Getting it Right:
Testing Storage Arrays The Way They’ll be Used

Peter Murray
Virtual Instruments
The Journey: How Did we Get Here?

- Storage testing was black art
- Test programs were derived from disk drive utilities
- Did not represent actual applications
- Could not emulate temporal or spatial locality
- Did not emulate Data Content
- Difficult to emulate varying loads on many LUNs
- Difficult or impossible to configure the metadata and structure required to emulate file-based apps
How is Flash Different?

- Addressable storage space is likely less than raw space
  - Designed to help increase flash life
- Can help avoid performance issues during garbage collection
- Other methods are available to avoid performance issues
- Deduplication & compression decrease storage requirements for an app
  - More storage per nominal byte
  - But, performance may be impacted
- Advanced metadata processing & workload profiles at scale make it harder to saturate an array
- Test at near full capacity to understand array performance
- Testing with hotspots helps model application behavior
- Garbage collection or metadata processing may affect performance
- Software services & protocols – software runs differently on SSD than on HDD
SS Arrays Require New Storage Testing Methods

- Applications exhibit spatial and temporal locality
  - Modern solid state arrays are designed with this in mind
- Application traffic contains data content
  - Data is random or compressible
  - Data may also be de-dupable
  - All content types are present in most applications
- Application traffic is “bursty”!
  - Testing without bursts is unrealistic
- Some all solid-state storage arrays must be tested with locality and content
  - Data reduction is a key feature - can’t be turned off
  - Legacy testing apps cannot emulate the locality, content or content flocking present in applications
- New thinking and testing applications are mandatory!
Realistic Access Patterns

- Testing should reflect the access patterns of applications
  - No application uses entirely random or sequential access
  - No application consist of only writes

- Access pattern factors:
  - Write/read ratios
  - Random/sequential access ratios
  - Access pattern drift
  - Realistic block-size mix
  - Alternate paths

- Should test with enterprise feature sets
  - Backups, snapshots, replication, etc.
Access Patterns

- Application access is not uniformly random
- Hot spots are storage locations accessed more frequently than others during a defined time period
  - Index Files
  - Temp Files
  - Logs
  - Journals
- Testing should reflect Hot Spots and Hot Spot Skew
  - 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
- Testing should accurately emulate data offset, or “Drift”, over time
  - Note: The developer of fio has written that skew is even greater than the example above
Locality

- Locality is present in virtually all applications
- Storage arrays use locality to determine where and when to write data
- Locality defines:
  - Where data is written or read – spatial locality
  - When data is written or read – temporal locality
- Hot spots/hot bands represent locality
- Testing without locality does not stress an array as it will be in production
Block Sizes

- Block sizes vary by application and operation
  - 25K-35K average size is common
- Applications do not use uniform block sizes
  - Sizes vary according to operations
    - OLTP transactions typically small
    - Analytics, reporting typically larger
- Testing must include representative block sizes
  - Block sizes should be mixed to reflect applications
    - E.g. 3% 4K, 15% 8K, 20% 16K, 52% 32K, 10% 64K
For larger data centers Top Max Pending can average: **500-700**
Bursts: What do Real Reads Look Like?

If constant would mean: **40K to 80K IOPs**

The bursts are ~40 times as high as the average
Bursts: What do Real Writes Look Like?

Even a bigger difference
For writes
Bursts: What do Real Writes Look Like?

Even a bigger difference
For writes

50ms bursts > 1 minute
Average IOPs
Bursts: What LDX generates by default

One second of data
Running at 80% of Maximum
Testing Without Bursts
Bursts: Real World Issue
Bursts: The difference?

<table>
<thead>
<tr>
<th>Test</th>
<th>IOPs</th>
<th>Throughput</th>
<th>Latency</th>
<th>Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>With bursts</td>
<td>20K</td>
<td>1250 MB</td>
<td>6.5 ms</td>
<td>Real world</td>
</tr>
<tr>
<td>With bursts</td>
<td>25K</td>
<td>1569 MB</td>
<td>80 ms</td>
<td>Unacceptable latency</td>
</tr>
<tr>
<td>Without bursts</td>
<td>25K</td>
<td>1566 MB</td>
<td>1.2 ms</td>
<td>Lab-Myth</td>
</tr>
</tbody>
</table>
Data Content

- Modern Storage arrays use data reduction
- Data reduction saves array space
- Consists of:
  - Deduplication
  - Compression
  - Pattern reduction
- Data content patterns are a must for testing data reduction
Measuring Data Reduction

- 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

- Thread counts and queue depth
  - Tests should include increasing thread counts to find maximums for each test case
  - Should include increasing queue depth to find maximums for each test case

- Find:
  - Max IOPs an array can do per thread/queue depth, and
  - Total for a given number of threads and queue depth

- Increase thread count past current requirements to show how array meets future needs
Methodology In Action
Actual results comparing 2 leading AFAs

IOPS Comparison for 3 Groups of Data Patterns & R/W Ratios

Which is best?
Depends on your workload.

Vendor A
Vendor B

Read/Write Ratios

0 50000 100000 150000 200000 250000 300000
20% /50% /80% / 20% 20% /50% /80% / 20% 20% /50% /80% / 20%

20% Reducible
50% Reducible
80% Reducible
Typical Performance Testing Questions

- Which is the best technology for my needs?
- Which is the best vendor / product for my needs?
- What is the optimal configuration for my array?
- Does performance degrade with enterprise features: Deduplication? Compression? Snapshots, Clones, Replication?
- What are the performance limits of a potential configuration?
- How does an array behave when it reaches its performance limit?
- Does performance degrade over time?
- Which workloads are best for an AFA? A hybrid storage array?
Traditional Storage Testing Approaches

- Limits finding
- Functional testing
- Error Injection
- Soak testing
Workload Modeling

**Performance Profiling**
Fully characterize performance of arrays under wide variety of load parameters

**Workload Modeling**
Simulate the I/O profiles of your production environment
Workload Modeling

Performance Comparison: NAS Vendor A vs. NAS Vendor B

Vendor A: Shallow Tree Structure (2014-03-21: 11:05:05 AM)
Vendor B: Shallow Tree Structure (2014-03-20: 12:48:12 AM)
Vendor B: Deep Tree Structure (2014-03-20: 12:45:36 AM)
Performance Profiling

- Performance Profiling
  - Fully characterize performance of arrays under wide variety of load parameters

Workload Modeling
- Simulate the I/O profiles of your production environment

Change Validation
- Effect of HW and SW changes

Pre-Production Staging Validation
- Hot staging and burn-in

Configuration Optimization
- Tiering, caching, HDD/SSD mix, ...

Product Evaluation
- Best product for your workloads

Technology Evaluation
- Flash, NFSv4, FCoE, OpenStack, Ceph, …
Benefits of Realistic Testing

- Performance assurance
- Reduced storage costs
- Increased uptime
- Acceleration of new application deployments
Summary

- Application Testing is now mandatory
- Black art has become repeatable
- No synthetic workload is perfect
- But is the best approach available
- This will only improve over time
- Customers can see:
  - How closely the model emulates apps
  - A realistic view of how an array operates
- This new model is changing storage testing