:** Fresh Out of Box

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

2.2.11 **LAT:** Latency

2.2.12 **LBA:** Logical Block Address

2.2.13 **OIO:** Outstanding IO

2.2.14 **PM:** Persistent Memory

2.2.15 **QD:** Queue Depth

2.2.16 **RND:** Random

2.2.17 **R/W:** Read/Write

2.2.18 **RWSW:** Real World Storage Workload

2.2.19 **SEQ:** Sequential

2.2.20 **SSD:** Solid State Drive

2.2.21 **SSS:** Solid State Storage

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

2.2.23 **SW Stack:** Software Stack

2.2.24 **TC:** Thread Count

2.2.25 **TC/QD Sweep:** Thread Count / Queue Depth Sweep, aka DIRTH test

2.2.26 **TOIO:** Total Outstanding IO

2.2.27 **TP:** Throughput

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

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

3 Key PM Test Process Concepts & Reporting

The performance of PM Storage is highly dependent on the PM Storage Server Set Up, Storage Configuration, PM File Configuration Mode, PM Workloads, PM Test Methodologies and PM Test Flow.

3.1 PM Storage Server Set Up

3.1.1. PM Storage Server

The performance test of Persistent Memory (PM) Storage requires use of a PM aware/capable storage server. Test software may be loaded and run directly on the target PM Storage server or otherwise connected by direct or remote means. See Appendix B: Persistent Memory Reference Test Platforms.

NUMA awareness is important because the NUMA setting can significantly bottleneck performance of PM. A complete discussion of NUMA topology and implementation is outside the scope of this document.

3.1.2. PM Server Reporting Requirements

The following items shall be reported in the PM Test Report Header or in any PM PTS performance report:

- Motherboard
- CPU
- NUMA settings
- DRAM configuration
- Operating System & kernel version
- BIOS version
- PM Firmware version
- Page Size Memory

3.2 PM Module Test Storage Configuration

3.2.1. PM Modules – Configuration, Namespaces & PM Storage Regions

PM Modules must be configured prior to test. This includes the number, type and capacity of PM Modules, the number of PM Modules per CPU socket, namespaces and storage regions. The namespace configures the access mode for a defined amount of PM created from a PM Storage Region. A system can have multiple namespaces/storage regions configured simultaneously.

3.2.2. PM Module – Interleaving

PM Modules can be configured as “interleaved” or “non-interleaved”.

In the case of interleaved, the memory controller stripes the data across all the modules within a single CPU socket presenting a unified whole (Figure 3-1). Note that interleaving across sockets is not allowed.

In the case of non-interleaved, Figure 3-2, a single Storage Region is created for each PM Module. Non-interleaving can be useful for software RAID configurations.

The selected interleaving configuration shall be disclosed.

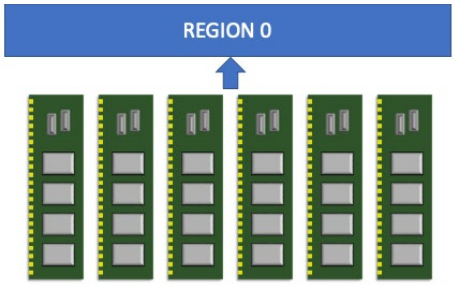


Figure 3-1 Interleaved PM Modules

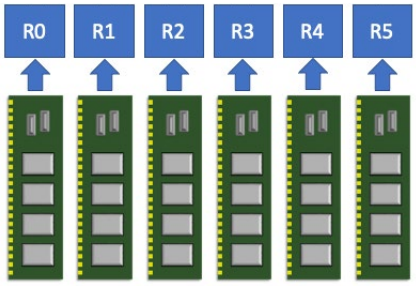


Figure 3-2 Non-Interleaved PM Modules

3.2.3. PM Module Test Storage Configuration - Reporting Requirements

The following items shall be reported in the PM Test Report Header or in any PM PTS performance report:

- Number & type of PM Modules
- Capacity per Module
- Interleaved or Non-Interleaved
- Test Storage Amount – amount of testable PM storage
- Storage Regions – number & capacity of storage regions
- NUMA aware – enabled / disabled

3.3 PM Settings: Namespace, File System & IO Access Mode

3.3.1. PM Namespace

After configuring the PM Storage Regions (interleaved or non-interleaved) the test operator shall configure the PM Namespace and storage regions. The PM Namespace configuration identifies both the IO Access Mode (sector, fsdax, devdax, or raw) and the Test Storage Amount. The PM Namespace and REGIONS shall be disclosed.

3.3.2. File System

Different File Systems can have different performance characteristics. The File System that is used shall be disclosed.

3.3.3. PM IO Access Modes

PM IO Access Modes refer to the manner in which IO Load-Stores are directed to target PM storage. This can include the use of flushes, caches and direct access to

media. The PM IO Access Mode selected shall be disclosed. Please refer to Figure 1-3. Examples of IO Access Modes include:

- **blockio_sync.** Synchronous Read/Write using systems calls (such as pread() and pwrite() in Linux) that include the execution of flush (using calls such as fsync() in Linux) after every write. This mode can be used with any Namespace type except devdax.
- **memcopy_nosync.** Memory copies (using memory mapped files) without flushing. To be used with Namespace Mode-fsdax and Namespace Mode-devdax.
- **memcopy_OSsync.** Memory copies (using memory mapped files) with flushing from the OS (using system calls such as msync() in Linux). To be used with Namespace Mode-fsdax and Namespace Mode-devdax.
- **memcopy_CLWB.** Memory copies (using memory mapped files) with flushing from user space with the CLWB (cache line write back) instruction. This mode requires the execution of a barrier instruction (such as SFENCE in x86) after every write to ensure that all previous issued flushing finished. To be used with Namespace Mode-fsdax and Namespace Mode-devdax.
- **memcopy_Non-Temporal Writes.** Memory copies (using memory mapped files) using non-temporal store instructions. These memory copies bypass the CPU caches but still require the execution of a barrier instruction (such as SFENCE in x86) after every write to ensure data persistence. To be used with Namespace Mode-fsdax and Namespace Mode-devdax.

Note 6. IO Access Modes. Memcopy access modes (with the exception of non-temporal writes) perform Reads & Writes using standard memcopy library functions (such as memcpy() in C/C++). Different modes of Writes use different synchronization methods. Non-Temporal Writes require the use of special instructions. Open source libraries, such as Persistent Memory Developer Kit (PMDK), provide high level APIs to implement non-temporal writes.

3.3.4. PM Settings – Reporting Requirements

The following items shall be reported in the PM Test Report Header or in any PM PTS performance report:

- PM Namespace
- File System Used
- PM IO Access Mode Selected

3.4 PM Workloads – Synthetic & Real World Workloads

3.4.1. Background

Workload content is a key PM Storage performance determinant. This PM PTS sets forth both Synthetic and Real World Storage workloads. These workloads are described as one or more IO Streams applied to the target storage. IO Streams are defined as random or sequential accesses of Read or Write IOs with a specified data transfer size. This allows the test operator to compare storage products across different manufacturers or by different workload types (Synthetic or Real World).

3.4.2. Synthetic Workloads

Synthetic workload tests are designed to evaluate storage performance across a range of criteria. These Performance Test Specifications focus on reporting IOPS, Bandwidth and Response Time performance using different workloads and test settings.

3.4.3. Real World Workloads

Real World Application and Storage tests apply combinations and sequences of IO Streams and Demand Intensity (or Thread Count) as observed in Real World Workload IO Captures.

Real World Workloads can be used in different tests (see 3.5 below) including:

1. Individual Streams tests – application of a single IO Stream across a range of Thread Count
2. Composite Streams tests – application of a composite IO Stream across a range of Thread Count
3. Replay-Individual Streams tests – application of individual IO Streams followed by a Replay of the sequence of IO Stream combinations observed in a Real World Workload IO Capture

3.4.4. Creating Real World Workloads based on IO Captures

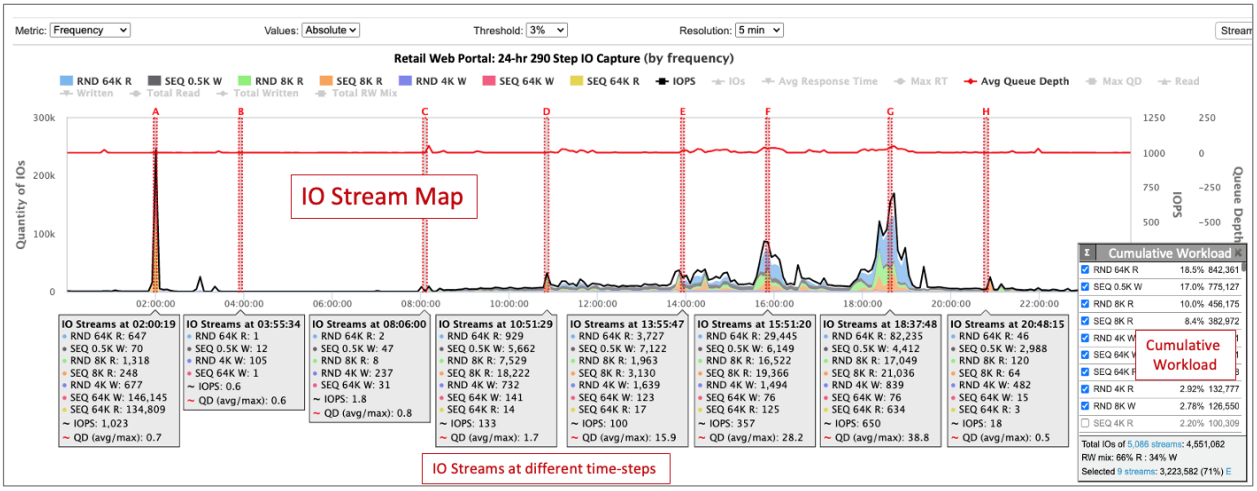
Real World Workloads are constructed from IO Captures observed during Real World Application and Storage use. IO Captures tabulate statistics on IO Streams observed during real world use.

IO Captures differ from IO Trace files. IO Captures are a series of time-steps across which IO Streams and metrics are aggregated and contain no actual data content.

IO Captures can be taken using IO Capture tools or can be created by converting IO Trace files into IO Capture files. See “Real World Workloads – A Primer” white paper on the CMSI website for more details and examples.

Once a Real World Workload is captured, a subset of the total IO Streams observed over the IO Capture are selected. Each step of the IO Capture is then converted into a test step. See Note 7.

Note 7. IO Capture Time-Steps & Replay. An IO Capture file consists of a series of time-steps that correspond to the observation of IO Streams over the course of an IO Capture. An IO Stream Map (below) shows the time-steps (x-axis) of a Real World Workload IO Capture.



Each time-step of an IO Capture aggregates the IO statistics observed over that time-step. In the 24-hr 290 time-step capture above, each time-step aggregates IOs over a 5 min duration.

By default, the IO Stream Map displays the 9 most frequently occurring IO streams over the course of the IO Capture. These IO Streams are displayed in the IO Stream Map Cumulative Workload box (see above). Cumulative Workload IO Streams are used to create composite stream workloads (see Note 8) as well as to create Replay test steps (see Note 9).

Each of the time-step points highlighted in the IO Stream Map above show a different number and combination of IO Streams (which are a subset of the selected 9 IO Stream workload).

Note that each time-step will have some number of IO Streams up to or equal to the 9 Cumulative Workload 9 IO Streams.

3.4.5. Workload Reporting Requirements

The following items shall be reported in the PM Test Report Header or in any PM PTS performance report:

- Workload Type
  - Description (source IO Capture application type)
  - Synthetic or Real World
- IO Streams
  - Number: Number of different IO Streams
  - Synthetic Stream(s): Access Pattern (RND/SEQ, RW, Transfer size)
  - Real World Streams: Access Patterns and % of total IO Streams
- IO Capture
  - Level – File system, Block level, other
  - IO Capture Source – Software tool
  - IO Capture Step Resolution – Number of Steps, Time of each step
  - Number of storage devices – logical storage drives captured

3.5 PM Test Methodologies, Test Flow and Interpretation

PM Performance test methodologies presented in this PM PTS are based primarily on tests defined in the Solid State Storage PTS and RWSW PTS for Datacenter Storage.

3.5.1. Individual Streams Test

The Individual Streams test runs one or more individual IO Streams. These individual IO Streams may be selected from IO Streams observations in an IO Capture, selected by the test operator, or selected from the table of recommended Individual IO Streams.

Each IO Stream test workload is run individually. After a warm-up, the test workload is applied for a specified duration and Thread Count which the Thread Count is increased for successive IO durations.

Each individual IO Stream shall be reported as prescribed with presentation of IOPS, Bandwidth and response times and a Demand Intensity Curve among other items.

The purpose of the Individual Streams test is to observe the performance of each individual IO Stream, to compare individual IO Stream performance to manufacturer IO Stream specifications, and to assess the performance of each IO Stream against an increasing range of Thread Count.

3.5.2. Composite Streams Test

The Composite Streams Test applies a subset of the IO Streams observed in a Real World Workload. The Cumulative Workload shows the IO Streams observed over the duration of a Real World Workload and lists the IO Streams in descending order based on most frequently occurring IO Streams. By default the Composite Workload is comprised of the 9 most frequently occuring IO Streams observed in the Real World Workload. See Note 8.

Results of the Composite Streams Test include reporting IOPS, Bandwidth and response times and a Demand Intensity Curve among other items.

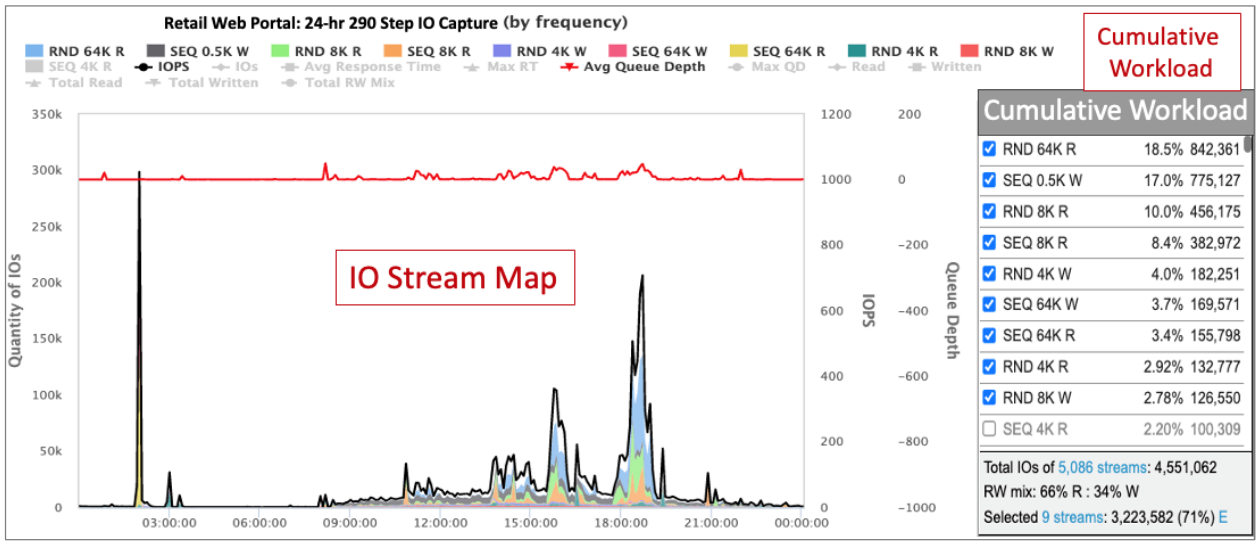
The purpose of the Composite Streams Test is to measure the performance of a composite multi-IO Stream workload. Composite IO Streams are typically taken from a reference Real World Workload Cumulative Workload such as those listed on the SNIA CMSI Reference Real World Workloads library, the SNIA IOTTA Repository or on the www.testmyworkload.com website.

3.5.3. Replay-Individual Streams Test

The Replay-Individual Streams test has two segments: Replay segment and Individual Streams segment. The Individual Streams segment measures the performance of each of the individual IO Streams observed in the IO Capture Cumulative Workload. The Replay segment applies the sequence of IO Stream combinations and Thread Counts observed during a Real World Workload IO Capture. See Note 9.

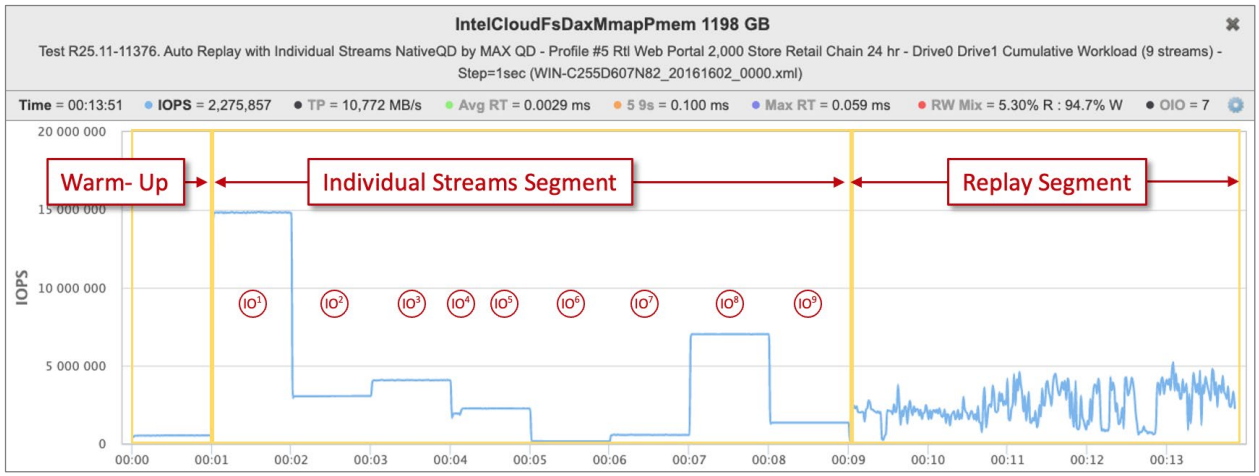
The purpose of the Replay-Individual Streams test is to observe the performance of the captured sequence of IO Stream combinations observed in an IO Capture compared to the performance of each individual IO Stream that comprises the workload.

Note 8. Cumulative Workload. A Real World Workload IO Capture shows an IO Stream Map (IO Streams vs Time) and a Cumulative Workload (Total IO Streams over the IO Capture). The Cumulative Workload is used to select the IO Streams used in the Composite Streams Test. By default, the 9 most frequently occurring IO Streams are used in the Composite Streams Test.



An IO Stream Map x-axis shows each of the time-steps of the IO Capture. While the Composite Streams test uses all 9 IO Streams, each step in a Replay Test will contain some number (up to 9) of the 9 IO Streams identified in the Cumulative Workload.

Note 9. Replay-Individual Streams Test. This test begins with a warm-up followed by an Individual Streams segment and a Replay segment. In the Individual Streams segment, the Cumulative Workload is used to select the 9 IO Streams to be individually tested.



In the Replay Segment, the sequence of IO Stream combinations observed in the IO Capture IO Stream Map are applied for the Replay Test segment.

The average performance for the Replay segment is reported in IOPS, MB/s and Response Time. Replay results can then be compared to the performance of each individual IO Stream tested in the Individual Streams segment.

4 Common Reporting Requirements

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

4.1 General

4.1.1. Test Date
4.1.2. Report Date
4.1.3. Test Operator name
4.1.4. Auditor name, if applicable
4.1.5. Test Specification Version

4.2 Test System Platform

4.2.1. Manufacturer/Model #
4.2.2. Mother Board/Model #
4.2.3. CPU
4.2.4. DRAM
4.2.5. BIOS firmware version
4.2.6. PM Driver version
4.2.7. Page Size Memory
4.2.8. NUMA setting
4.2.9. Test Storage Amount
4.2.10. File System or Block IO
4.2.11. Storage Access Modes: PM Block IO, PM Direct Access fsdax/devdax
4.2.12. PM Namespaces: sector, fsdax, devdax, raw
4.2.13. Key Test Settings: [insert from Test Sections]

4.3 Test System Software

4.3.1. Operating System & kernel Version
4.3.2. File System and Version
4.3.3. Test Software
4.3.4. IO Access Mode - blockio_sync, memcopy_nosync, memcopy_OSsync, memcopy_CLWB, memcopy_Non-Temporal Write

4.4 Device Under Test

4.4.1. Manufacturer
4.4.2. Model Number
4.4.3. Serial Number
4.4.4. Firmware Revision
4.4.5. PM Capacity per module
4.4.6. PM module count
4.4.7. Interleaved/non-interleaved

4.5 Workload

4.5.1. Synthetic or Real-World
4.5.2. Workload: Duration, steps, OIO range
4.5.3. Source Capture: storage configuration, file system or block IO, other
4.5.4. Test Type: Replay, TC/QD Sweep composite, individual streams
4.5.5. Pre-conditioning, Steady State & Demand Intensity

5 Software Tools & Reporting Requirements

This Specification is software tool and hardware agnostic. This PM PTS requires the use of both synthetic and real-world workload software tools. Any IO Capture tool or software tool that meets the requirements of the Specification may be used to capture real world workloads such as the tools available for free at www.TestMyWorkload.com or popular freeware such as blktrace, dtrace and others. IO stimulus tools must meet the requirements below and may be pubic, such as fio and vdbench, or private, such as Calypso CTS, software tools.

5.1 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, the IO metrics that are catalogued and the manner in which they are calculated. It is important to understand and disclose the level in the SW Stack where the IO Capture is taken and the methodology used to characterize 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.

5.2 IO Capture Steps & Conversion of IO Traces

It is important to capture real-world workloads in IO Capture steps of a given time period or duration to apply the tests set forth in this PM PTS. IO Capture tools collect the observed IO Stream and metrics over the specified time step. The given IO Capture tool will determine the main IO Stream characteristics such as Random or Sequential access, R or W IO and data transfer (or block) size. The selected IO Capture tool should be disclosed in the PTS report. Further, the IO Capture tool shall, upon request, disclose the relevant algorithm and criteria for determining such IO Stream characteristics. Please see Note 6.

By capturing IO Capture steps, IO Captures are able to record longer duration captures without the corresponding very large data sets associated with IO Traces. IO Capture steps allow the user to more easily characterize longer term workloads by identifying the main IO Stream components and saving those IO Streams and metrics in a table of statistics as opposed to saving the IO Stream block sizes themselves. Finally, IO Capture steps can be replayed as a stimulus in a more compressed timeframe than the original capture by reducing or expanding the time duration of each step. i.e., an IO Capture of sixty, one-second steps can be replayed as sixty, one-second steps or sixty, one-minute steps during the test thereby compressing or expanding the test duration time.

In order to convert an IO Trace to an IO Capture, the software tools must collect IO statistics for the corresponding IO step over a defined duration or the user must post process IO Traces to generate the required statistical averages for each IO capture step.

Note 10. IO Capture Tool Algorithms. The selected IO Capture tool will determine IO Stream characteristics such as Random or Sequential Access, R or W IO and data transfer (or block) size. The IO Capture tool should generally describe how they are calculated. For example, in determining whether a given IO is a Random or Sequential Access, the IO Capture tool may use block size, temporal occurrence, quantity of IOs or other criteria in its IO characterization algorithm. More specific disclosure should be provided upon request.

5.3 IO Stimulus Software Tools

IO Stimulus generating 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[1]
3) Issue Random (RND) and Sequential (SEQ) Block level I/O
4) Issue RND and SEQ file system byte addressable I/O
5) Restrict LBA accesses to a particular range of available user LBA space
6) Limit "total unique LBAs used" to a specific value (aka Test Active Range)
7) Set R/W percentage mix % for each test step[2]
8) Set Random/Sequential IO mix % for each test step[3]
9) Set IO Transfer Size for each test step[4]
10) Set Queue Depth for each test step[5]
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.

[1] This feature is necessary for RWSW test step parameter requirements
[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 Individual Streams Test

6.1 Individual Streams Test - Descriptive Note:

General Description:

This Individual Streams test is designed to evaluate the performance of a single IO Stream over a range of Thread Count. The test operator may select Individual IO Stream workloads listed in table 6.3 Recommended Synthetic IO Stream Workloads or any desired IO Stream Access Pattern (including those observed in a real world workload IO Capture).

After a warm-up, the test workload is applied over a range of increasing Thread Count (TC). The Demand Intensity Curve will show the IOPS/TC (or MB/s/TC) operating point that results in the highest IOPS (or MB/s) and/or lowest Response Time and a response time histogram.

Test Flow:

1. **Set Parameters & Conditions.** Set server settings, IO Engine & IO Access Mode.
2. **Select a test workload.** Select a test workload either from the list of recommended synthetic IO Streams or a user selected IO Streams.
3. **Warm-up.** Run the selected test workload for a warm-up period for a specified duration at a specified Thread Count. Note: No pre-conditioning is required.
4. **Run the test workload.** After a warm-up period, run the workload for a given duration at a series of TC settings, e.g., Duration=30 seconds, TC = [1,2,4,8,16,32,64,128].
5. **Record the data.** Record the prescribed data for each of the Individual Streams tested. Data will include IOPS, MB/s and Response Times (average, 5 9s and maximum).
6. **Create a Demand Intensity Curve.** Create a Demand Intensity (DI) Curve showing IOPS or MB/s vs Thread Count and Average Response Times (ART). The DI Curve shall show MAX, MIN and MID IOPS (or MB/s) points as defined in the test pseudo code.
7. **Create Response Time Histograms.** Create Response Time Histograms for Max, Mid and Min IOPS (or MB/s) points showing the distribution of IOs across time bins. Report minimum, average, maximum and 5 9s Quality of Service Response Times.
8. **Create a Confidence Level Plot Compare.** Create a plot that compares Response Times, Thread Counts and MB/s for Max, Min and Mid IOPS (or MB/s) Operating points.
9. **Create CPU System Usage % vs TC plot.** Create a CPU Usage % v IOPS/MB/s and Total Thread Count, showing the CPU System Usage %, Thread Count setting and the IOPS and MB/s observed.

Test Results:

The test plots present IO rate and response times for a varying number of TC settings. Secondary plots present response time statistics (or histograms) for the selected operating points and IOPS/MB/s, Response Times & CPU System Usage % v TC. See 6.5.2 for examples of Plots.

Test Interpretation:

The Individual Streams test is designed to present Persistent Memory performance for specific IO Streams Access Patterns. The IOPS/MB/s TC Loops show how performance changes with TC while CPU System Usage % shows overall CPU saturation.

Note: IO Stream workloads of smaller data transfer sizes (such as 64B, 128B, 256B and 512B) are typically of greater interest as in-memory application IOs tend to be of smaller data transfer size.

6.2 Individual Streams Test Pseudo Code

For (ActiveRange=100%, optional ActiveRange=Test Operator Choice, Access Pattern = (R/W Mix=RW1, Block Size=BS1, Random, Max TC=Test Operator Choice)

1. **Set Server & Test Settings**

   1.1. Record PM Server Settings for later reporting
   - 1.1.1 Manufacturer
   - 1.1.2 Model No.
   - 1.1.3 Motherboard
   - 1.1.4 CPU Type and No.
   - 1.1.5 CPU Clock Speed
   - 1.1.6 CPU Cores & Threads/Core
   - 1.1.7 OS type and version
   - 1.1.8 PM driver version

   1.2. Set Persistent Memory Module Settings and record for later reporting
   - 1.2.1 NUMA – default ON
   - 1.2.2 Page Size Memory
   - 1.2.3 Interleave – default ON
   - 1.2.4 Region Configuration
   - 1.2.5 Namespace Configuration
   - 1.2.6 File System Configuration

   1.3. Test Settings: IO Engine & Access Mode set & record for later reporting
   - 1.3.1 Select IO Engine:
     - FSDax
     - Sector
     - DevDax
   - 1.3.2 Select IO Access Mode:
     - Blockio_sync
     - Memcopy_nosync
     - Memcopy_OSsync
     - Memcopy_CLWB
     - Memcopy_Non Temporal Writes

2. **Select the Test Workload (or IO Stream)**

   2.1 Select a synthetic IO Stream from section 6.3 recommended IO Streams

   2.2 Select a user choice synthetic IO Stream

3. **Warm-up using the Test Workload but with R/W Mix=0% (100% Write)**

   3.1 Set test parameters and record for later reporting
   - PURGE – No Purge
   - Volatile write cache user choice
   - Thread Count or OIO/Thread: 16 or user choice
   - Data Pattern: Required = Random, Optional = Test Operator Choice

   3.2 Run Warm Up Access Pattern, using the required ActiveRange=100% or the corresponding desired optional ActiveRange.
   - Duration=30 sec
   - TC setting=16
   - RW Mix = 100% W

4. **Run the Test Workload Access Pattern while varying demand settings:**

   4.1 Set test parameters and record for later reporting
   - Volatile device write cache user choice
   - Data Pattern: Same as Warm-up

IO Access Mode: Same as Test Settings

PM Module: Interleaved or Single; Capacity

Where Max TC=128, Vary TC using TC = [1,2,4,8,16,32,64,128]

5. **Apply Test Workload**

   5.1 Apply Test Workload Duration=5 seconds for each TC in TC range, using TC = [1,2,4,8,16,32,64,128] or user selected TC range

   5.2 Record elapsed time, IOPS (or MB/s), ART, MRT and Percentage CPU Utilization by System (SYS_CPU) every 1 second.

   5.3 Using Test Workload Data:

   5.4 Plot All TP vs Time - IO Stream MB/s vs Time for Warm-up and TC range (or loop).

   5.5 Plot IO Streams v TC - IO Stream MB/s & ART vs TC

   5.6 Plot IO Streams v CPU Usage % - IO Stream MB/s & ART vs CPU Usage %

6. **Determine MaxIOPS, MinIOPS and a minimum of 1 MidIOPS operating point:**

   6.1 A MaxIOPS point shall be chosen from the TC operating points, such that:

   6.2 The MaxIOPS point shall be chosen to represent the operating point where the IOPS are highest while achieving a reasonable ART.

   6.3 The MinIOPS point is defined to be the operating point where Thread Count=1 and OIO/Thread=1.

   6.4 Choose a minimum of 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 the IOPS value between MinIOPS and MaxIOPS – recommended value at 85% of MaxIOPS point.

   6.5 MB/s – MaxIOPS, MinIOPS & MidIOPS may be calculated and reported as MaxMB/s, MidMB/s and MinMB/s

7. **Response Time Histogram at Maximum IOPS (or Maximum MB/s):**

   7.1 Select a TC operating point that yields maximum IOPS using the lowest number of Total Outstanding IO (TOIO=Thread Count x Queue Depth)

   7.2 Execute R/W Mix=RW1, Random IO, Block Size=BS1 KiB for 20 seconds. 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 20 second capture. Default bin size=100 nanoSec.

8. **Response Time Histogram at Minimum IOPS (or Minimum MB/s):**

   8.1 Select Thread Count=1 operating point

   8.2 Execute R/W Mix=RW1, Random IO, Block Size=BS1 KiB for 20 seconds. 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 20 second capture. Default bin size=100 nanoSec.

9. **Response Time Histogram at MidIOPS (or Mid MB/s) operating point:**

   9.1 Select a Thread Count operating point that yields an IOPS result that lies approximately 85% between Maximum IOPS in (6) above, and the Minimum IOPS in (7) above.

   9.2 Execute R/W Mix=RW1, Random IO, Block Size=BS1 KiB for 20 seconds. 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 20 second capture. Default bin size=100 nanoSec

10. **Process and plot the accumulated data, report as specified in section 6.4.**

6.3 Recommended IO Steams for Individual Streams Test

The following IO Stream Access patterns are recommended to be run for this test:

Recommended Synthetic IO Stream Access Patterns			
RND 64B Read	RND 64B Write	SEQ 64B Read	SEQ 64B Write
RND 4K Read	RND 4K Write	SEQ 4K Read	SEQ 4K Write
RND 16K Read	RND 16K Write	SEQ 16K Read	SEQ 16K Write
RND 64K Read	RND 64K Write	SEQ 64K Read	SEQ 64K Write

Figure 6-1. Table of Recommended Synthetic IO Stream Access Patterns

6.4 Test Specific Reporting for **Individual Streams** Test

Reporting requirements common to all tests are documented in Section 4. Reports specific to the Individual Streams Test follow.

6.4.1. Test Settings & Set Up Configuration

The Test Operator shall disclose:

1. Server Settings – mfgr, model, motherboard, CPU type and number, CPU clock speed, CPU cores, RAM, OS and PM driver
2. PMEM settings - NUMA, Page Size Memory, PMEM modules and capacity, interleave setting, region configuration, namespace configuration, file system configuration
3. IO Engine – Sector, FSDax, DevDax
4. IO Access mode - and blockio_sync, memcopy_nosync, memcopy_OSsync, memcopy_CLWB and memcopy_Non Temporal Writes
5. See Appendix B: Sample Test Report Header

6.4.2. Test Measurement Report

The Test Operator shall generate Measurement Plots for:

P1. Workload Streams Distributions – IO Stream Access Pattern
P2. All IOPS vs Time – Warm-up and TC Loop
P3. All Throughput (MB/s) vs Time - Warm-up and TC Loop
P4. Demand Variation – TP vs TC
P5. Demand Intensity – Individual IO Stream
P6. DI Outside Curve – IOPS & ART vs MaxIOPS, MinIOPS & MidIOPS operating points
P7. DI Outside Curve – TP & ART vs MaxMB/s, MinMB/s & MidMB/s operating points
P8. Max IOPS Histogram – Max IOPS & MB/s vs IO Distribution Time & Count
P9. Confidence Level Plot Compare – Max, Mid and Min IOPS & MB/s v TC & ART
P10. Throughput & ART v Total TC
P11. CPU Sys Usage %, Throughput v Total TC

6.5 Sample Data

6.5.1. Sample Data Set-up

The following hardware, software, workload and storage set ups were used in taking the sample data plots in this section. Note that not all of the required measurement reports are included.

Hardware Platform

Intel Wolfpass; OS - Linux Ubuntu 20.04.4
DDR4 256 GB ECC RAM
Dual 24 core 2.4Ghz Intel XEON 8260 CPU
24 Cores/48 Threads/CPU
NUMA enabled

Software Platform

DAX FS; Memcopy_OSsync
Page Size Memory 2MB
Test Software: CTS PM PMDK

Synthetic Workload

RND 64b Read
See Section 6.3 Recommended Synthetic IO Stream Workloads
Demand Intensity - max OIO=128, TC=128 QD=1

Storage

6 x 256 GB DCPM
Total Storage 1198 GB
Interleaved

6.5.2 Sample Data Plots

The following sample data plots should be included in the SNIA PM PTS Report. Each plot shall include a PM PTS Report Header as set forth in Appendix B.

P1 – Workload Streams Distribution – Single Workload IO Stream

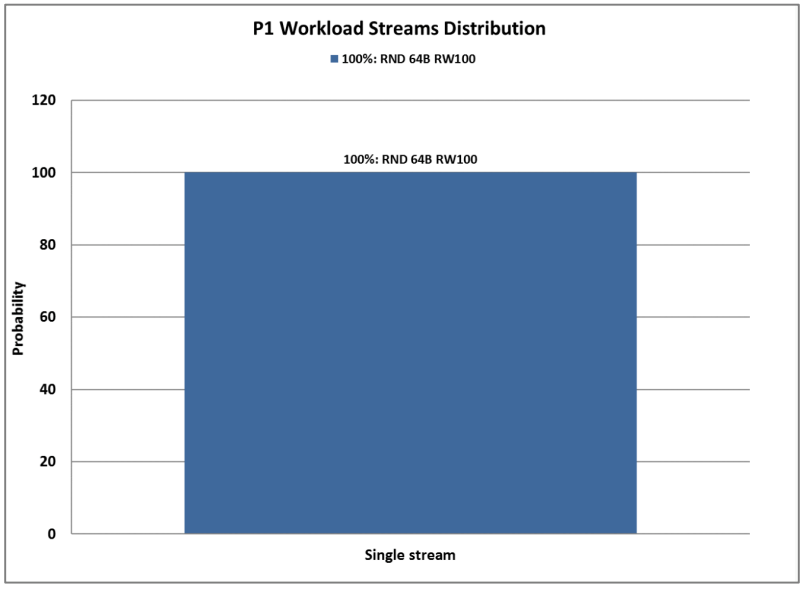


Figure 6-2. Test Workload: RND 64B Read

P2 - All IOPS v Time – Warm-up & TC Loop

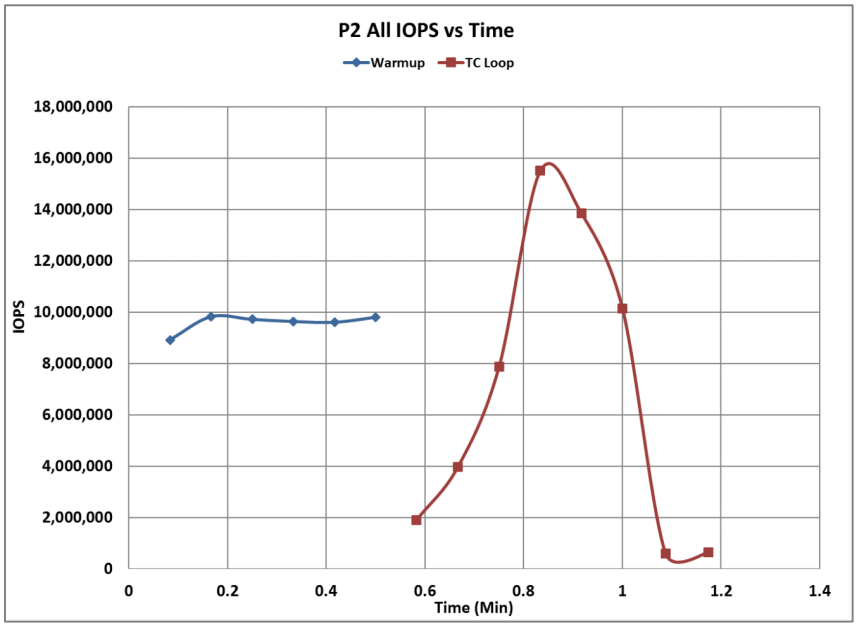


Figure 6-3. Warm-up & TC Loop - IOPS

P3 - All Throughput v Time – Warm-up & TC Loop

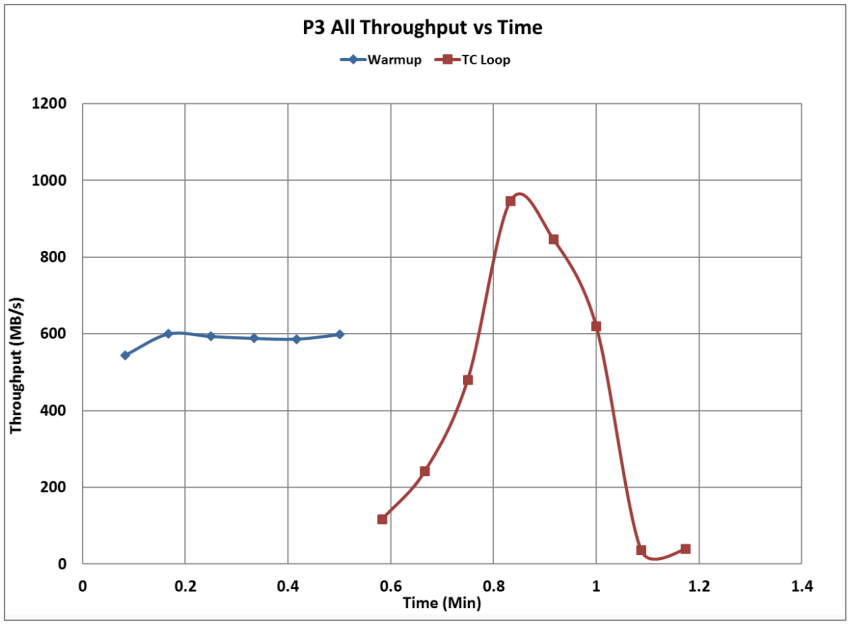


Figure 6-4. All Throughput v Time – Warm-up & TC Loop

P4 - Demand Variation – Throughput v TC

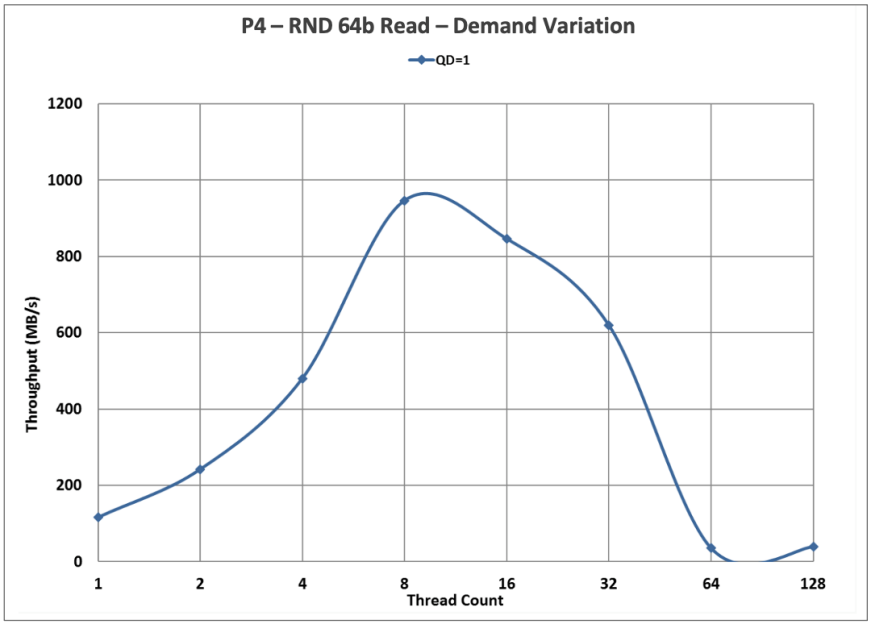


Figure 6-5. Demand Variation Plot – TP v TC

P5 – Demand Intensity – Individual IO Stream

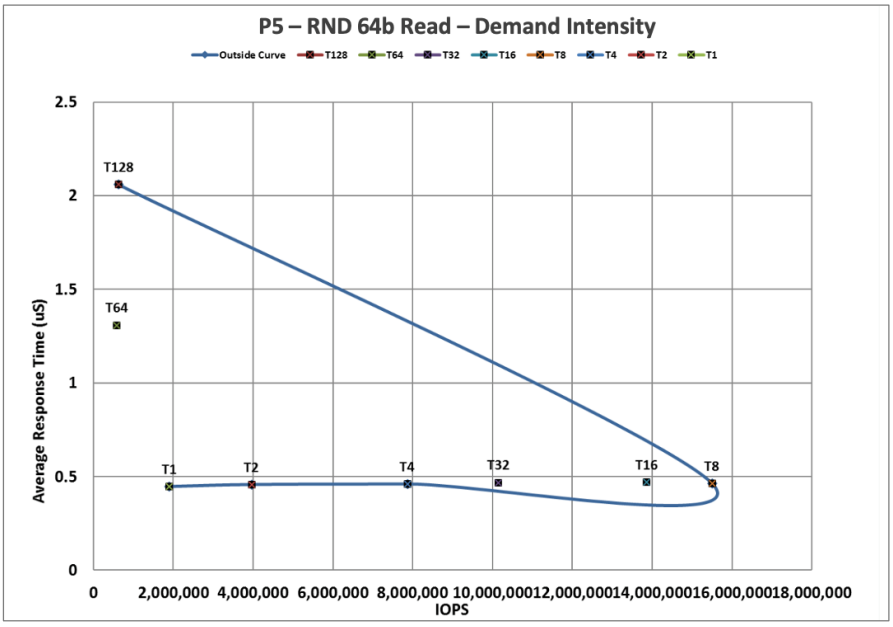


Figure 6-6. Demand Intensity Plot – Individual IO Stream

P6 - Demand Intensity Outside Curve – Max, Min, Mid IOPS Points

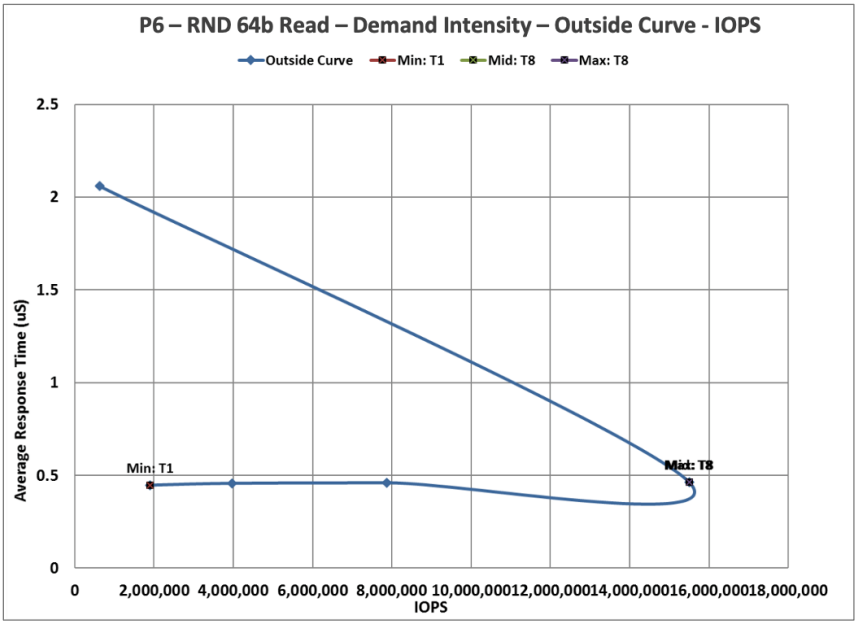


Figure 6-7. Demand Intensity Plot – Max, Min, Mid IOPS Points

Note: Mid and Max Operating point algorithm overlaps at T8/Q1

P7 - Demand Intensity Outside Curve – Max, Min, Mid MB/s Points

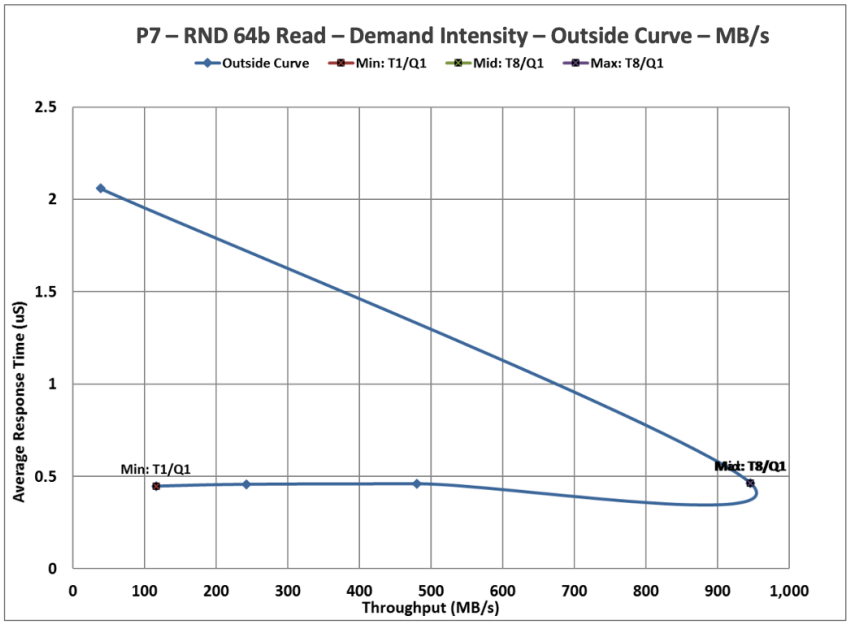


Figure 6-8. DI Outside Curve – Max, Min, Mid MB/s Points

Note: Mid and Max Operating point algorithm overlaps at T8/Q1

P8 - Max IOPS Histogram

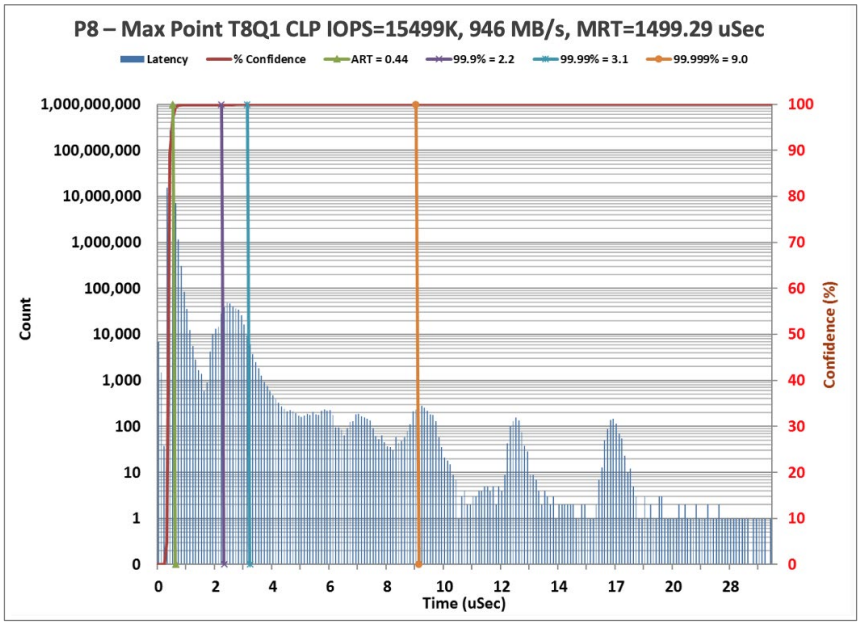


Figure 6-9. Max IOPS Histogram

P9 - Confidence Level Plot Compare

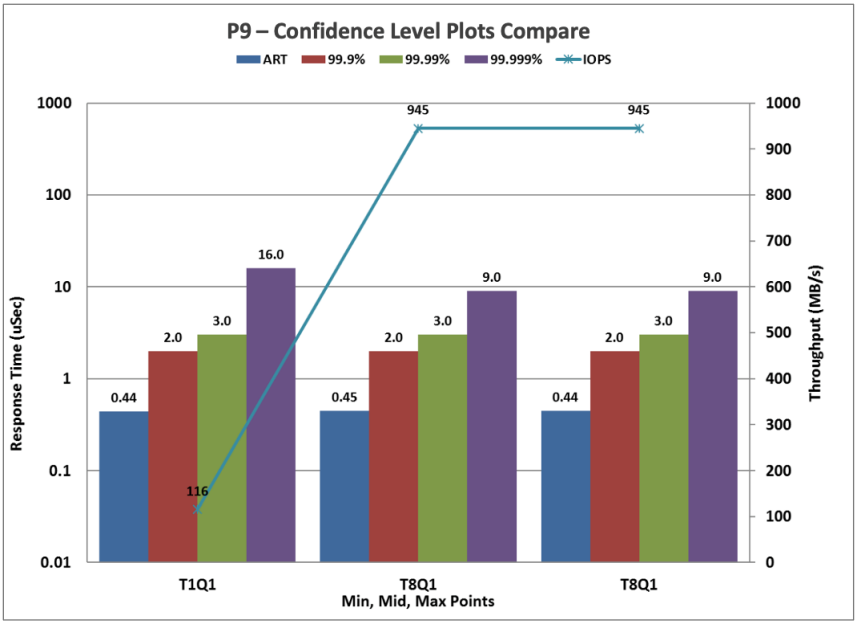


Figure 6-10. Confidence Level Plot Compare

P10 – Throughput & ART v Total TC

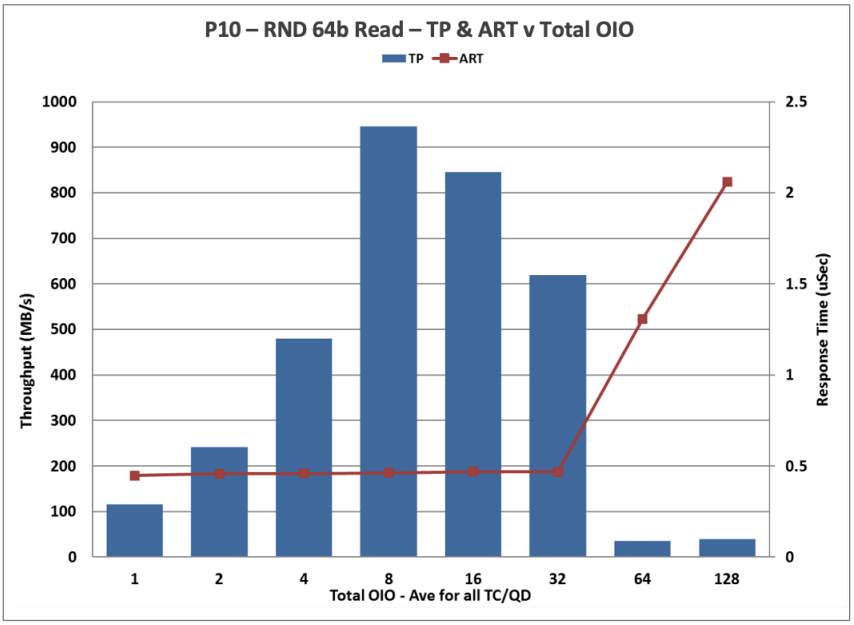


Figure 6-11. Throughput & ART v Total TC

P11 – Throughput & CPU System Usage % v Total TC

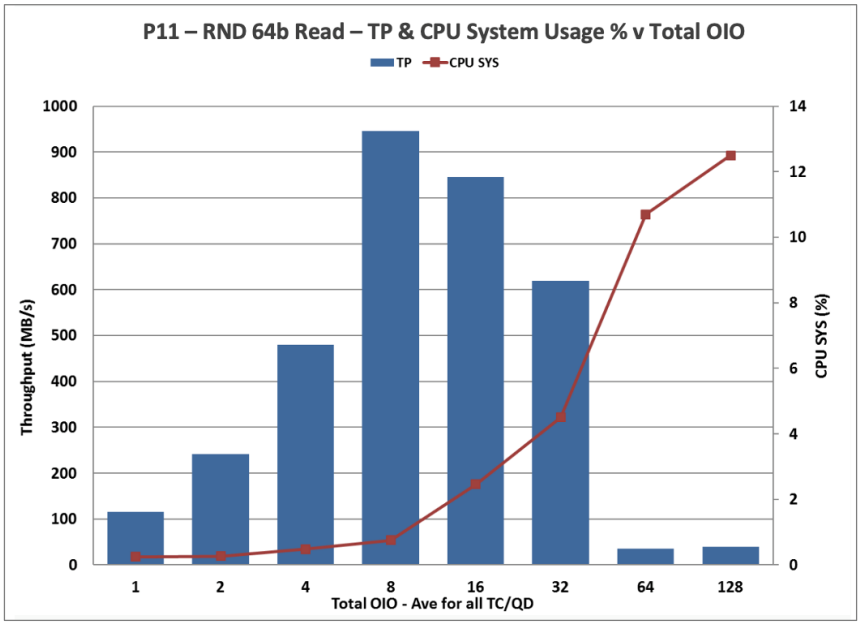


Figure 6-12. Throughput & CPU System Usage % v Total TC

7 Composite Streams Test

7.1 Composite Streams Test Descriptive Note:

General Purpose:

The purpose of the Composite Streams Test is to measure the performance of a composite IO Stream workload observed in a real world workload.

The Composite Streams Test applies a composite IO Stream workload across a range of Thread Counts. The test workload is derived from either a synthetic or real-world workload listed in table 7.3.

Test Flow:

1. **Set Parameters & Conditions.** Set server settings, IO Engine & IO Access Mode.
2. **Select a test workload.** Select a test workload either from the list of Reference Real World Workloads (table 7.3) or a user selected IO Stream composite.
3. **Warm-up.** Run the selected test workload for a warm-up period of a specified duration at a specified Thread Count. Note: No pre-conditioning is required.
4. **Run the test workload.** After a warm-up period, run the workload for a given duration at a series of TC settings, e.g., Duration=30 seconds, TC = [1,2,4,8,16,32,64,128].
5. **Record the data.** Record the prescribed data for the Composite IO Streams tested. Data includes IOPS, MB/s and Response Times (average, 5 9s and maximum).
6. **Create a Demand Intensity Curve.** Create a Demand Intensity (DI) Curve showing IOPS or MB/s vs Thread Count and Average Response Times. The DI Curve shall show MAX, MIN and MID IOPS (or MB/s) operating points as defined in the test pseudo code.
7. **Create Response Time Histograms.** Create Response Time Histograms for Max, Mid and Min IOPS (or MB/s) operating points showing the distribution of IOs across time bins. Report minimum, average, maximum and 5 9s Quality of Service Response Times.
8. **Create a Confidence Level Plot Compare.** Create a plot that compares Response Times, Thread Counts and MB/s for Max, Min and Mid IOPS (or MB/s) operating points.
9. **Create CPU Usage % vs TC plot.** Create a CPU Usage % v IOPS (or MB/s) and Thread Count, showing the CPU Usage %, Thread Count setting and the IOPS (or MB/s) observed.

Test Results:

The Composite Streams test is identical to the Individual Streams test except that the Composite Streams test uses a composite of 9 IO Streams whereas the Individual Streams test is a single IO Stream Access Pattern. Both tests present IO rate and response times at varying TC settings with histogram, Average Response Time and CPU System Usage % plots.

Test Interpretation:

The performance of a multiple stream Composite workload is different from a single IO Stream workload. The Composite IO Stream workload more closely emulates the expected performance of PM when exposed to multiple IO Stream real world workloads.

IOPS and MB/S performance can be compared between Individual Stream and Composite Stream tests. Also, the reader should observe the difference in Average Response Times (P10) and CPU System Usage % (P11) between the Individual and Composite workloads.

7.2 Composite Streams Test Pseudo Code

For (ActiveRange=100%, optional ActiveRange=Test Operator Choice, Access Pattern = (R/W Mix=RW1, Block Size=BS1, Random, Max TC=Test Operator Choice)

4.1 **Set Server & Test Settings**

1.1 Record PM Server Settings for later reporting
Manufacturer
Model No.
Motherboard
CPU Type and No.
CPU Clock Speed
CPU Cores & Threads/Core
OS type and version
PM driver version

1.2 Set Persistent Memory Module Settings and record for later reporting
NUMA – default ON
Page Size Memory
Interleave – default ON
Region Configuration
Namespace Configuration
File System Configuration

1.3 Test Settings: IO Engine & Access Mode set & record for later reporting
1.3.1. Select IO Engine:
FSDax
Sector
DevDax
1.3.2. Select IO Access Mode:
Blockio_sync
Memcopy_nosync
Memcopy_OSsync
Memcopy_CLWB
Memcopy_Non Temporal Writes

4.2 **Select the Test Workload (or IO Stream)**

1.1. Select a composite IO Stream from section 7.3 recommended IO Streams

1.2. Select a user choice composite IO Stream

4.3 **Warm-up using the Test Workload but with R/W Mix=0% (100% Write)**

3.1 Set test parameters and record for later reporting
PURGE – No Purge
Volatile write cache user choice
Thread Count or OIO/Thread: 16 or user choice
Data Pattern: Required = Random, Optional = Test Operator Choice

4. **Run Warm Up Access Pattern, using the required ActiveRange=100% or the corresponding desired optional ActiveRange.**
Duration=30 sec
TC setting=16
RW Mix = 100% W

5. **Run the Test Workload Access Pattern while varying demand settings:**

5.1 Set test parameters and record for later reporting
Volatile write cache user choice

Data Pattern: Same as Warm-up

IO Access Mode: Same as Test Settings

PM Module: Interleaved or Single; Capacity

Where Max TC=128, Vary TC using TC = [1,2,4,8,16,32,64,128]

5.2 Apply Test Workload

Apply Test Workload Duration=5 seconds for each TC in TC range, using TC = [1,2,4,8,16,32,64,128] or user selected TC range

Record elapsed time, IOPS (or MB/s), ART, MRT and Percentage CPU Utilization by System (SYS_CPU) every 1 second.

6. **Using Test Workload Data Plot**

6.1 All TP v Time

IO Stream MB/s vs Time for Warm-up

IO Stream MB/s vs Time TC for TC loop

6.2 Individual IO Streams v TC

6.3 IO Stream MB/s & ART vs TC

6.4 IO Streams v CPU Usage %

6.5 IO Stream MB/s & ART vs CPU Usage %

7. **Determine MaxIOPS, MinIOPS and a minimum of 1 MidIOPS operating point:**

7.1 A MaxIOPS point shall be chosen from the TC operating points, such that:

7.2 The MaxIOPS point shall be chosen to represent the operating point where the IOPS are highest while achieving a reasonable ART.

7.3 The MinIOPS point is defined to be the operating point where Thread Count=1 and OIO/Thread=1.

7.4 Choose a minimum of 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 the IOPS value between MinIOPS and MaxIOPS – recommended value at 85% of MaxIOPS point.

7.5 MB/s – MaxIOPS, MinIOPS & MidIOPS may be calculated and reported as MaxMB/s, MidMB/s and MinMB/s

8. **Response Time Histogram at Maximum IOPS (or Maximum MB/s):**

8.1 Select a TC operating point that yields maximum IOPS using the lowest number of Total Outstanding IO (TOIO=Thread Count x Queue Depth)

8.2 Execute R/W Mix=RW1, Random IO, Block Size=BS1 KiB for 20 seconds. 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 20 second capture. Default bin size=100 nanoSec.

9. **Response Time Histogram at Minimum IOPS (or Minimum MB/s):**

9.1 Select Thread Count=1 operating point

9.2 Execute R/W Mix=RW1, Random IO, Block Size=BS1 KiB for 20 seconds. 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 20 second capture. Default bin size=100 nanoSec.

10. **Response Time Histogram at MidIOPS (or Mid MB/s) operating point:**

10.1 Select a Thread Count operating point that yields an IOPS result that lies approximately 85% between Maximum IOPS in (6) above, and the Minimum IOPS in (7) above.

10.2 Execute R/W Mix=RW1, Random IO, Block Size=BS1 KiB for 20 seconds. 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 20 second capture. Default bin
        size=100 nanoSec
```

11. **Process and plot the accumulated data, report as specified in section 7.4.**

7.3 Recommended IO Steams for Composite Streams Test

Figure 7-1 lists Composite IO Stream Access Patterns that are recommended to be run for this test. The SNIA Reference Real World Workloads Library lists the IO Capture Cumulative Workloads.

SNIA Reference Real World Workloads Library Listing	
Retail Web Portal	Drive0 Drive1 – 24-hour; 290 5 min steps
GPS Nav Portal Boot Drive	DriveC – 24-hour; 720 2 min steps
GPS Nav Portal Storage Drive	Drive0 – 24-hour; 720 2 min steps
VDI Storage Cluster	6 Drive Cluster – 13.8-hour; 158 5 min steps

Figure 7-1 – SNIA Reference Real World Workloads Library Listing

The SNIA Reference Real World Workloads Library Listing, Workloads Table and IO Stream Maps can be viewed at https://www.snia.org/technology-focus-areas/physical-storage/real-world-workloads/reference-real-world-workloads.

The IOTTA Repository of Real World Workloads lists the individual time-step IO Streams for the above listed Reference Real World workloads.

7.4 Test Specific Reporting for **Individual Streams** Test

Reporting requirements common to all tests are documented in Section 4. Reports specific to the Individual Streams Test follow.

7.4.1. Test Settings & Set Up Configuration

The Test Operator shall disclose:

1. Server Settings – mfgr, model, motherboard, CPU type and number, CPU clock speed, CPU cores, RAM, OS and PM driver
2. PMEM settings: NUMA, Page Size Memory, PMEM modules and capacity, interleave setting, region configuration, namespace configuration, file system configuration
3. IO Engine – Sector, FSDax, DevDax
4. IO Access mode - and blockio_sync, memcopy_nosync, memcopy_OSsync, memcopy_CLWB and memcopy_Non Temporal Writes
5. See Appendix B: Sample Test Report Header

7.4.2. Test Measurement Report

The Test Operator shall generate Measurement Plots for:

P1. Workload Streams Distributions – IO Stream Access Pattern
P2. All IOPS vs Time – Warm-up and TC Loop
P3. All Throughput (MB/s) vs Time - Warm-up and TC Loop
P4. Demand Variation – MB/s vs TC
P5. Demand Intensity – Cumulative 9 IO Stream
P6. DI Outside Curve – IOPS & ART vs Max, Min & Mid IOPS operating points
P7. DI Outside Curve – MB/s & ART vs Max, Min & Mid MB/s operating points
P8. Max IOPS Histogram – Max IOPS & MB/s vs IO Distribution Time & Count
P9. Confidence Level Plot Compare – Max, Mid and Min IOPS & MB/s v TC & ART
P10. Throughput & ART v Total TC
P11. Throughput & CPU Sys Usage % v Total TC

7.5 Sample Data

6.5.2. Sample Data Set-up

The following hardware, software, workload and storage set ups were used in taking the sample data plots in this section. Note that not all of the required measurement reports are included.

Hardware Platform

Intel Wolfpass; OS – Linux Ubuntu 20.04.4
DDR4 256 GB ECC RAM
Dual 24 core 2.4Ghz Intel XEON 8260 CPU
24 Cores/48 Threads/CPU
NUMA enabled

Software Platform

FSDax; Memcopy_OSsync
Page Size Memory 2MB
Test Software: CTS PM PMDK

Synthetic Workload

Retail Web Portal 9 IO Stream Cumulative Workload
SNIA CMSI Reference Real World Workload Library
SNIA IOTTA Repository
Demand Intensity - max OIO=128, TC=128 QD=1

Storage

6 x 256 GB DCPMM; Total Storage = 1198 GB
Interleaved

7.5.2 Sample Data Plots

The following sample data plots should be included in the SNIA PM PTS Report. Each plot shall include a PM PTS Report Header as set forth in Appendix B.

P1 – Workload Streams Distribution – Retail Web Portal 9 IO Streams

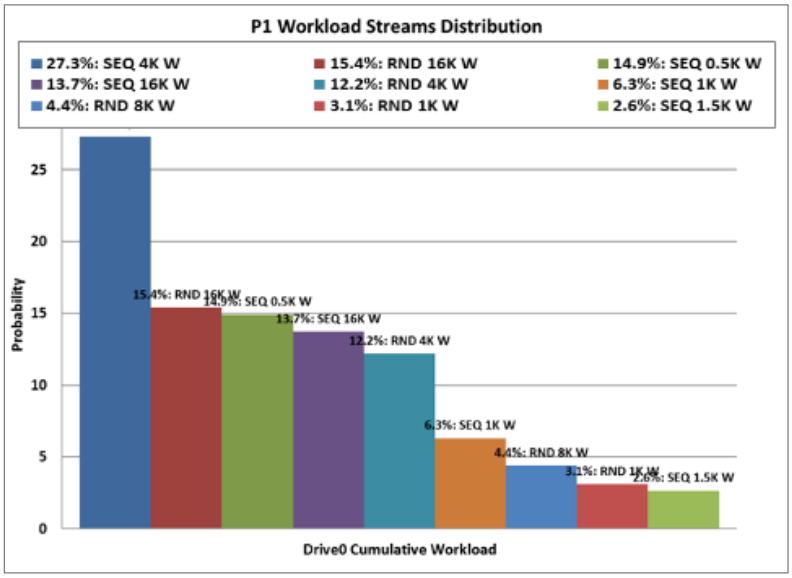


Figure 7-2. Test Workload: Retail Web Portal 9 IO Stream Composite

P2 - IOPS v Time – Warm-up & TC Loop

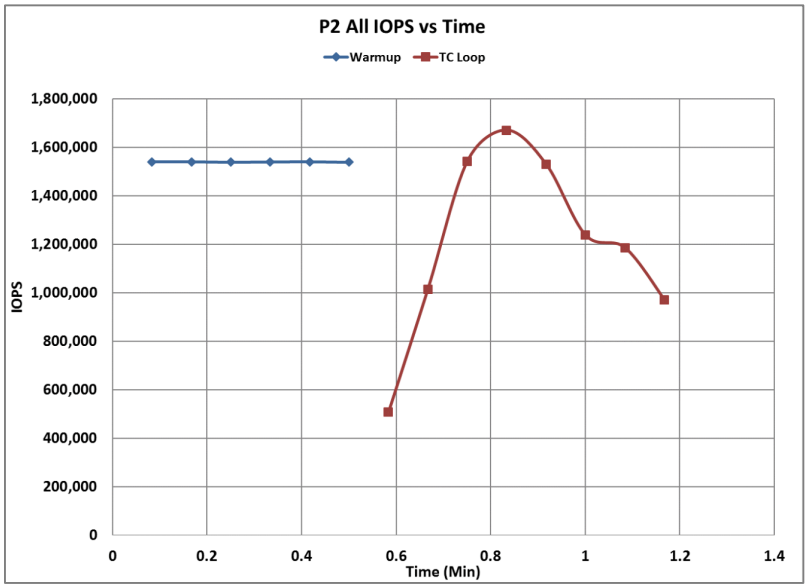


Figure 7-3. Warm-up & TC Loop - IOPS

P3 - Throughput v Time – Warm-up & TC Loop

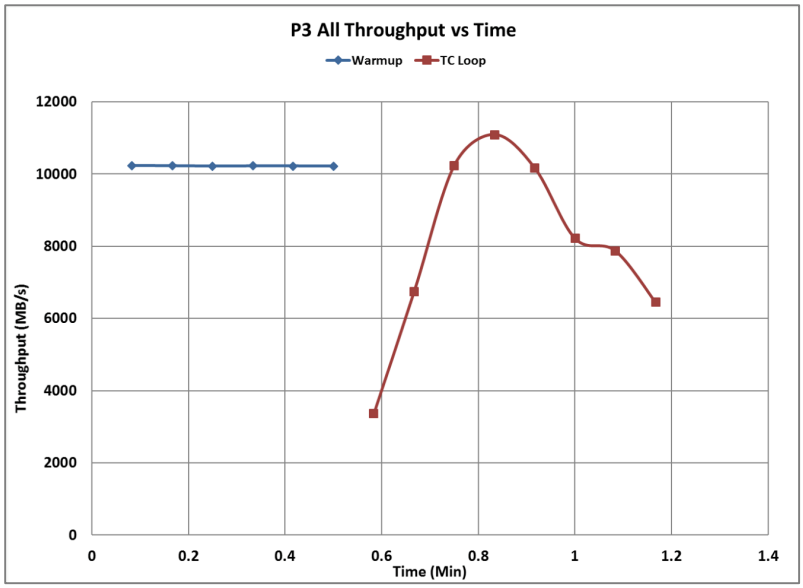


Figure 7-4. Throughput v Time – Warm-up & TC Loop

P4 - Demand Variation – Throughput v TC

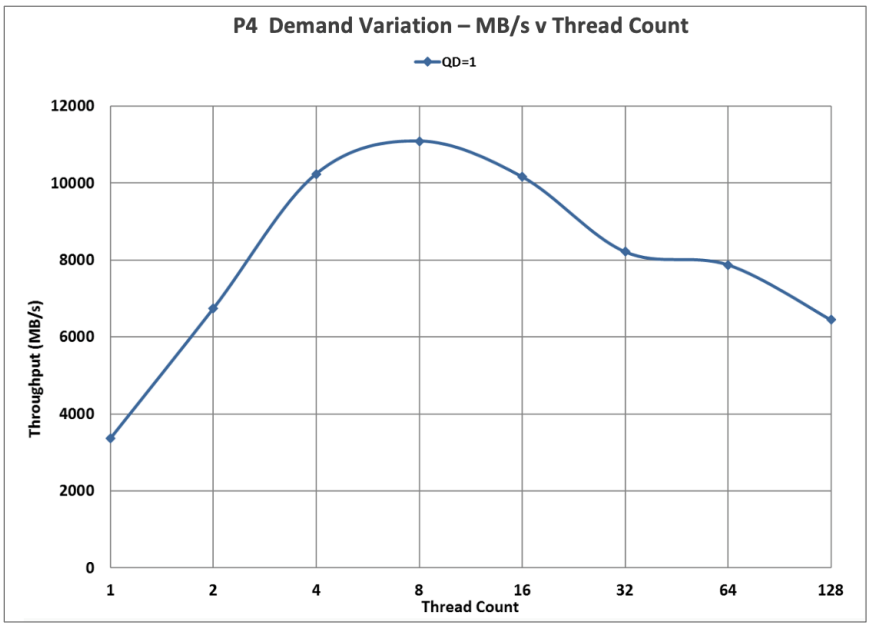


Figure 7-5. Demand Variation Plot – TP v TC

P5 – Demand Intensity - Cumulative Workload

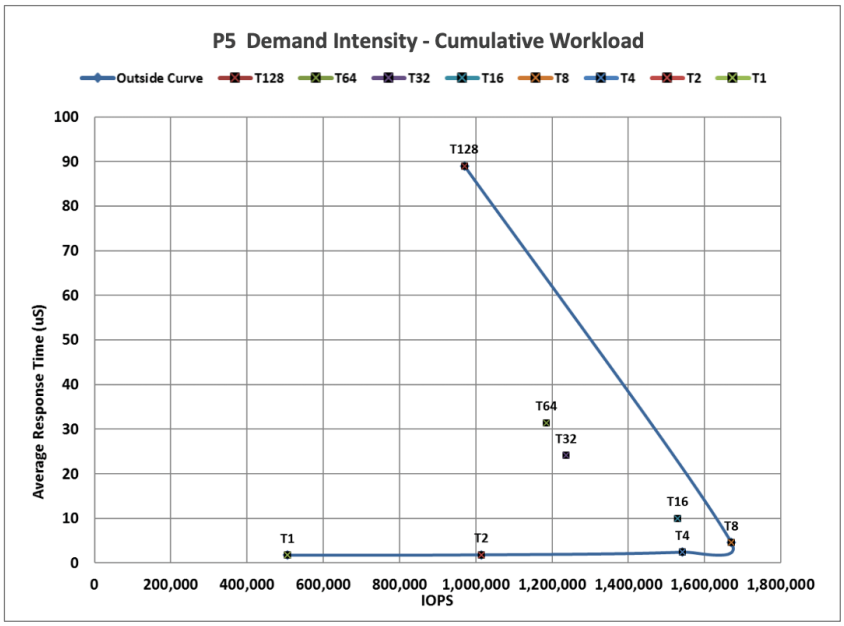


Figure 7-6. Demand Intensity Plot – Cumulative Workload

P6 - Demand Intensity Outside Curve – MinIOPS, MidIOPS, MaxIOPS

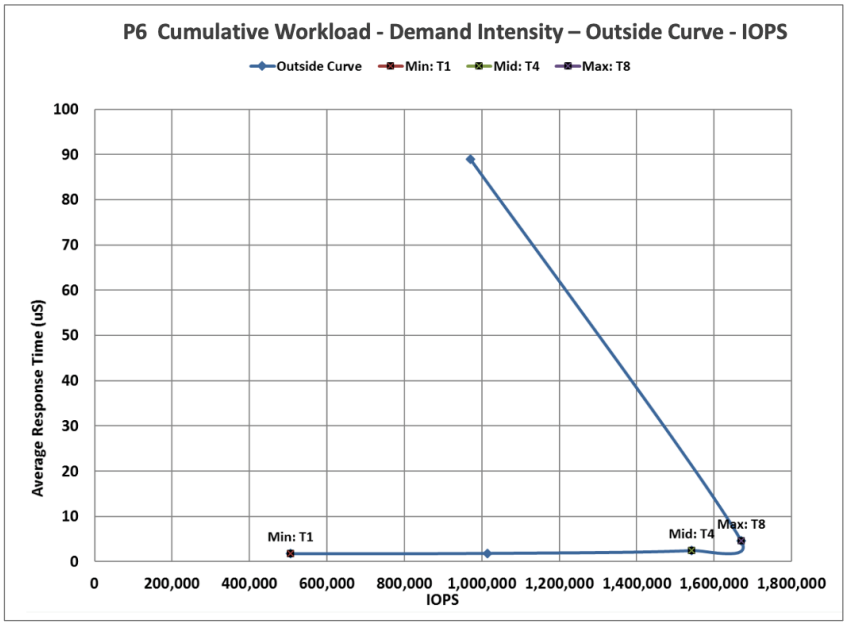


Figure 7-7. DI Outside Curve – Max, Min, Mid IOPS Points

P7 - Demand Intensity Outside Curve – MinMB/s, MidMB/s, MaxMB/s

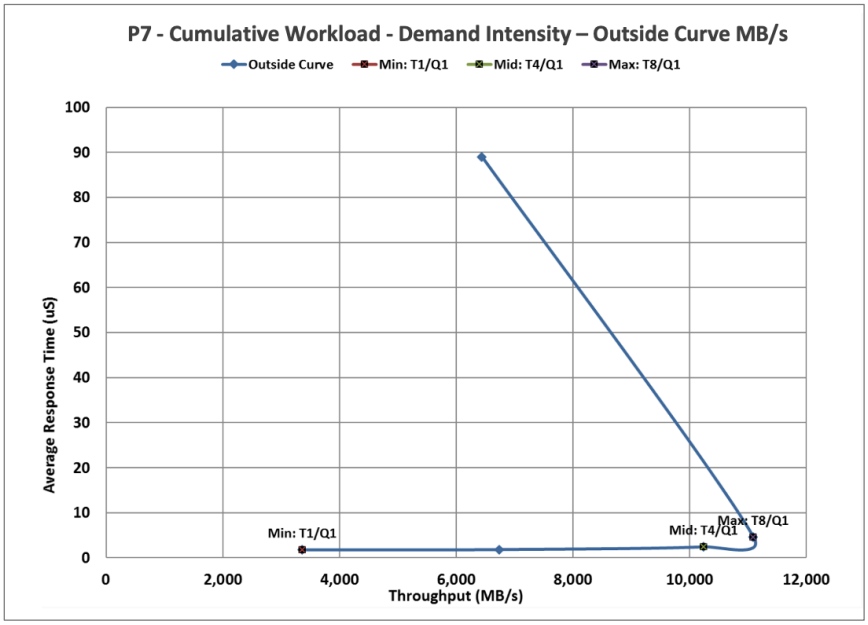


Figure 7-8. DI Outside Curve – Max, Min, Mid MB/s Points

P8 - Max IOPS Histogram

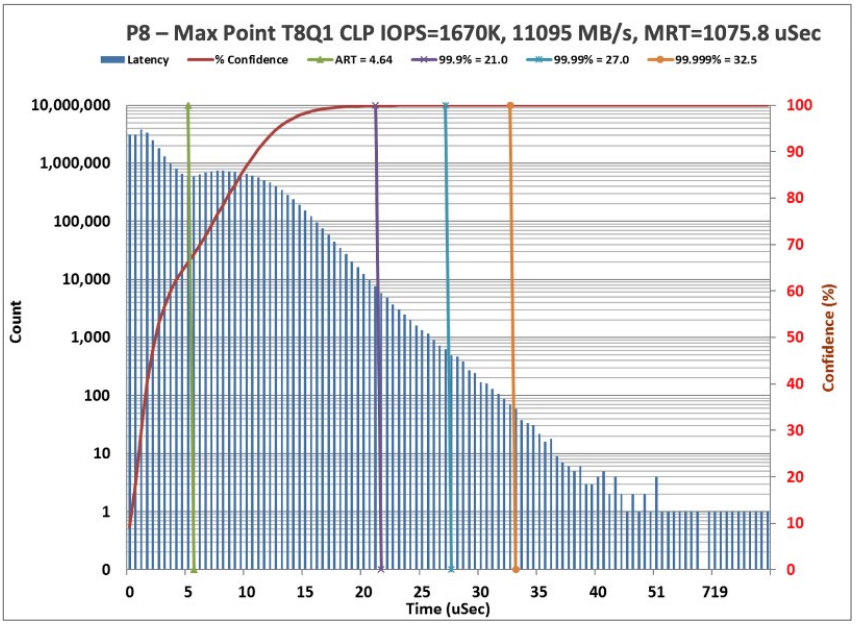


Figure 7-9. Max IOPS Histogram

P9 - Confidence Level Plot Compare

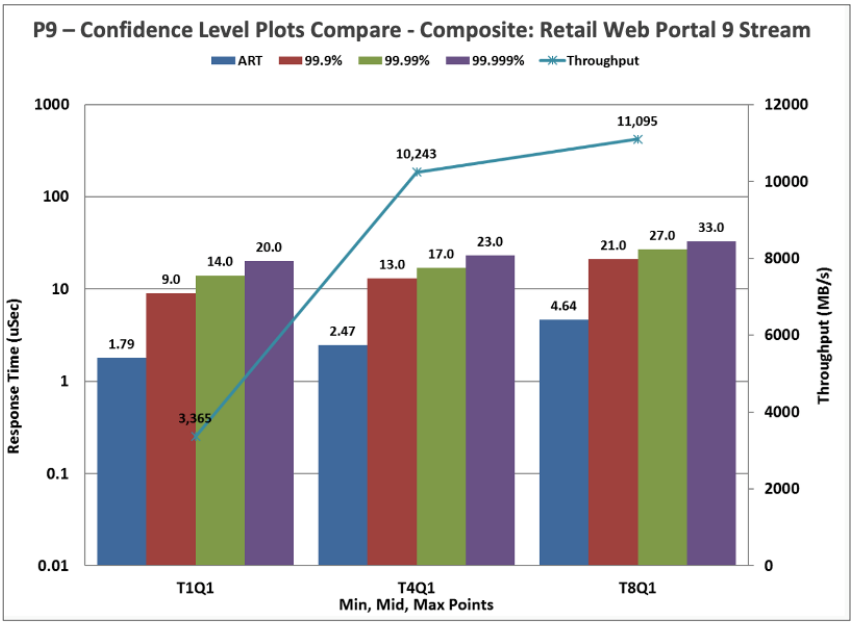


Figure 7-10. Confidence Level Plot Compare

P10 – Throughput & ART v Total TC

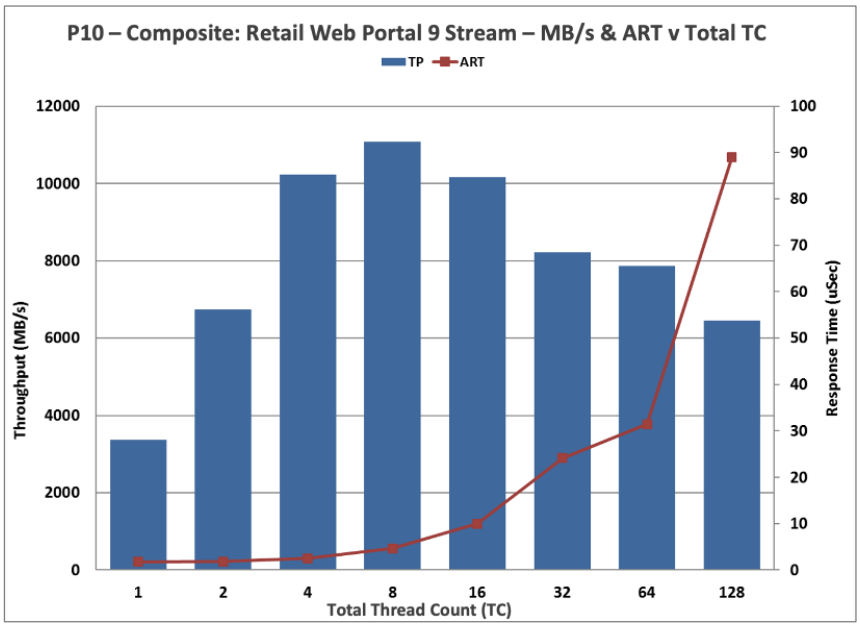


Figure 7-11. Throughtput & Average Response Time v Total TC

P11 – Throughput & CPU Usage % v Total TC

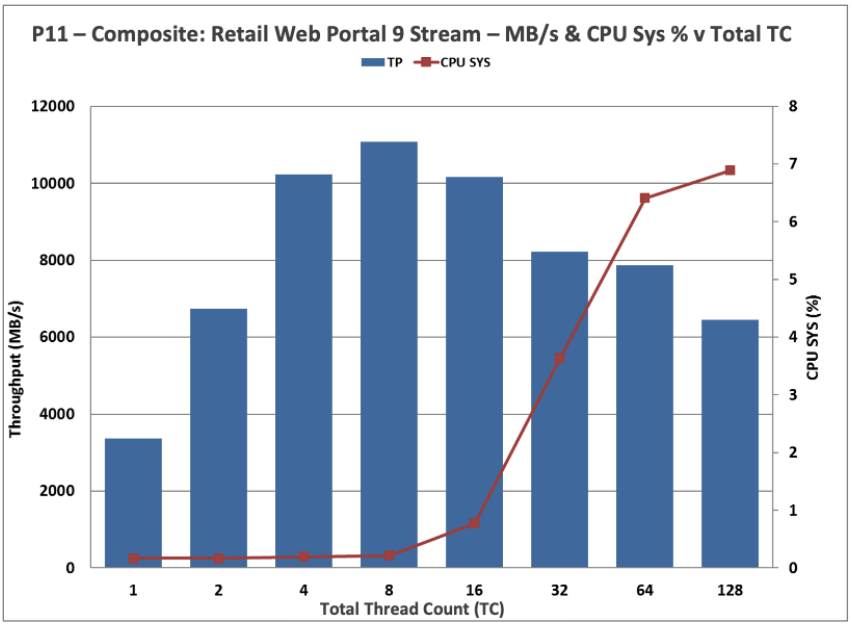


Figure 7-12. Throughput & CPU Sys Usage % v Total TC

8 Replay-Individual Streams Test

8.1 Replay-Individual Streams Test Descriptive Note

General Purpose:

The purpose of the Replay-Individual Streams test is to measure and compare the performance of individual IO Streams (Individual Streams Segment) and the sequence and combinations of IO Streams (Replay Segment) observed in a Real World Workload IO Capture.

The Replay-Individual Streams test begins with an optional warm-up. The Individual Streams segment is a measure of the performance of each of the individual IO Streams observed in the IO Capture Cumulative Workload. See Note 8 – Cumulative Workload.

The Replay segment next applies the sequence of IO Stream combinations observed during the real-world workload IO Capture. See Note 9 – Replay-Individual Streams Test.

Replay and Individual Streams segment performance are reported in IOPS, MB/s and Response Times. Replay segment results can then evaluated in conjunction with the performance of each individual IO Stream.

Test Flow:

1. **Set Parameters & Conditions.** Set server settings, IO Engine & IO Access Mode.
2. **Select a test workload.** Select a test workload either from the list of Reference Real World Workload Cumulative Workload or a user selected IO Stream Capture.
3. **Warm-up.** Optionally run a warm-up of an Individual Streams or Composite Streams test. Otherwise begin with the Individual Streams Segment.
4. **Run the Individual Streams Segment.** After the optional warm-up period, run each of the individual IO Streams from the Cumulative Workload for a given duration (e.g., duration = 1 minute) at a TC setting of user choice (e.g., TC=16).
5. **Run the Replay Segment.** Immediately after the last Individual Stream is tested, run a Replay of the sequence of IO Stream Combinations observed in the IO Capture. Set the duration of each step for a given duration (e.g., time-step duration = 1 second) at a TC setting of max TC observed or user choice (e.g., max TC=128).
6. **Record the data.** Record the prescribed data for the Individual Streams and Replay segments. Data includes IOPS, MB/s and Response Times (average, 5 9s and maximum).
7. **Create a Workload Distribution Plot.** Create a Workloads Distribution plot that shows the individual IO Streams and percentages of the 9 IO Stream Cumulative Workload.
8. **Create IOPS, Throughput (MB/s) and Latency vs Time plots.** Create IOPS, MB/s and Latency vs Time plots for All IOs. Each plot will show the Individual Streams followed by the Replay segment.
9. **Create IOPS & Throughput (MB/s) & ART vs Replay Segment/Individual Streams plots.** Create IOPS, MB/s and ART plots for the Replay Segment and each of the Individual IO Streams tested.
10. **Create an All Throughput (MB/s) vs TC plot.** Create a Throughput vs TC plot that shows the total Throughput and the Total TC vs Time.
11. **Create an IO Stream Map by Quantity of IOs plot.** Create an IO Stream Map by Quantity of IOs plot for the Replay segment showing IO count, IOPS and IO Streams.

Test Results:

The Individual Streams segment presents the performance for each IO Stream of the Cumulative Workload while the Replay segment performance presents the Replay Segment as an average (or “single number”) value.

Test Interpretation:

Single number values for the Replay segment allows for easy comparison among different Replay Tests as IOPS & MB/s v Time show varying IO performance over time.

Performance metrics over time for the Replay segment are useful for analysis and optimization. IO metrics and performance can be observed during the Replay segment to see how performance changes for different workloads.

8.2 Replay-Individual Streams Test Pseudo Code

```
For (ActiveRange=100%, optional ActiveRange=Test Operator Choice, Access Pattern
= (R/W Mix=RW1, Block Size=BS1, Random, Max TC=Test Operator Choice)

  1. Set Server & Test Settings

     1.1. Record PM Server Settings for later reporting
          1.1.1.  Manufacturer
          1.1.2.  Model No.
          1.1.3.  Motherboard
          1.1.4.  CPU Type and No.
          1.1.5.  CPU Clock Speed
          1.1.6.  CPU Cores & Threads/Core
          1.1.7.  OS type and version
          1.1.8.  PM driver version

     1.2. Set Persistent Memory Module Settings and record for later reporting
          1.2.1.  NUMA – default ON
          1.2.2.  Page Size Memory
          1.2.3.  Interleave – default ON
          1.2.4.  Region Configuration
          1.2.5.  Namespace Configuration
          1.2.6.  File System Configuration

     1.3. Test Settings: IO Engine & Access Mode set & record for later
          reporting
          1.3.1.  Select IO Engine:
                  FSDax
                  Sector
                  DevDax
          1.3.2.  Select IO Access Mode:
                  Blockio_sync
                  Memcopy_nosync
                  Memcopy_OSsync
                  Memcopy_CLWB
                  Memcopy_Non Temporal Writes

  2. Select the Test Workload (or IO Stream)

     2.1   Select a Replay test Cumulative Workload from section 8.3

     2.2   Select a user choice Real World Workload Cumulative Workload

  3. Optional Warm-up

     3.1   Select Composite Streams Test as Optional Warm-up

     3.2   Set test parameters and record for later reporting
           3.2.1.  PURGE – No Purge
```

3.2.2. Volatile write cache user choice

3.2.3. Thread Count or OIO/Thread: 16 or user choice

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

3.3 Run Warm Up Access Pattern, using the required ActiveRange=100% or the corresponding desired optional ActiveRange.

3.3.1. Duration=30 sec

3.3.2. TC setting=16

3.3.3. RW Mix = 100% W

4. **Run the Individual Streams Segment**

4.1 Set test parameters and record for later reporting

4.1.1. Volatile write cache user choice

4.1.2. Data Pattern: Same as Warm-up

4.1.3. IO Access Mode: Same as Test Settings

4.1.4. PM Module: Interleaved or Single; Capacity

4.1.5. TC = 16

4.2 Individually Run each IO Stream from the Cumulative Workload

4.3 Apply each IO Stream Duration=60 seconds at TC=16 or user choice

4.4 Record elapsed time, IOPS (or MB/s), ART, MRT and Percentage CPU Utilization by System (SYS_CPU) every 1 second.

5. **Run the Replay Segment**

5.1 Set test parameters and record for later reporting

5.1.1. Volatile write cache user choice

5.1.2. Data Pattern: Same as Warm-up

5.1.3. IO Access Mode: Same as Test Settings

5.1.4. PM Module: Interleaved or Single; Capacity

5.1.5. Max TC = 128 or user choice

5.1.6. Step Duration = 1 sec or user choice

5.2 Run Replay IO time-steps

5.2.1. Duration = 1 second

5.2.2. TC=max (128 or user choice)

6. **Using Test Workload Data Plot:**

6.1 Workload IO Streams Distribution vs Segments (Cumulative Workload & %)

6.2 All Throughput (MB/s) vs Time

IO Stream MB/s vs Time for Warm-up

TC range for TC loop

6.3 IOPS & Average Response Time vs Segments (Replay & Ind Streams)

6.4 MB/s & Average Response Time vs Segments (Replay & Ind Streams)

6.5 IO Streams Distribution by Quantity of IOs (IO Stream Map)

6.6 Replay Segment – Average IOPS

6.7 Replay Segment – Average MB/s

6.8 Replay Segment – Average Response Time

6.9 Replay Segment – 5 9s Quality of Service Response Times

7. Process and plot the accumulated data, report as specified in section 8.4.

8.3 Recommended Workloads for Replay-Individual Streams Test

Figure 8-1 lists SNIA Reference Real World Workloads that are recommended to be run for the Replay-Individual Streams test. The SNIA Reference Real World Workloads Library lists the Real World Workloads from which the Replay segment and Individual Streams are selected. See Note 8 Cumulative Workload - 9 IO Stream Composite Access Patterns and Note 9 Replay-Individual Streams Test.

SNIA Reference Real World Workloads Library Listing	
Retail Web Portal	Drive0 Drive1 – 24-hour; 290 5 min steps
GPS Nav Portal	DriveC – 24-hour; 720 2 min steps
GPS Nav Portal	Drive0 – 24-hour; 720 2 min steps
VDI Storage Cluster	6 Drive Cluster – 13.8-hour; 158 5 min steps

Figure 8-1 – SNIA Reference Real World Workloads – Cumulative 9 IO Stream Access Patterns

The SNIA Reference Real World Workloads Library Listing, Workloads Table and IO Stream Maps can be viewed at https://www.snia.org/technology-focus-areas/physical-storage/real-world-workloads/reference-real-world-workloads.

The IOTTA Repository of Real World Workloads also lists the individual time-step IO Streams for the listed Reference Real World workloads.

8.4 Test Specific Reporting for Replay-Individual Streams Test

Reporting requirements common to all tests are documented in Section 4. Reports specific to the Individual Streams Test follow.

8.4.1. Test Settings & Set Up Configuration

The Test Operator shall disclose:

1. Server Settings – mfgr, model, motherboard, CPU type and number, CPU clock speed, CPU cores, RAM, OS and PM driver
2. PMEM settings: NUMA, Page Size Memory, PMEM modules and capacity, interleave setting, region configuration, namespace configuration, file system configuration
3. IO Engine – Sector, FSDax, DevDax
4. IO Access mode - and blockio_sync, memcopy_nosync, memcopy_OSsync, memcopy_CLWB and memcopy_Non Temporal Writes
5. See Appendix B: Sample Test Report Header

8.4.2. Test Measurement Report

The Test Operator shall generate Measurement Plots for:

P1. Workload Streams Distributions – Cumulative Workload Segments
P2. IO Stream Map by Quantity of IOs & IOPS
P3. Probability of IO Streams by Quantity of IOs
P4. IOPS vs Time – Individual Streams segment & Replay segment
P5. MB/s vs Time - Individual Streams segment & Replay segment
P6. Latency vs Time – Individual Streams segment & Replay segment
P7. Throughput (MB/s) & TC v Time – MB/s and Total TC vs Time
P8. IOPS & Response Times v Segments – Replay & Individual Streams
P9. Throughput (MB/s) & Response Times v Segments – Replay & Ind Streams
P10. Replay Segment Ave IOPS – Ave IOPS for Replay segment
P11. Replay Segment Ave Throughput – Ave Throughput for Replay segment
P12. Replay Segment Ave Response Time – ART for Replay segment
P13. Replay Segment Max Response Time – MRT for Replay segment
P14. Replay Segment 5 9s RT Quality of Service – 5 9s QoS for Replay segment

Note 11. P1 Workload Streams Distributions. The Workload Streams Distributions plot is intended to show the IO Streams that comprise the test workload. P1 shows IO Streams observed over the entire IO Capture and shows the test workload for the Individual Streams test segment.

Unlike the Individual Streams test segment, the Replay test segment workload is comprised of hundreds or more time-step IO Stream combinations. Therefore, it is not practical to list IO Stream combinations for every step of the Replay Segment. However, the Cumulative Workload is useful to show the list of IO Streams from which each step of the Replay test segment is derived.

8.5 Sample Data Set-up

The following hardware, software, workload and storage set ups were used in taking the sample data plots in this section. Note that not all of the required measurement reports are included.

Hardware Platform

Intel Wolfpass; OS – Linux Ubuntu 20.04.4
DDR4 256 GB ECC RAM
Dual 24 core 2.4Ghz Intel XEON 8260 CPU
24 Cores/48 Threads/CPU
NUMA enabled

Software Platform

FSDax; Memcopy_OSsync
Page Size Memory 2MB
Test Software: CTS PM PMDK

Synthetic Workload

Retail Web Portal 9 IO Stream Cumulative Workload
SNIA CMSI Reference Real World Workload Library
SNIA IOTTA Repository
Demand Intensity - max OIO=128, TC=128 QD=1

Storage

6 x 256 GB DCPMM; Total Storage = 1198 GB
Interleaved

8.5.1. Sample Data Plots

The following sample data plots should be included in the SNIA PM PTS Report. Each plot shall include a PM PTS Report Header as set forth in Appendix B.

P1 – Workload IO Streams Distributions - Cumulative Workload Segments

The Test Operator shall report the Cumulative Workload IO Streams Distributions from the selected Real World Workload. Note that each test step will have a unique combination of IO Streams different from the Cumulative Workload (overall average IO Streams). The Cumulative Workload IO Stream distribution, shown below, is used because it is impractical to list the IO Stream combinations for each of the numerous, and unique, Replay test steps.

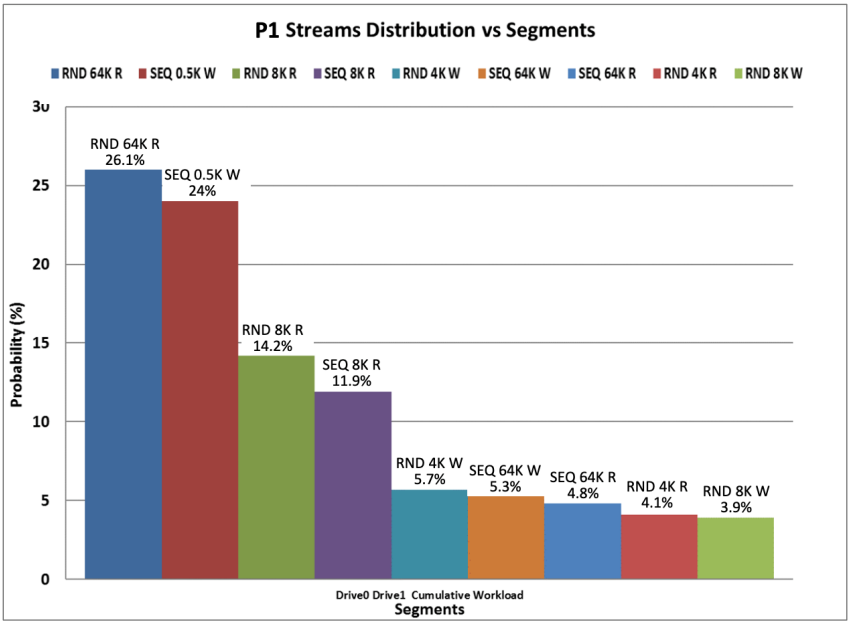


Figure 8-2. Workload Streams Distribution – Cumulative Workload Segments

P2 - IO Streams Map by Quantity of IOs & IOPS

IO Streams Map by Quantity of IOs shows IO Streams and IOs (colored bars) and IOPS (blue dots) as they occur for each time-step.

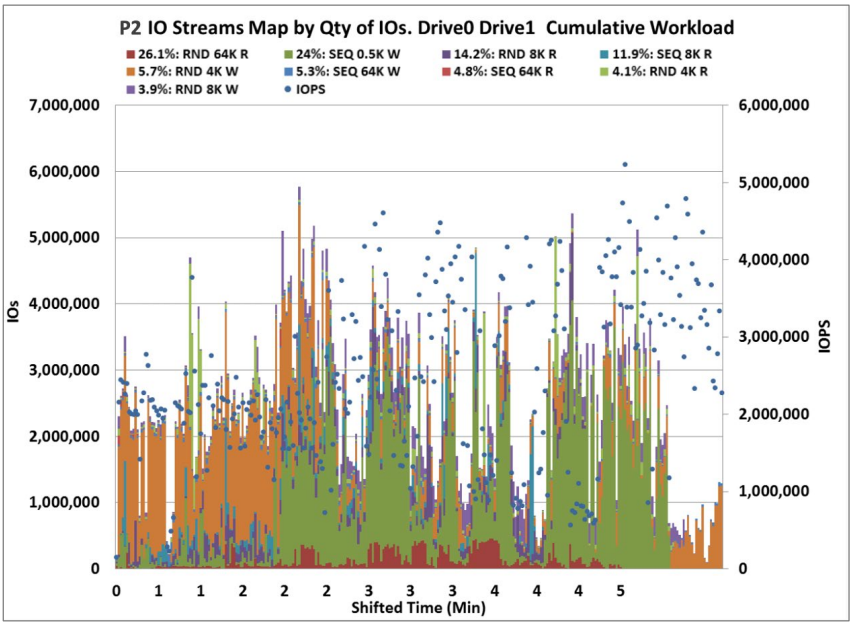


Figure 8-3. IO Streams Map by Quantity of IOs & IOPS – 2 Drive Retail Web Portal

P3 - Probability of IO Streams by Quantity of IOs

Probability of IO Streams by Quantity of IOs shows each IO Stream and its probability of IO Stream occurrence by number of IOs over the duration of the IO Capture.

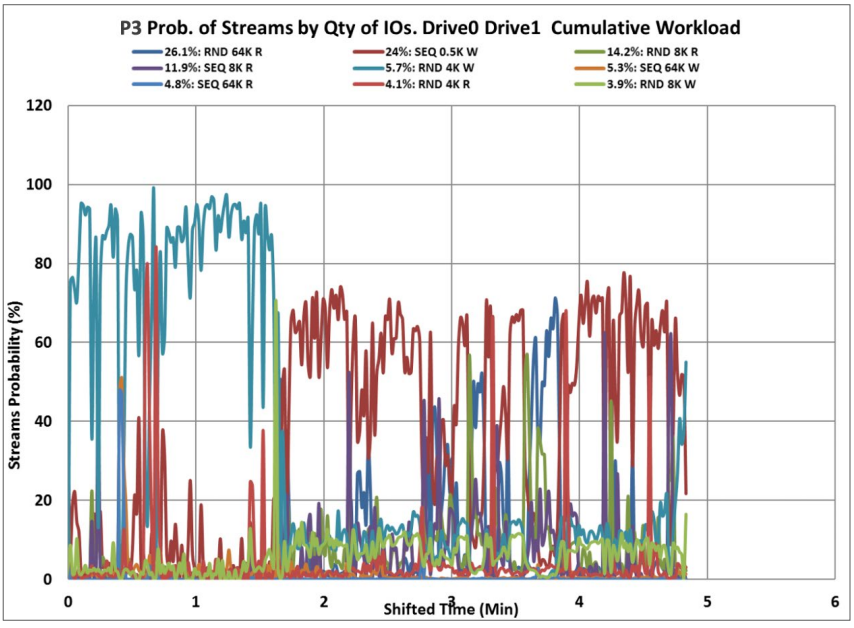


Figure 8-4. Probability of IO Streams by Quantity of IOs

P4 - IOPS v Time

IOPS v Time shows the cumulative IOPS for each data point of the IO Capture.

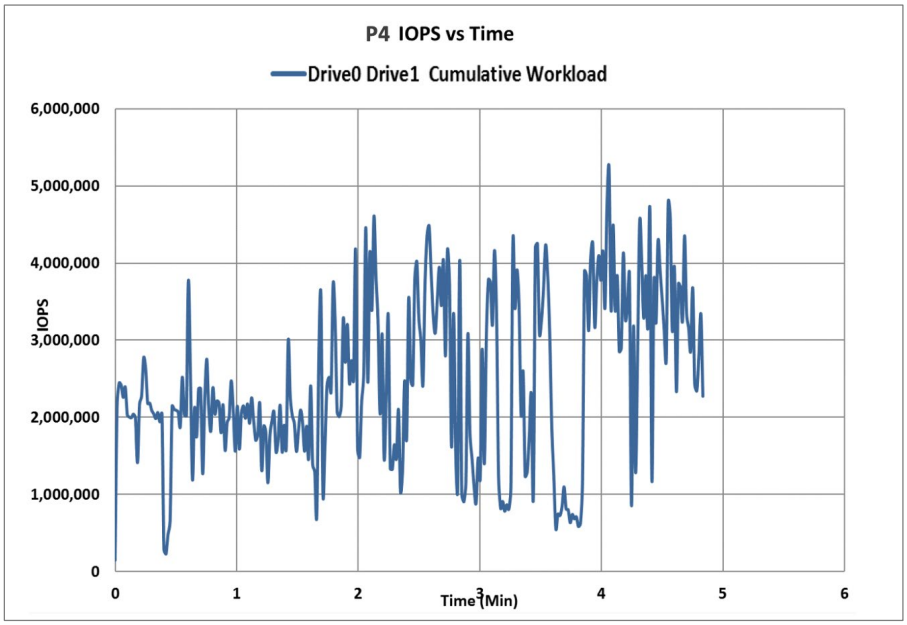


Figure 8-5. IOPS v Time

P5 – Throughput v Time

Throughput (MB/)s v Time shows the cumulative MB/s for each data point of the IO Capture.

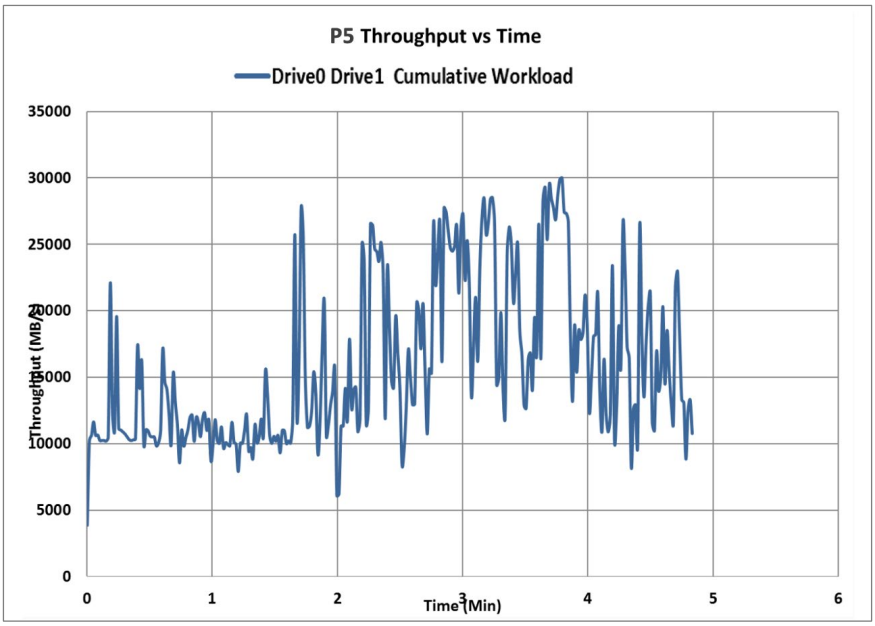


Figure 8-6. Throughput v Time

P6 - Latency v Time

Latency v Time shows the cumulative response times for each data point of the IO Capture. Figure 8-7 shows Average Response Times (blue), 5 9s Response Time Quality of Service (red) and Maximum Response Times (green).

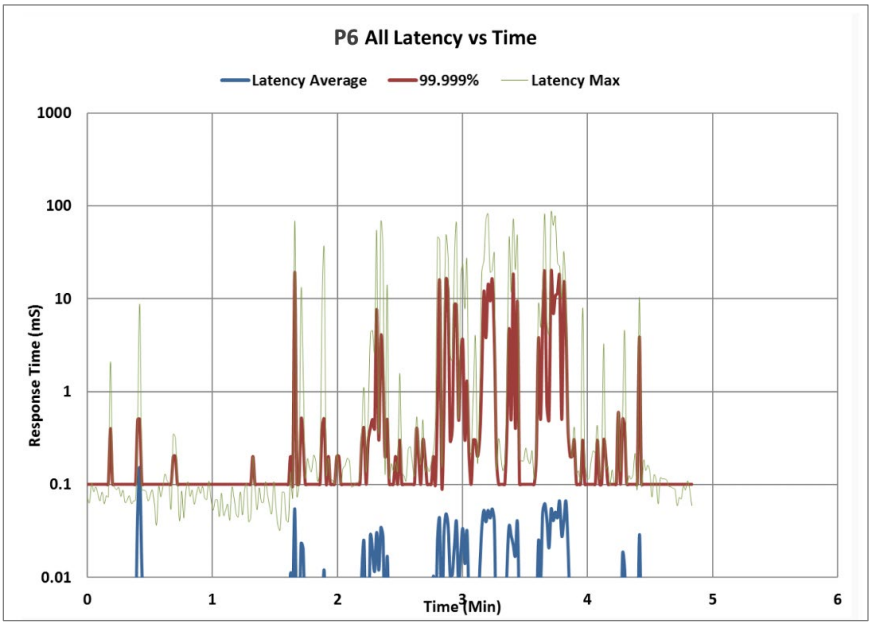


Figure 8-7. Latency v Time

P7 – Throughput & TC v Time

Throughput (MB/s) & TC v Time shows MB/s (blue) vs TC (red) for each data point.

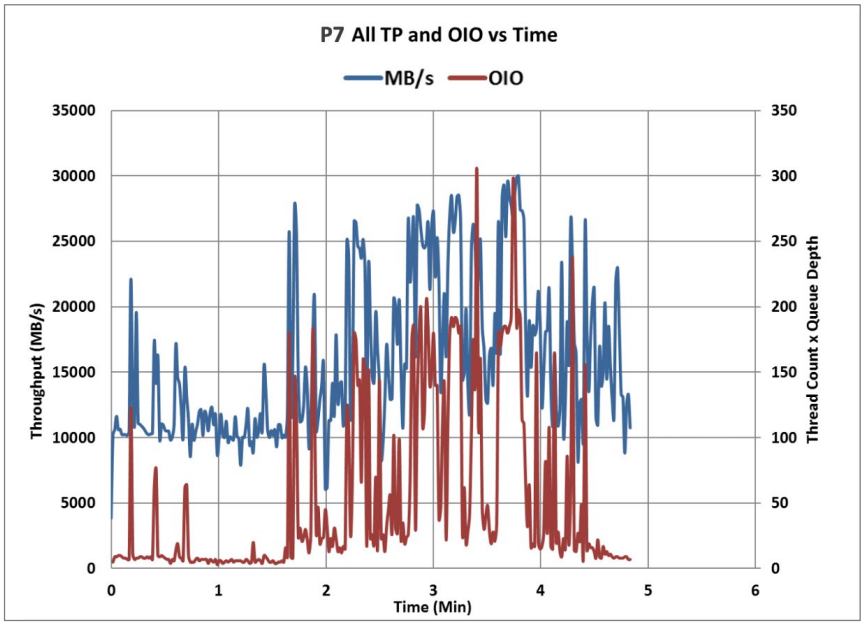


Figure 8-8. Throughput & TC v Time

P8 - IOPS & Response Times v Segments (Replay & Individual Streams)

IOPS and Average Response Times are shown for each segment (Replay segment Drive0Drive1 and Individual Streams segments).

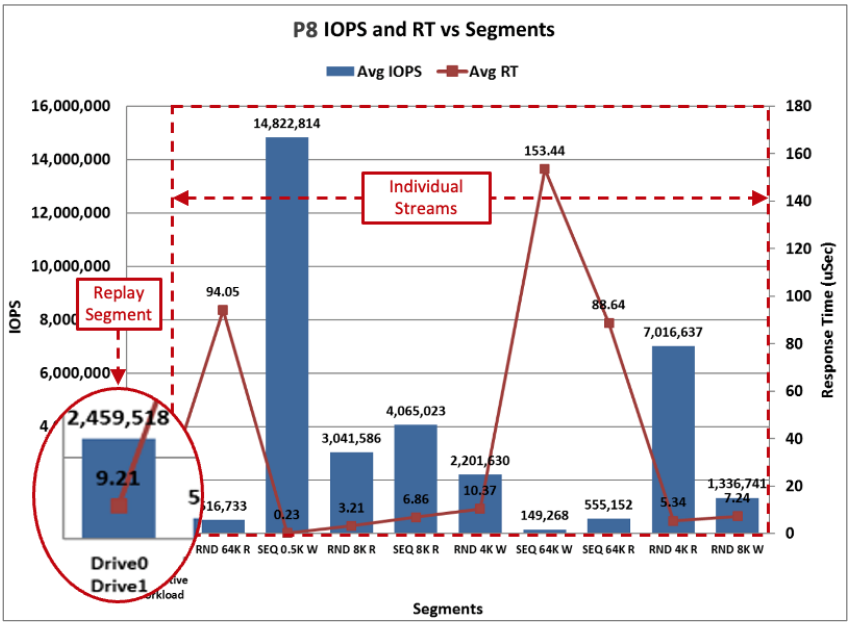


Figure 8-9. IOPS & Response Times v Segments (Replay & Individual Streams)

P9 – Throughput & Response Times v Segments (Replay & Individual Streams)

Throughput (MB/s) and Average Response Times are shown for each segment (Replay segment Drive0Drive1 and Individual Streams segments).

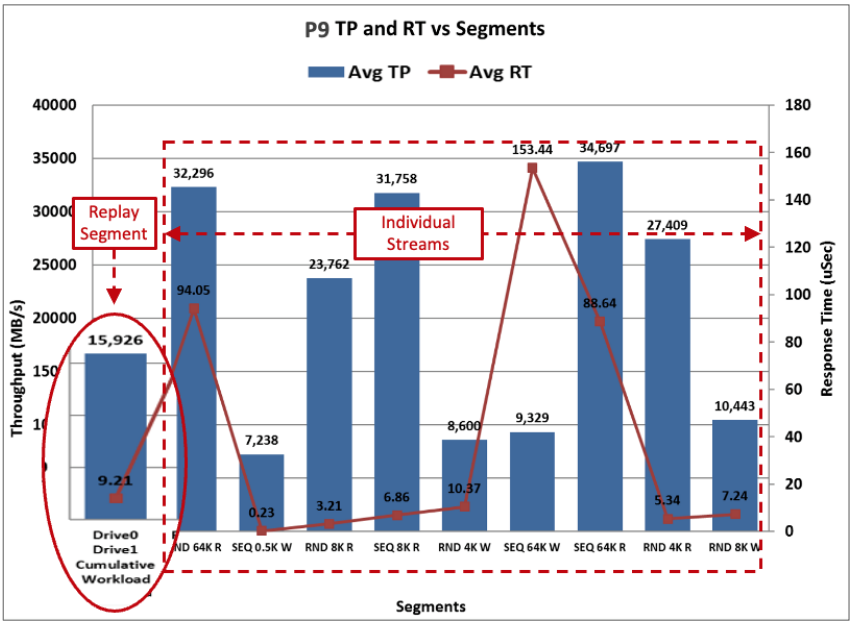


Figure 8-10. Throughput & Response Times v Segments (Replay & Individual Streams)

P10 – Replay Segment - Average IOPS

Replay Segment Average IOPS shows the average values over the duration of the Replay test.

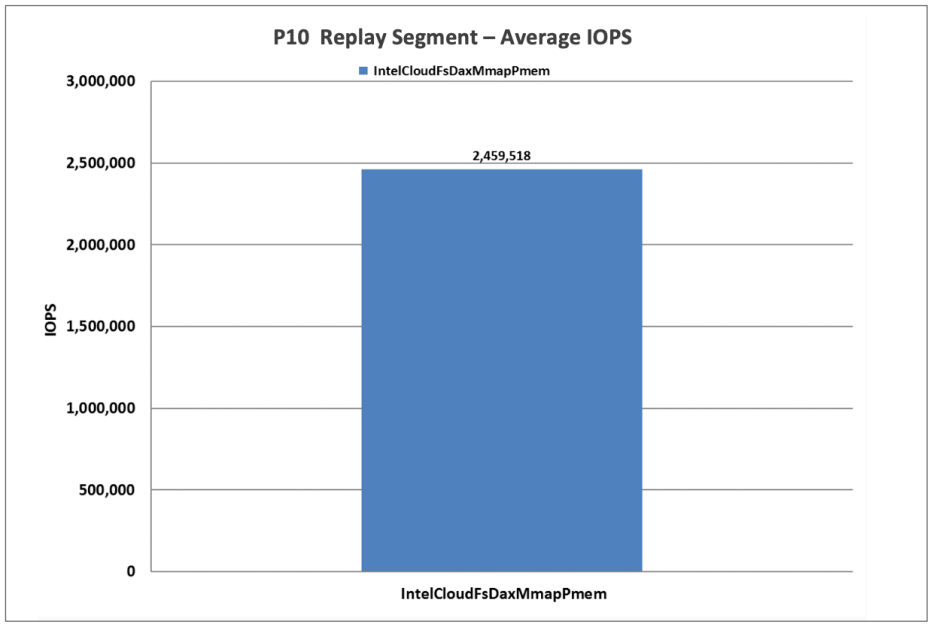


Figure 8-11. Relay Segment - Average IOPS

P11 – Replay Segment – Average Throughput

Replay Segment Average Throughput (MB/s) shows the average values over the duration of the Replay test.

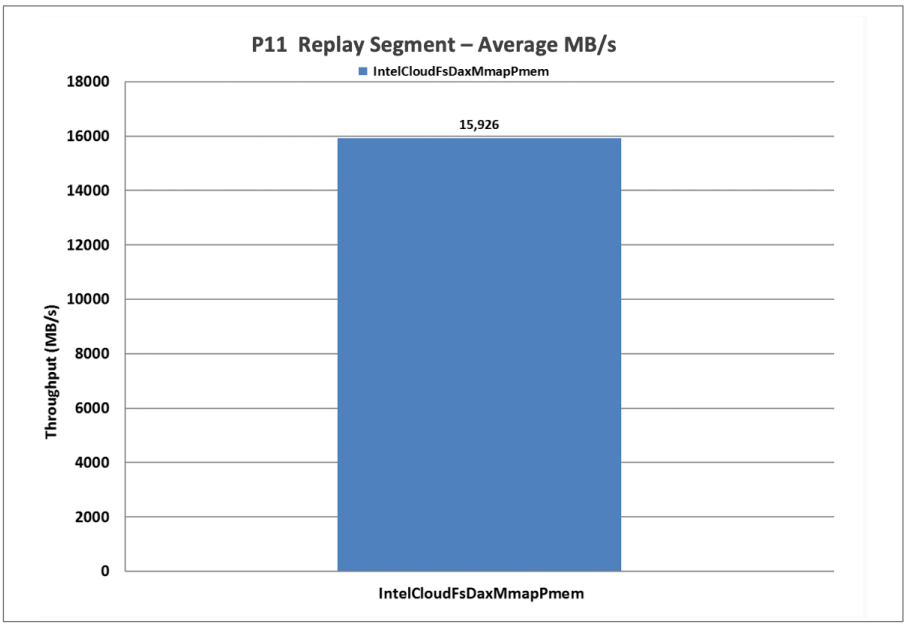


Figure 8-12. Replay Segment - Average Throughput

P12 – Replay Segment – Average Response Time

Replay Segment Average Response Time shows the average response time values over the duration of the Replay test.

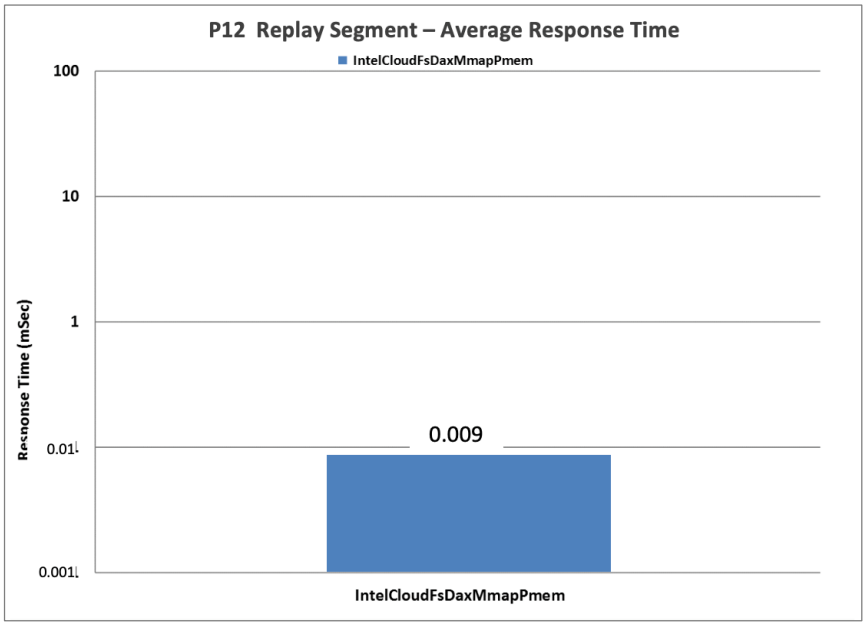


Figure 8-13. Replay Segment – Average Response Time

P13 – Replay Segment - Maximum Response Time

Replay Segment Maximum Response Time shows the maximum response time value over the duration of the Replay test.

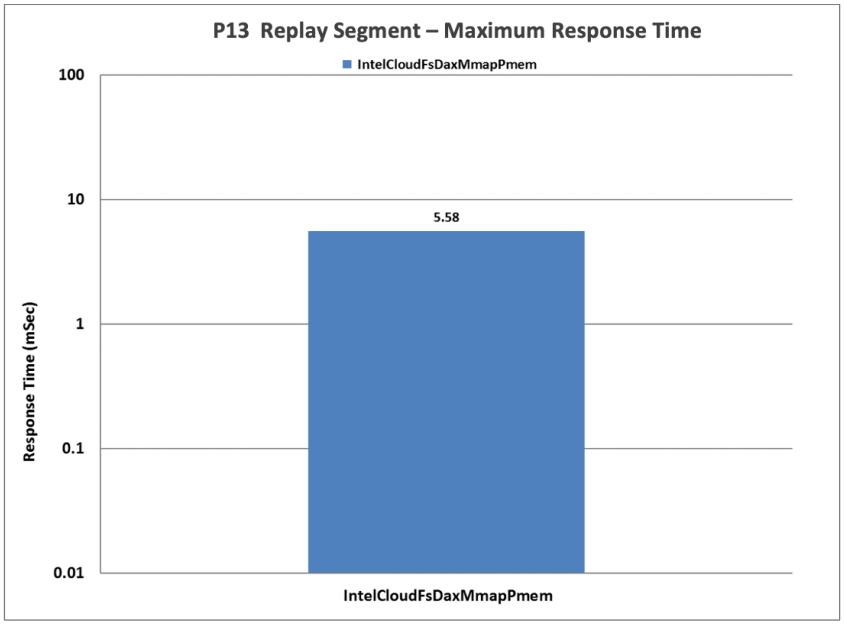


Figure 8-14. Replay Segment - Maximum Response Time

P14 – Replay Segment – 5 9s Response Time Quality of Service

Replay Segment 5 9s Response Time Quality of Service shows the 5 9s response time values over the duration of the Replay test.

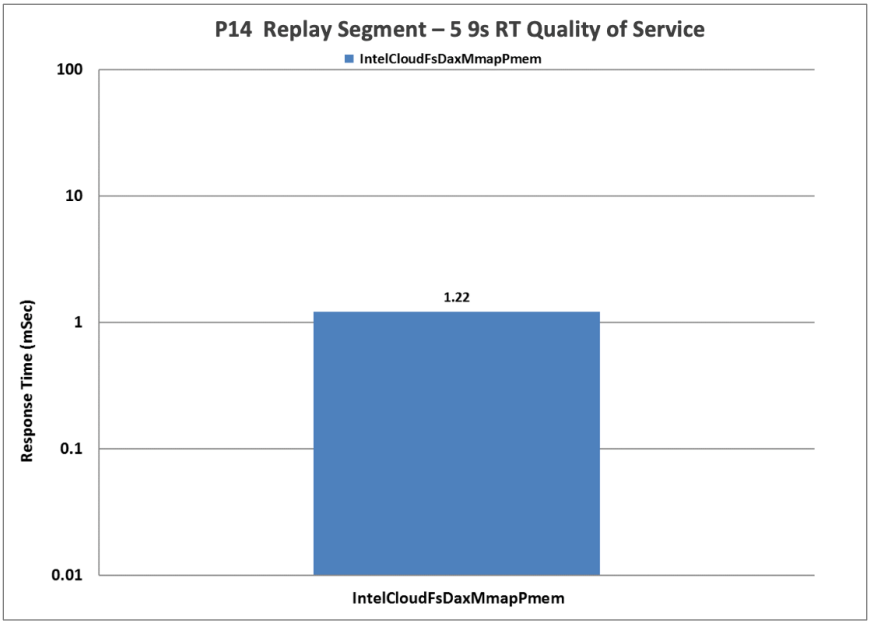


Figure 8-15. Replay Segment – 5 9s RT Quality of Service

Annex A Sample PM PTS Test Report Header

This annex displays a sample PM PTS version 1.0 SNIA Test Report Header. Individual Report Pages shall contain mandatory Report Headers on each page that set forth required reporting information pertinent to the tests presented.

A. Sample PM PTS Test Report & Header

Figure A-1 below shows a sample P3 All Throughput v Time report and header for the Individual Streams Test using a RND 64b Read workload.

Test Run Date:		06/27/22 03:16:32 PM		Report Run Date:		08/19/22 12:39:49 PM	
Demand Intensity Response Time Histogram - Thread Count/Queue Depth Sweep (REQUIRED) - Report Page							
SNIA SSS TWG	Persistent Memory Performance Test Spec (PM PTS)	Individual Streams Test - RND 64B Read				Rev.	PM PTS 1.0
						Page	3 of 14
Vendor:	ABC Co.	PM Storage Model:	PMEM ABC123		TEST SPONSOR	XYZ Test Co.	
PM Server Settings		PMEM Settings		IO Engine		Test Results	
Mfg Motherboard	Intel Wolfepass	NUMA	ON	block IO sync	Off	Access Pattern	RND 64B R
CPU type & No.	2.4 Ghz XEON 8260 x 2	Page Size Memory	2 MB	memcopy_nosync	Off	Thread Count Range	1,2,4,8,16,32,64,128
CPU Cores & Threads	24 (48 Threads) / CPU	PMEM Modules	6 x 256GB	memcopy_Ossync	ON	Warm-up Duration	30 sec
CPU Clock Speed	2.4 Ghz	Interleaving	ON	memcopy_CLWB	Off	Warm-up TC	T=16
System RAM	96 GB ECC DDR4	Region Config	FSDax	memcopy_Non Temp W	Off	Warm-up MB/s	600 MB/s
PM Driver	Linux 5.4.0-109	File Sys Config	6	Workload		Max MB/s TC	T=8
Operating System		IO Access Mode		Individual Stream	RND 64B R	Max MB/s 5 TP	945 MB/s
OS	Linux Ubuntu 20.04.4	Sector	OFF	Composite Stream	Off	Max MB/s 5 9s QoS	9 uSec
Kernel	LTS kernel 5.4.0-109	FSDax	ON	Real World Replay	Off	Max MB/s ART	.45 uSec
Driver	Linux 5.4.0-109	DevDax	OFF	Cumulative 9 IO Stream	Off	Max MB/s MRT	1499.29 uSec

All Throughput vs Time

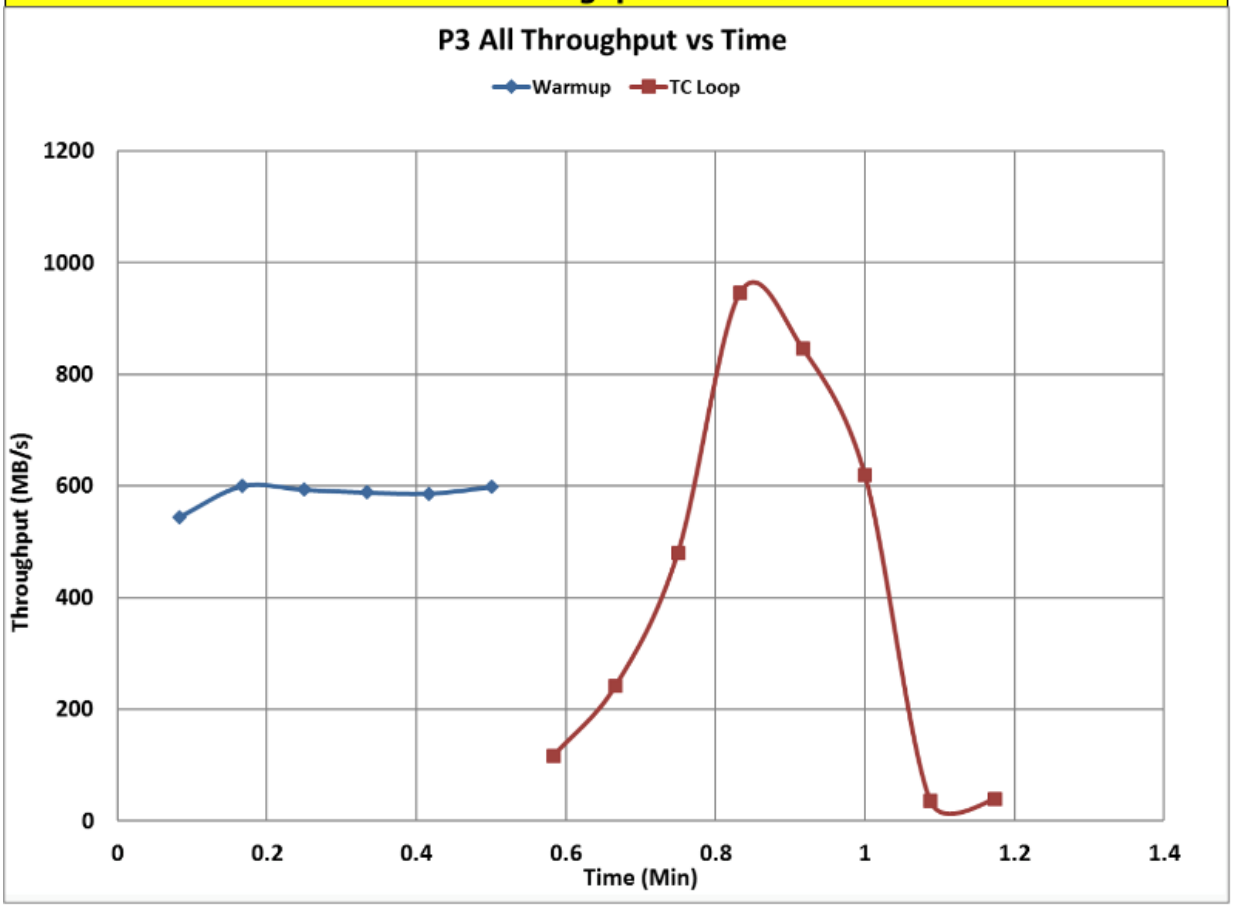


Figure A-1. Sample PM PTS Report & Header – Individual Streams Test RND 64B Read

Annex B (Informative) Reference Test Platform Example

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

The RTP is not required to run the PM PTS tests; it is an example of a platform that was used to develop the PTS and can be used to test PM storage to the PM PTS. Any one of the below listed commercial servers may be used as a PM PTS. Any listed commercial PM server should allow the user to achieve performance results similar to those examples listed in this PM PTS.

Any of the following commercial PM servers with associated OS, drivers and software, may be used as a PM RTP. The test operator shall disclose, among other items, which of the below servers is being used as an RTP, and other performance related software and components that are in the PM data path. Commercial PM servers are listed in Table 1 while performance related software and components are listed in Table 2 below.

Use of an RTP will allow the user to run the tests and procedures set forth in this PM PTS without significant bottleneck, or limitation, at a minimum acceptable level of performance. While performance related software and components in the data path, and/or their equivalents, should ensure a minimum level of performance, higher performance may be achieved with similar, but higher performance equivalents (such as CPUs with a higher number of CPU cores, faster CPU clock speed, larger DRAM amount, faster speed DRAM, operating system, kernel version, PM driver revision, test software, etc.).

The RTP listing is updated from time to time by the CMSI. The most recent RTP listing can be viewed at http://www.snia.org/forums/sssi/rtp where PM RTP are listed for PM 3D Cross Point and PM NVDIMM-N/P.

The PM RTP/CTS is used to test PM to the PTS and publish results, from time to time, on the SNIA CMSI website http://www.snia.org/forums/cmsi. Other Operating Systems (e.g., Windows, FreeBSD, etc.), test hardware and software can be used but results may differ from the PM RTP/CTS. Use of different OS, test software or hardware should be disclosed with any published PTS test results.

B-1. Commercial PM RTP Servers

The PM PTS is intended to be run on a commercial PM RTP server as listed below in Table 1. Key settings and set-up configurations that may impact performance are listed in Table 2. Test operators are encouraged to use the below, or equivalent, RTP hardware and software. Regardless of the RTP and/or hardware/software selected, test operators shall disclose all OS, kernel versions, bios, PM drivers, firmware revs, test software, and test settings used when reporting PM test results to this PM PTS.

Manufacturer	Server Name	Comment
Dell	Dell R740	2u server; 2 Socket CPU
Supermicro	Max IO	2u server; 2 Socket CPU
HPE	DL 380/385	2u server; 2 Socket CPU

Table *1 – Commercial PM RTP Servers*

B-2. Performance Related Software & Components

The PM RTP is intended to be run on a commercial PM RTP server as listed Table 1 above. Performance related software and components as listed in Table 2 below are required to be used with the commercial PM RTP server and shall be disclosed in the test reporting. The user shall list and disclose the components listed in Table 2.

Software /Component	Disclosed Components
Processor	
Main Memory	
Operating System	
kernel (Linux)	
BIOS version	
PM Firmware	
Memory Channels	
Memory Modules	
Test Software	
Set Up	See Section 5
Test settings	See Section 6,7,8

Table *2 – Performance Related Software & Components*