Test Validation

Evaluation of Application based Intelligent Caching

Author: Russ Fellows

July 28, 2015

Evaluator Group

Enabling you to make the best technology decisions

This Page Left Intentionally Blank.

Table of Contents

Executive Summary

Ensuring that critical applications perform well is a significant consideration for IT organizations of all sizes. In choosing the best approach for application optimization, there are several considerations. A key aspect is the cost effectiveness of the solution, as well as ease of implementation and use. The ability to deploy new technologies with minimal disruption to existing systems and procedures is also a critical factor, and is helpful in enabling rapid deployment.

Some technologies provide improvements, but do so at the cost of disruption to the IT infrastructure. Ideally, products that provide measurable financial benefits with a minimal amount of risk or disruption are those that find the greatest success within companies.

Until recently, solid-state storage systems had not been adopted rapidly, due in part to the high cost along with the potential for disruptions. With improving economics of solid-state, many organizations are now re-evaluating their storage needs and exploring ways to either increase their existing use of solid-state or deploy these technologies for the first time.

However, concerns over the lack of enterprise features and high costs persist. For some environments and applications, it is possible to add a dedicated, all flash external storage system. In other instances, organizations do not want to incur the disruption that occurs when deploying a new storage, but still need to address performance for specific applications. In still other cases, application performance may become critical, and there isn't sufficient time or budget to purchase a new storage system.

Evaluator Group was asked to analyze the effects of deploying the PrimaryIO APA product. This software utilizes host-based solid-state media to deploy a read and write cache for specific applications. In this lab evaluation, Evaluator Group measured the performance of two database applications, first without PrimaryIO APA and then again after adding APA.

This validation report documents the tests utilized, the system configurations, and the performance results of using APA for two common applications, MySQL in a Linux environment and SQL Server running on a Windows 2012 server.

Evaluation Result Summary

The tests utilized were designed to represent application workloads utilizing two common database applications. Both MySQL on Linux and SQL Server on Windows systems were tested.

Some of the results include:

- Optimizing MySQL and SQL Server required no application downtime, after PrimaryIO APA had been installed

- PrimaryIO APA analyzed the system and provided recommendations for database objects, enabling application DBA's to specify what application components to accelerate
- For the workloads tested, PrimaryIO APA enhanced performance 4X (400% faster) for SQL server and 5X (500%) faster for MySQL

For details on the test cases, testing tools, hardware and software utilized refer to Appendix A.

Comparing Solid-State Acceleration Options

A majority of IT consumers prefer to purchase solid-state storage only when needed, and are typically trying to solve a particular issue. The most common issue is the need to accelerate a specific business critical application. This data comes from research Evaluator Group has conducted on IT purchasing preferences in discussions with our corporate IT clients.

Currently, three primary methods are available to enhance application performance by utilizing solid-state NAND Flash storage technologies: 1) Host based PCIe solid-state devices; 2) External solid-state storage systems, or 3) PCIe solid-state cards plus software intelligence.

Each method has benefits and shortcomings; each is typically optimal for a specific type of workload or application environment as outlined below in Table 1.

	Host PCIe SSD Only	PCIe as Cache	External SSD Storage
Acceleration	SSD is primary location of data, SSD adds capacity	SSD is an additional copy of data, SSD does not add capacity	Acceleration for all data that is migrated to reside on shared SSD.
Storage Features	Very limited features and limited HA capabilities.	Excellent, storage capabilities from existing storage.	Excellent features with robust HA capabilities.
Impact on Existing Infrastructure	Moderate to High. Existing workloads must be migrated to new storage.	Moderate, requires adding a physical card and software. No migration required.	Moderate to High. Existing workloads must be migrated to new storage.
Cost Effectiveness	Excellent performance for entire SSD Tier, limited capacity. Limited ROI.	Excellent cached data performance. Excellent capacity and high ROI.	Very good performance, Very good capacity and moderately high ROI.

Table 1: Comparing Storage SSD Options for Performance

How PrimaryIO APA Works

PrimaryIO APA is transparent to the application and OS, working at the block device driver layer. However, APA uses its application-specific level integration to intelligently map storage blocks to application objects. As a result, acceleration is targeted at a specific application, using PrimaryIO's application performance acceleration technology (APA) for application awareness and context.

The PrimaryIO APA is installed as an OS device driver, operating below any filesystem or application using the device. This enables caching either an entire LUN, or using application intelligence with specific APA modules. Management of PrimaryIO APA is performed using a web based graphical interface that enables the setup for caching policies, by application and device, which is used to establish the caching policies on a per instance basis.

Typical generic LUN-based caching solutions do not have the context to understand I/O beyond read or writes, and are unable to differentiate between an I/O request that is caused by an application or a some other process. The result is that I/O's from a database for backup or replication are also cached, resulting in less than optimal application performance. In contrast, APA can distinguish between different I/O streams, and is able to intelligently cache I/O from the application rather than secondary backup or replication processes.

PrimaryIO APA offers three alternative modes of caching:

- **Write-through cache** - Data is written to both the cache media and to the backing storage before the I/O is acknowledged. This enables accelerated reads without risking data loss, since non-volatility is not required in the cache layer. The underlying storage limits the write performance; as a result, read operations are accelerated but write performance is not accelerated.

- **Write-around cache** - This mode bypasses the cache for write I/O's, with data written directly to the backing storage. This may provide better performance for workloads with high writes. However, data in the read cache is not refreshed automatically, requiring the cache to issue a read to request to update data from backing storage. The result is no acceleration for writes and longer delays for reads than the other two modes.

- **Write-back cache -** In this mode, I/O goes directly into cache, and is acknowledged to the host as soon as the copy is in cache. Existing cache entries are updated with the new data. Asynchronously, data is then written to the permanent backing storage media. Write-back caching provides the highest performance, and accelerates both read and write operations, however data protection mechanisms must exist since the only copy of new data may exist in the cache media.

Evaluation Process

Two application environments were tested using PrimaryIO's APA product to accelerate specific application workloads. The evaluation occurred in June 2015, with Evaluator Group having access to all systems being tested. Testing occurred in PrimaryIO's labs, using their equipment for all tests. PrimaryIO personnel performed the tests, with the testing reviewed by Evaluator Group. This analysis in this report includes a review of all tests and equipment and highlights the results found.

Evaluator Group commentary provides context and a narrative assessment of the results as experienced by Evaluator Group personnel. Details of the testing, along with the tools used, are provided in Appendices.

The following were the areas of focus for the validation testing with PrimaryIO:

- Application performance gains as a result of deploying PrimaryIO APA
- The amount of time necessary to optimize application performance
- Disruption to application due to optimization with PrimaryIO APA

Test Cases

Testing was designed to measure relevant performance information of each system, and to provide information that would allow assessment of the validation objectives previously specified. The following test cases were used for evaluation:

1. Measure the results of using PrimaryIO APA with MySQL application
    a. Measure application performance increases
    b. Measure the amount of time required to maximize application performance
2. Measure the application performance increases for SQL Server application
    a. Measure application performance increases
    b. Measure the amount of time required to maximize application performance

In each instance, the test was run on the following systems:

- Intel CPU based server with multiple CPU cores
- A local SAS disk was used for the baseline or un accelerated storage access
- A PCIe based SSD device was used for accelerated storage access
- PrimaryIO APA application specific software to accelerate application

There are multiple methods of measuring storage performance. In these tests, we evaluated performance from the applications perspective. The test details are provided in Appendix A.

Test Findings

As previously indicated, PrimaryIO's APA provided significantly higher application performance rates than the baseline system without accelerated storage. The environment with PrimaryIO APA outperformed the application environment that did not utilize PrimaryIO, as measured by the number of transactions per second processed by the database.

An overview of results is shown below, with additional details provided in subsequent sections:

- PrimaryIO APA provided up to **5X better performance** for a MySQL workload
  - Maximum performance after optimization completed approximately 12 hours total
  - No application downtime was required to move or migrated data
- PrimaryIO APA provided up to **4X better performance** for a SQL Server workload
  - Maximum performance after optimization completed approximately 12 hours total
  - No application downtime was required to move or migrated data

Note: Both test cases were performed by PrimaryIO and reviewed during their progression by Evaluator Group.

Test 1 - Find Performance Optimization with MySQL

This test measured the performance benefits of using PrimaryIO APA with a MySQL based application. The process occurred as follows:

1. System is analyzed for SSD devices and applications
2. Monitoring phase occurs, with PrimaryIO APA analyzing potential performance benefits
3. Recommendations are provided to administrator for database components to optimize
4. Database components are specified by the administrator for caching on recognized SSD devices
5. Optimization phase occurs, where data blocks are cached to SSD device specified

The initial monitoring phase (steps 1 and 2) took approximately 1.5 hours to begin accelerating the application after the table was marked for acceleration. This time was spent monitoring the access patterns, and then optimizing the data by placing the most critical regions in the assigned cache device.

The next two steps, (3 and 4) were performed quickly by the system or application administrator in less than 3 minutes. Once items were marked for optimization, step 5 took approximately 5 hours. During this time, PrimaryIO APA created a cached copy of the application on the caching device, utilizing approximately 20 GB of capacity of the SSD device.

The result was that the queries per second increased from 150 per second using the original storage media, to over 600 queries per second with PrimaryIO APA caching data on the SSD device. This represents more than a 4X increase in performance. This performance is achieved with over 90% cache hit ratio. As the application continued to run, the performance achieved additional gains, ultimately achieving over 95% cache hit ratio, with 750 queries per second, representing a 5X performance increase over the original performance. The total time elapsed for the entire process was 24 hours.

Additional applications could be accelerated, however; each original device will require a unique raw volume on an SSD acceleration device. There is a 1:1 requirement for accelerating volumes, with each data volume requiring its own caching volume.

Results

Shown below in Figure 1 is a chart showing the performance improvement of the application both without and with PrimaryIO APA.

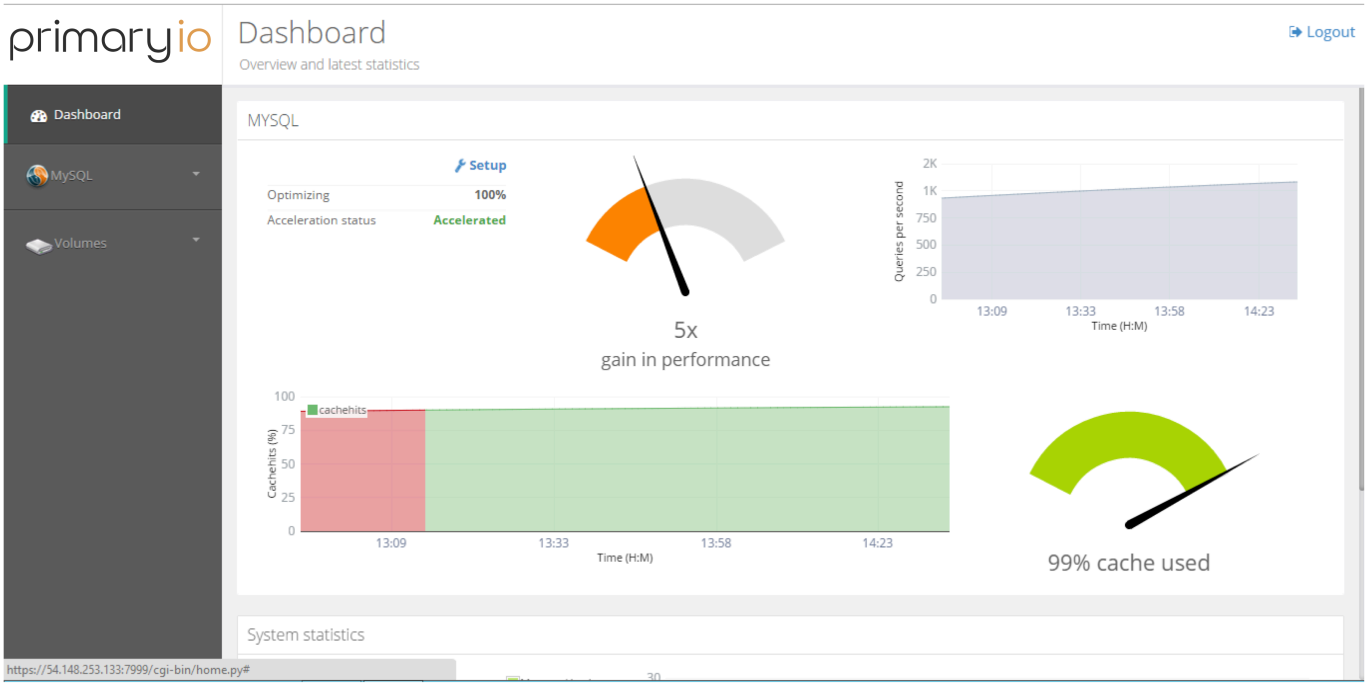

Figure 1: Performance Improvement with PrimaryIO APA for MySQL

Test 2 - Find Performance Optimization with SQL Server

This test measured the performance benefits of using PrimaryIO APA with a SQL Server application. The process occurred as follows:

1. System is analyzed for SSD devices and applications
2. Monitoring phase occurs, with PrimaryIO APA analyzing potential performance benefits
3. Recommendations are provided to administrator for database components to optimize
4. Database components are specified by the administrator for caching on recognized SSD devices
5. Optimization phase occurs, where data blocks are cached to SSD device specified

The recommendations for caching are shown on a per table basis, with internal SQL Server table names. These include tables from the primary database being accessed, along with SQL Server system tables. Any number of these tables may be selected for acceleration. Each item selected will require its own caching device, and the size of the cache may be smaller than the amount recommended. Thus, on a system that has a single instance of SQL Server installed and with multiple databases running, the administrator is able to select which databases and (optionally) system tables should be accelerated, and which device should be used for acceleration may be chosen.

The analysis found that allocating 12 GB of cache would result in a 90% effective cache hit rate. Allocating 40 GB of cache would achieve a 100% cache hit rate. For the purposes of the test, only 12 GB was allocated to the cache, rather than the entire database size of 40 GB. Thus, PrimaryIO APA ASM was only provided with a portion of the database to cache.

Additionally, the recommendations are based upon the I/O access patterns, along with the number and size of caching devices are available for use. Thus, on systems with large capacity for caching, PrimaryIO APA will recommend a larger amount of data be cached, up to achieving a 100% hit rate. In other scenarios, when less solid-state media is available, PrimaryIO APA will recommend a smaller capacity, as long as it can achieve a 90% cache hit rate. Below 90% cache hit rates, it has been found that the benefits are not sufficient. Thus, the minimal recommendation is the most accessed blocks sufficient to achieve a 90% cache hit rate for that data.

The entire optimization process took approximately 12 hours to provide a 4X performance increase and achieved just slightly less than 90% cache hit rate. Additional applications could be accelerated, however; each original device will require a unique raw volume on an SSD acceleration device. There is a 1:1 requirement for accelerating volumes, with each data volume requiring its own caching volume.

Evaluation Summary

One key aspect of the PrimaryIO approach is their independence from hardware products. This enables customers to choose the most appropriate acceleration technology available. Including external, shared solid-state storage or host-based PCIe based solid-state technology. Perhaps more importantly, as new hardware and access technologies emerge, PrimaryIO intelligent caching will be able to leverage these new technologies to accelerate applications.

There are several emerging technologies that will further enhance storage performance, such as NVMe, NVDIMM, new non-volatile solid-state media and others. All of these emerging technologies promise significant performance improvements, but lack the inherent intelligence needed to know what data should be acted upon. PrimaryIO's APA is designed specifically to address the lack of application awareness.

Many new storage technologies will have high costs initially, requiring intelligence in order to be cost effective. Thus application intelligence and awareness of data access patterns will play an increasingly important role in helping new applications utilize new storage technologies cost effectively. The effectiveness of APA technology lies in its ability to understand applications and their data access patterns intelligently, independent of the hardware used.

The tests performed showed that supported applications may be accelerated, without migrating data or application downtime. Additionally, the choice of application components to be accelerated is designed for application owners. These factors enable rapid deployment for critical applications with reduced risk associated with storage migration projects.

Issues and Concerns

One noted concern is that it does take several hours for the cache acceleration features to produce meaningful results. Therefore, it may take several hours after a system is restarted for PrimaryIO APA provides benefits to running applications. As long as systems remain up and running, this is not an issue, but is a consideration for applications that must be restarted.

Another consideration is the lack of HA capabilities inherently designed into PrimaryIO APA. As a result, if the user chooses to use Write-Back caching mode, sufficient availability protection must be provided via another mechanism.

Final Observations

The ability to enhance application performance is a common request for business application owners. However, there is substantial risk whenever application data is moved, migrated or when any configuration changes are made.

Typically, application owners must first request their applications storage to be moved or reconfigured in order to improve performance. The IT or storage administrator must then prioritize the request and perform the actions without impacting IT operations. Due to the potential for downtime or other issues, these activities are often delayed until time, resources and application workloads permit.

As a result, business application owners are not able to have their requests acted upon rapidly. Even worse, if the requested changes do not solve the problem, additional requests must then be made. This de-coupling of applications from storage and perceived delays often leads to dissatisfaction by application owners.

In contrast, the ability for application owners, or storage administrators to analyze and enhance an applications performance in hours without impacting existing storage or other applications can significantly reduce the risk of performing these actions.

Appendix A - Configuration Overview

Test Environment

OS

- Ubuntu 12.04 LTS OS (Test #1)
- Windows Server 2012 R2 (Test #2)

Physical Server for PrimaryIO APA testing

- System for Test #1
  - 3 x 8 Intel CPU's (24 E5 CPU cores)
  - 4 GB RAM
  - 1 x 400 GB Micron P320h PCIe SSD
  - 1 x 1 TB SAS disk
  - OS: Ubuntu Linux 14.04
- System for Test #2
  - 3 x 8 Intel CPU's (24 E5 CPU cores)
  - 8 GB RAM
  - 1 x 400 GB Micron P320h PCIe SSD
  - 1 x 1 TB SAS disk
  - OS: Windows Server 2012 R2

Storage Acceleration

PrimaryIO APA ASM modules with MySQL or SQL Server were used for Test #1 and Test #2 respectively.

Software Testing Applications

Sysbench 0.4.12

The Sysbench tool was used to generate and measure the workload for Test #1 with the following settings:

- Sysbench configuration
  - Access was set for 2 databases
  - Each database was 23 GB + logs = 60 GB total
  - 1 remote replica was enabled
  - Buffer Pool was 3 GB
  - Threads = 32 for each database (64 total)
  - Access pattern (30% of data accessed with 90% frequency)

SQL Query Stress

The SQL Query Stress tool was used to generate the application workload for the SQL Server database running on Windows.

- SQL Query Stress configuration
    - Access was set for 1 database
    - Database size was 40 GB
    - 1 remote replica was enabled
    - Buffer Pool was 5 GB
    - Threads = 64 total
    - Access pattern (30% of data accessed with 90% frequency)

About Evaluator Group

*Evaluator Group Inc. is dedicated to helping **IT professionals** and vendors create and implement strategies that make the most of the value of their storage and digital information. Evaluator Group services deliver **in-depth, unbiased analysis** on storage architectures, infrastructures and management for IT professionals. Since 1997 Evaluator Group has provided services for thousands of end users and vendor professionals through product and market evaluations, competitive analysis and **education**. **www.evaluatorgroup.com** Follow us on Twitter @evaluator_group*

This document was developed with PrimaryIO funding. Although the document may utilize publicly available material from various vendors, including PrimaryIO and others, it does not necessarily reflect the positions of such vendors on the issues addressed in this document.