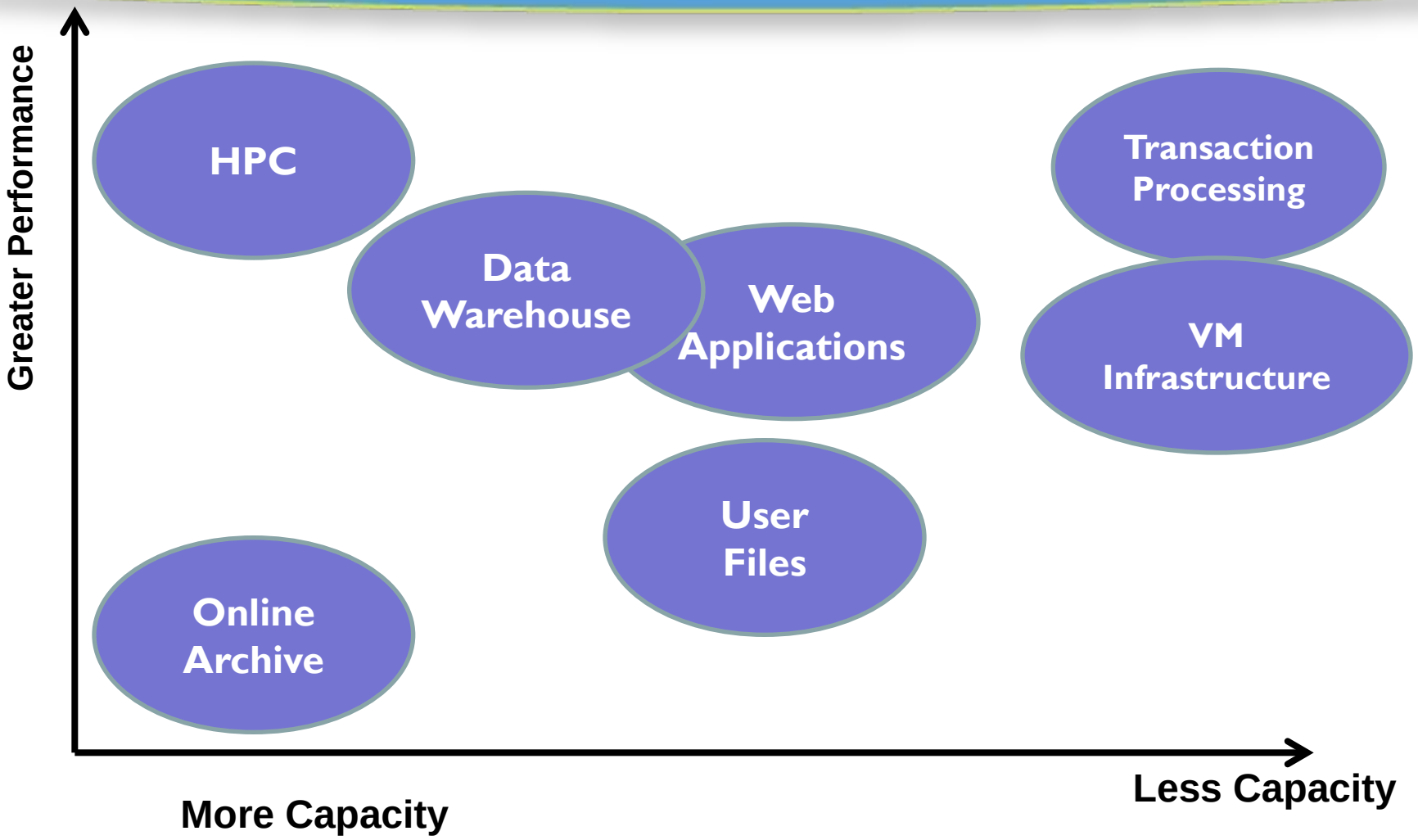
Considerations to Accurately Measure Solid State Storage Systems

Leah Schoeb

Storage Solutions and Performance Manager

Intel

Landscape for Solid State Storage



Advanced AFAs are a Different Animal

- Flash behavior is unique
- AFAs have a different performance curve
- Advanced AFAs do not merely store data
  - Most perform extensive metadata processing
    - Deduplication
    - Compression
    - Elimination of repeating character strings
- These new arrays require a new performance testing methodology

3

Considerations for Solid State Storage Arrays

Data Services Management	Investment Protection	Support
Data Reduction - Deduplication - Compression - Thin Provisioning	**Self-healing techniques (Reliability)**	**Hypervisor** - VMware vSphere - MS Hyper V
Replication - Local (writable) - Remote (Future)	**Hardware Redundancy (Availability)**	**Scale out and Clustering**
Management - Non-disruptive upgrades - REST APIs	**Serviceability**	**Application & OS**

Measuring Accurate Performance w/ All Flash Arrays

- **Problem**
  - Traditional IO generation tools don't work
- **Flash as a unique behavior**
  - Not a hard disk drive
- **Built-in data services**
  - Inline data reduction technologies
- **Different Performance curve**
  - Flash arrays measure differently than traditional systems

Inadequate Tool Sets

- Measuring new technology based on old assumptions – **Don't Do It!**
- Result – **Inflated performance results**, inaccurate measurements
- **Negatively Impacts Everyone**
- Setting accurate expectations

Modern All Flash Arrays vs. HDD Arrays

Modern Flash Arrays

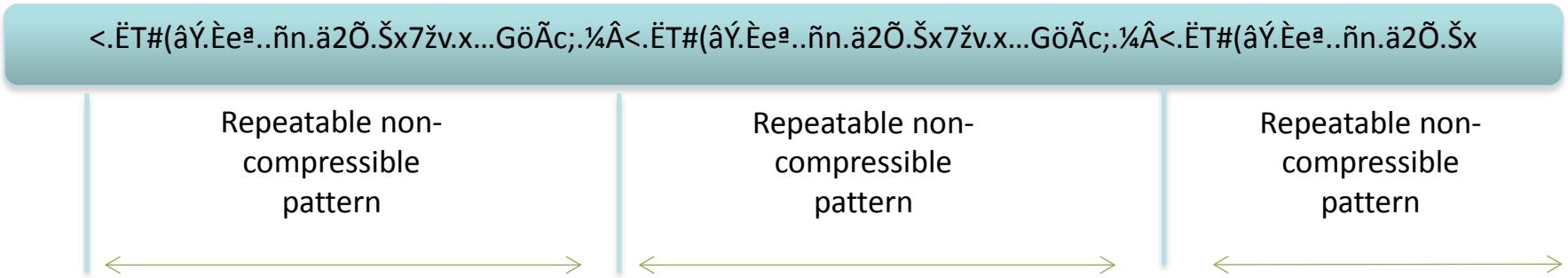
- Wear Leveling
- Garbage collection
- Metadata Management
- Self-healing techniques
- Inline Data deduplication
- Inline Compression

Traditional HDD Arrays

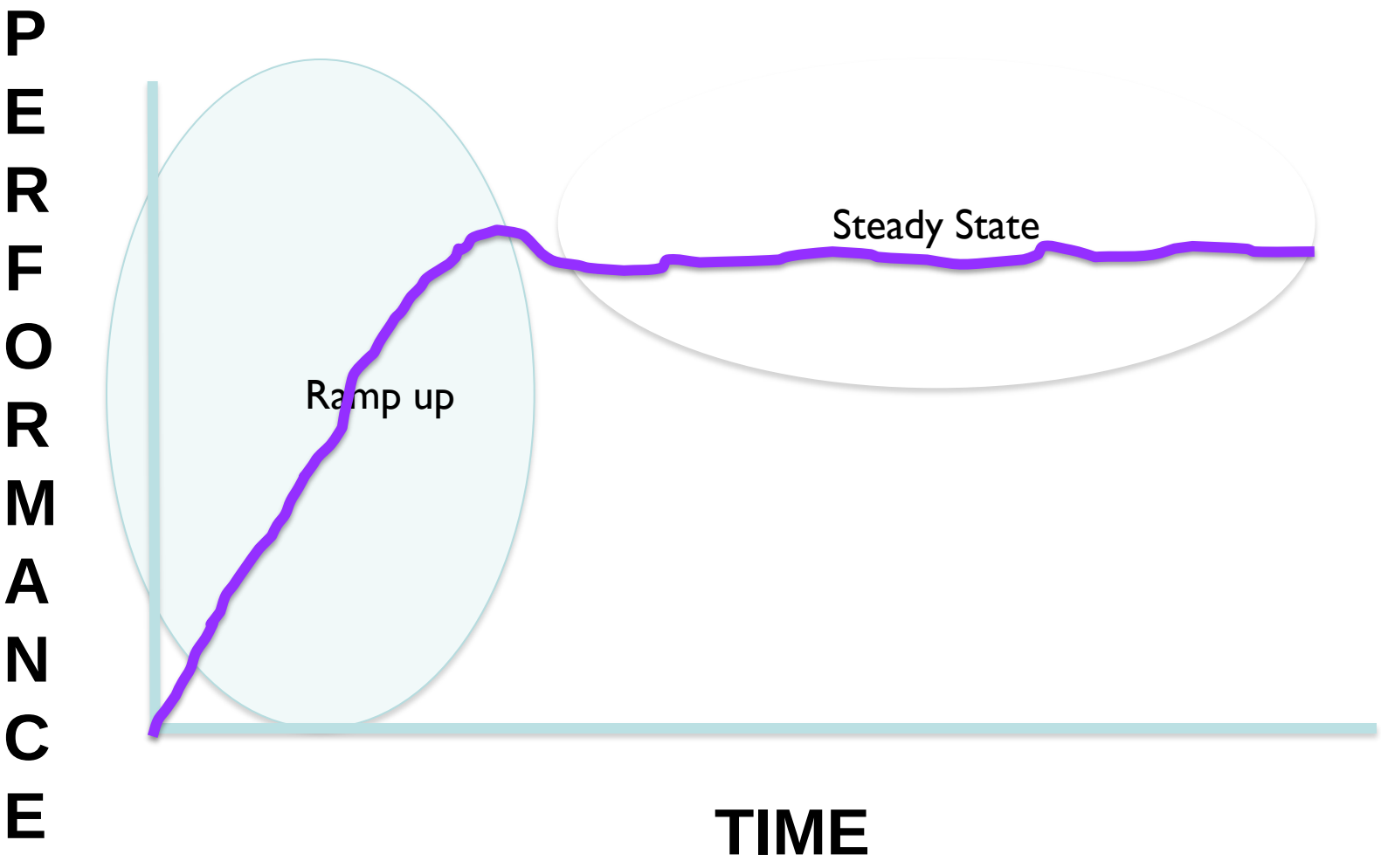
- Rotational Latency
- Seek Times
- Mechanical parts
- Controllers designed to handle HDD

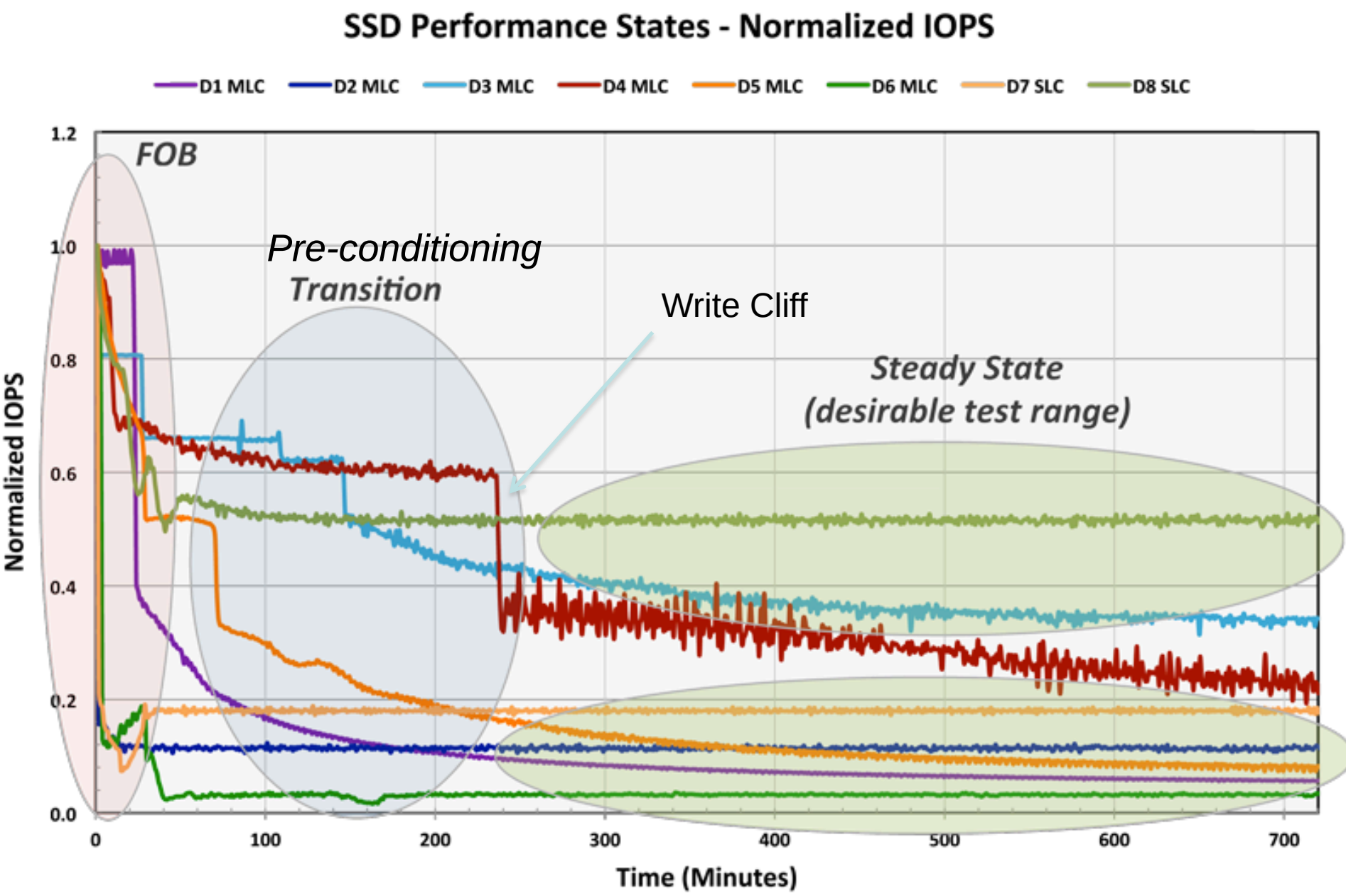
Measuring w/ inline Data Reduction

- Data content patterns and data streams
  - Patterns written to disk as part of pre-conditioning
  - Patterns presented to an array during steady state
- Repeating and Non-Repeating
  - random patterns
  - Compressible patterns
- Varying pattern lengths
- Most IO generators are inadequate



Traditional Disk Performance Curve





(SNIA SSSI Specification)

SNIA DSI DATA STORAGE INNOVATION CONFERENCE

Methodology Overview

SNIA™

Methodology Elements

- Pre-conditioning
- Creating a realistic data set
- Writing to create an application data set
- Writing to exercise the array emulating an appropriate workload
- Other tests to emulate realistic, simultaneous writing and reading

12

Pre-Conditioning

- Involves breaking in entire flash array
  - Writing to every cell to achieve steady state
  - Helps to ensure garbage collection during main test cycles
- Goal: create a realistic data set
  - Dedupeable and non-dedupeable blocks
  - Compressible and non-compressible blocks
  - Combined using varying block sizes
  - Written to emulate hot spots and drift
  - Written with appropriate dedupe/compression ratios

13

Write Performance Tests

- Exercising array like an application does
  - Writing at high load to find limits
  - Writing using a data stream relevant to the data set
  - Writing to emulate long-term application access
- Goal: Exercising the array realistically
  - Using a variation of the pre-conditioning data set
  - Writing with same levels of data reduction
  - Using multiple block sizes
  - Including hot spots and drift to emulate temporality

Read/Write Workload Tests Scenarios

- Tests that write and read simultaneously
  - All-write tests do not exercise an array the way an operating application does
  - Reading must be combined with writing for realism
    - Tests using all-write data patterns, but reading also
  - Run at expected application load
- What if testing to determine capacity
  - Magnifying the load to test future expected loads

15

SNIA DSI DATA STORAGE INNOVATION CONFERENCE

Methodology Components

SNIA™

Block Size

- Block sizes vary by application and operation
  - 25K-35K average size is common
  - However, no application uses uniform block sizes
  - Sizes vary according to operations
- OLTP transactions typically small
- Analytics, reporting typically larger
- AFA methodology should reflect real access
  - Single application
  - IO Blender (multiple applications)
  - Either model requires multiple block sizes
- Should reflect application/blender access
  - E.g. 3% 4K, 15% 8K, 20% 16K, 52% 32K, 10% 64K

Hot Spots / Hot Bands and Drift

- Application access is not uniformly random
  - Hot spots are storage locations accessed more frequently than others
  - Hot spot regions change over time
  - Called drift
    - E.g. Index file growth as transactions are processed
- Hot Spot examples:
  - Index Files
  - Temp Files
  - Logs
  - Journals

18

Hot Spots/Bands and Drift (continued)

- Hot spot emulation example:
  - 1% of all access regions receive 35% of the IOs
  - 1.5% of all access regions receive 15% of the IOs
  - 2.5% of all access regions receive 15% of the IOs
  - 5% of all access regions receive 15% of the IOs
  - 7% of all access regions receive 10% of the IOs
  - 6% of all access regions receive 5% of the IOs
  - 7% of all access regions receive 3% of the IOs
  - 5% of all access regions receive 1% of the IOs
  - 65% of all access regions receive 1% of the IOs

Access Patterns

- Tests must reflect realistic access patterns
  - Should emulate real applications
  - Should avoid uniform random write distribution
  - Should use multiple block sizes
    - Both result in unrealistic access patterns that skew towards systems that maintain larger amounts of reserve flash memory
- Methodology should include testing in the presence of:
  - Backups
  - Snapshots
  - Replication

20

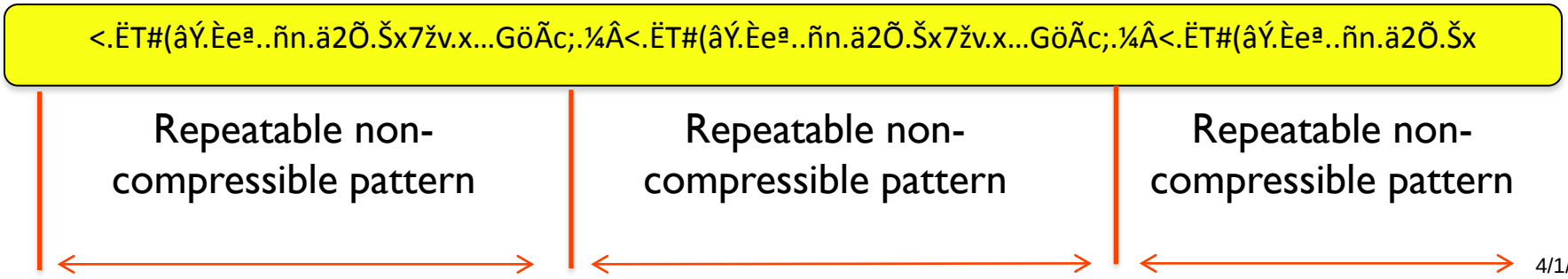
Complex Data Patterns

- Complex data patterns model workloads
- Pattern types:
  - Unique
  - Repeating
  - Uncompressible
  - Compressible
- Combined to represent data content representing:
  - Data set at rest after pre-conditioning
  - Data patterns that emulate traffic during operation

21

Data Content

- Data content patterns
  - Created before testing
- Data content streams
  - Written during testing
- Repeating and non-repeating patterns
  - Random
  - Compressible
- Varying pattern lengths



Thread Count and Queue Depth

- Helps find max performance for each:
  - Thread count
  - Queue depth
- Tests must find max IOPs an array can do per:
  - Thread count (workers)
  - Queue depth (outstanding I/Os)
  - Combination of threads and queue depth
- Increasing thread count past current requirements shows how array meets future needs

23

New SNIA Technical Working Group

Solid State Storage System Technical Working Group

(s4twg.snia.com)

Solid State Storage Systems TWG

- Solid State Storage Systems Technical Work Group (S4 TWG)
- Why another solid state technical workgroup?
  - How can we set correct expectations?
  - What modern tool set can be used for measuring performance?
  - How can vendors and test sponsors that provide inaccurate results be held accountable?

Why another Solid State TWG?

- Address the unique performance behavior of SSS storage systems
- Inline-advanced features
- Measuring Performance of enterprise arrays vs. devices
- System wide housekeeping vs devices level
- Caching and DRAM tiering

S4 Formulation

- Approved – August 2014
- First teleconference – September 2014
- First Face-to-Face – November 2014
- Current # Members - 24 members

Charter

- **Identify, develop, and coordinate** systems standards to **enable accurate performance** measurement for solid state storage systems
- Produce a **comprehensive set of specifications** and drives consistency of **measurement guidelines** and messages related to solid state storage systems
- Documents **system-level requirements** and shares these with other performance standards organizations

Program of Work

- **Develop a specification** - for measuring the performance of solid state systems.
- Includes **support for inline advanced storage features** that directly impact performance and the long term behavior of the array.
- **Note: This will build upon process methodology developed by the SSS TWG**

Current focus areas

- Targeted Workload Modeling
  - Database
  - Server virtualization and VDI
- Characteristics
  - Access Patterns
  - File access (structure, location, metadata, etc.)
  - Data Reduction technologies (inline & post)
  - Caching affects and SRAM tiering
  - System wide vs device level housekeeping

Areas of focus

- Defining a SSS System / SSS Taxonomy
- **Use cases**
  - **workload analysis**
  - **Stimulus concepts**
  - **Model virtualized environments (use as part of the stimulus?)**
  - **Including filesystems (server side filesystem activity)**
- Should we include server side storage
- Power (power budgets, Green TWG, SSS)
- SMI-S integration points

Summary

- All-Flash Arrays are unlike disk-based arrays
- Data reduction dramatically changes performance characteristics
- Tests must include rich data content to be valid
- Tests must model real-world access patterns
- Testing must be fair, unbiased and repeatable
  - "One size fits all" may not be fair to tiered arrays

32

Summary (con'd)

- Tiered arrays are unlike all-flash arrays
  - This methodology valid for arrays that implement data reduction
  - but may not be appropriate for tiered arrays
  - A second methodology may be required,
  - especially for tiered arrays that do implement data reduction

References

- www.evaluatorgroup.com - “Measuring Performance of Solid State Arrays”
- www.loaddynamix.com – “Go Daddy White Paper: Storage Validation”