VIRTUAL INSTRUMENTS®

Moving Data to the Cloud – Best Practices

How Load DynamiX® Performance Analytics can support your public cloud storage initiative

WHITEPAPER

Introduction and Overview

Information can be the most valuable asset to any organization of any size. As a result, the amount of data that is being stored and the amount of time it is kept online is growing rapidly, while the resources and budgets to manage it are not. This presents a considerable challenge to companies striving to exploit information for competitive advantage and storage administrators tasked with storing, managing, and protecting that data. The changing mix of applications and evolving data management requirements is driving major change in storage requirements. IT managers are demanding storage solutions that allow them to deploy complementary tiers of networked storage optimized to meet specific requirements for performance, capacity, reliability, and cost.

Movement of data to the cloud is being considered by more and more companies as a key way to address these challenges. One reason that cloud storage is emerging as a solution is because it holds the promise of not only supporting increased storage needs but also simplifying operations and improving resource utilization and efficiency (read "budgets").

Load DynamiX offers a workload analysis, workload modeling and workload generation solution that storage architects and engineers use to accurately emulate real-world application workload performance in a lab environment, which can include cloud storage targets. Many of our customers are moving data from on-prem to the cloud service providers. Though there are many considerations in these decisions, not the least of which are cost, security, and availability, customers use us to determine the best storage tier based on application workload performance requirements.

This document is a guide covering best practices in ensuring that performance SLAs are met with moving data to a public cloud infrastructure. The best Load Dynamix practices use a guided approach through the observable manifestations of complex hardware/software interactions in order to model and analyze performance characteristics, and test the cloud deployment, in a pre-production environment.

Performance Testing for Cloud Data – Why It's Important

This paper addresses common questions that our cloud testing methodology can help answer:

- Can I move my data to the cloud and meet performance SLAs?
- Can I use a tiered approach to cloud storage?
- How do I select the best cloud storage provider and best storage gateway product?
- How much does my cloud / tiered performance degrade with dedupe, compression, snapshots, etc?
- Where are the performance limits of potential cloud offerings?
- How will my application workload behave when it reaches its cloud data performance limits?
- Does performance degrade over time and to what extent?

Cloud Data Use Cases

It's important to remember that the majority of corporate data isn't being used by primary applications. For each byte of data, there are 5 – 7 copies being used for testing, disaster recovery, and archiving. Because most of these uses do not require millisecond response times, these data copies are more and more being stored in public cloud datacenters.

- By far, the most popular use of cloud storage is for disaster recovery. Typically, you'll replicate to a cloud provider. DR can be slow or it can be fast, but performance limitations are usually bandwidth related. Performance indicators like latency aren't usually a factor.
- Cloud is also used as primary storage, often with on-premises caching, while the primary copy of the data is in the cloud. In this model the on-prem access is considered the hub of the deployment. With highly dispersed workforces, or when file sharing is the use case, the cloud could be the hub. Testing needs to occur with any load generation facing the hub.
- Tiering is becoming more common, with the cloud playing an integral part. Many storage systems enable you to automatically move data between tiers based on user defined criteria. Tiering, whether in the cloud or not, brings its own special testing requirements. Specifically, many tiering tools don't adjust data placement more frequently than once a day, so your test periods need to be aware of that.
- And last but not least, there's development and testing. It can be expensive to provision storage for new app development, for two reasons. First, the ROIs for big CAPEX can be pretty far out in time, especially for long development cycles. And second, because the storage system probably has to scale to where it'll need to be in the production deployment, you may need a lot of storage. So many developers are purchasing cloud resources as development progresses. Stress testing can still happen early due to the easily scaled cloud resources. This requires

performance testing, something that doesn't always happen during the development cycle.

The Cloud's Effect on Testing

Storage in the cloud creates challenges for performance measurement, in part due to the massive variability you see in shared environments. In the cloud, unless you are doing a bare metal deployment, you are likely to have noisy neighbors. "Noisy neighbor" is a phrase used to describe a cloud computing infrastructure co-tenant that monopolizes bandwidth, disk I/O, CPU and other resources, and can negatively affect your cloud performance. A bare metal cloud deployment can mitigate this problem by creating a single-tenant environment. While single-tenant environments avoid the noisy neighbor effect, they do not solve the problem of cloud testing.

Infrastructure over-commitment, or when an environment is shared by too many applications, limits overall cloud performance. The use of all flash arrays can mitigate the problem. Some cloud storage providers are deploying all-flash arrays that have built-in storage I/O quotas that can be set at an individual VM layer. But you can still have contention in other areas, like the network, which can produce very inconsistent test results.

In testing, due to the variability of the shared environment, it's hard to replicate results, even over multiple runs. Differences in results can easily be 10X or greater; and it's random. There can be a big difference between disk and network performance by providers and even within providers, to a lesser extent. And you can get "failure" instances, where one instance may get so bad it needs to be restarted. This really skews test results. And finally, something that rarely gets talked openly about is the issue of cheating. It has been alleged that some providers cheat on IO performance; they will configure systems to get better IO but may sacrifice availability, until there's a problem. Then it all changes.

Mitigating the "Cloud Effect"

To mitigate the variability of the cloud, you need to repeat your performance tests many times. So it's even more important to start with an accurate workload model than with on-prem testing. Even though you'll get a distribution of test results, you can't take the average as your answer. There's simply too much variability. Instead, you should think in terms of how much cloud overprovisioning you'll need to do to achieve a certain probability of achieving your SLAs. For example, your performance SLA might call for 20ms response times. Your average test results might be 20ms. This means that 50% of the time, you won't meet your SLA. In this scenario, you may have to divide your workload over greater links and targets to achieve a higher percentage of 10ms results.

Load DynamiX Cloud Storage Performance Testing Methodologies

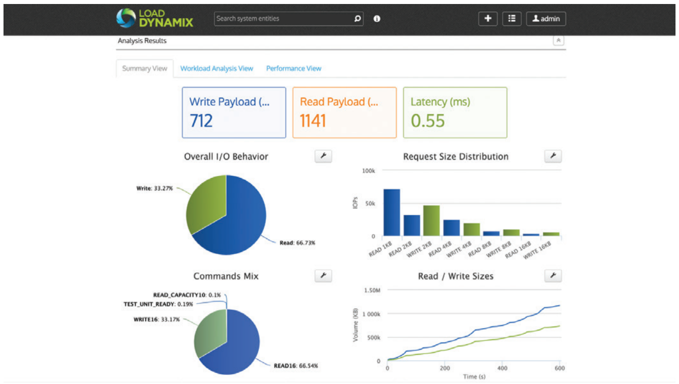
The methodology you choose is largely based on the problems you are solving for. So let's start with the most basic cloud performance question: Can I move my data to the cloud and meet performance SLAs? For that, we turn to our Workload Performance Modeling methodology.

Workload Performance Modeling

First, characterize your existing workloads, the ones you are considering migrating. While many storage engineers have a pretty good understanding of their current performance (e.g. Latency) from the perspective of their storage arrays, this doesn't help them much when trying to size new arrays, or build new storage infrastructures, or migrate workloads to the cloud. What's missing is the analysis from the point-of-view of their applications, that is, workload analysis. Workload analysis is the ability to statistically analyze application workload data over a period of time and create workload models that can be used as a basis for storage performance planning and troubleshooting.

Load DynamiX Enterprise offers Storage Performance Analytics, a capability that allows storage engineers to analyze temporal and spatial workload behavior via powerful visualization to better understand workload I/O patterns that affect storage performance. The Workload Data Importer and Workload Analyzer features of Load DynamiX Enterprise processes historical production data and creates a detailed workload profile. The storage performance analytics capabilities within Load DynamiX Enterprise include solutions for real-time and historical workload acquisition and analysis.

In the illustration below, the Workload Analyzer module of Load DynamiX Enterprise helps visualize KPIs like latency, throughput, or IOPS, R/W mix, random / sequential mix, block size distributions, temporality over any time period. This data can be used as the basis to automatically create the workload models.



Analysis Results

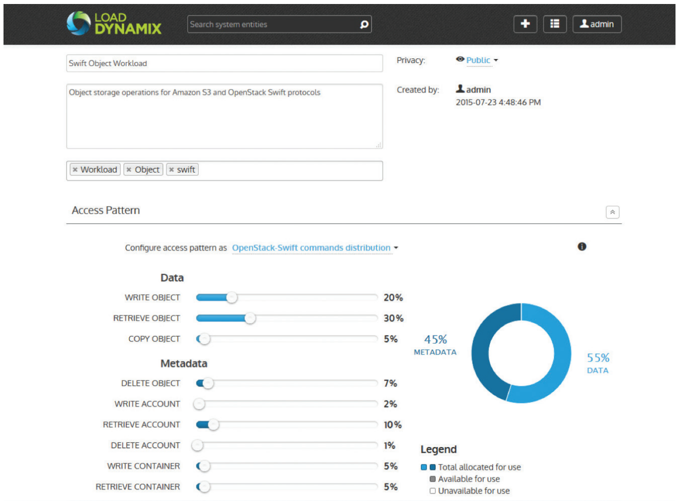
Write Payload (... 712

Read Payload (... 1141

Latency (ms) 0.55

Figure 1: Workload analysis of production system

The object of Workload Modeling is to stress a storage system, in this case, a cloud storage system, under a realistic simulation of the workload(s) it will be supporting in production. Using the Workload Data Importer function of Load DynamiX Enterprise, you can easily create a realistic model of your existing application behavior. Or if this is a green field environment, you can start with one of our many protocol-based workload library examples and adjust the parameters to suit your expectations. There are examples for fibre channel, NFS, SMB, iSCSI, and object storage. An example partial screenshot of an object workload model for Amazon S3 and Swift is below:



Swift Object Workload

Object storage operations for Amazon S3 and OpenStack Swift protocols

× Workload × Object × swift

Privacy: Public

Created by: admin
2015-07-23 4:48:46 PM

Access Pattern

Configure access pattern as OpenStack-Swift commands distribution

Data

WRITE OBJECT 20%
RETRIEVE OBJECT 30%
COPY OBJECT 5%

Metadata

DELETE OBJECT 7%
WRITE ACCOUNT 2%
RETRIEVE ACCOUNT 10%
DELETE ACCOUNT 1%
WRITE CONTAINER 5%
RETRIEVE CONTAINER 5%

Legend
- ■ ■ Total allocated for use
- ■ Available for use
- □ Unavailable for use

Figure 2: Partial example of an object workload model for Amazon S3 and Swift protocols

This workload is then executed on one of the Load DynamiX Workload Generation Appliances, located at the same site as the production application servers, and pointed at the cloud target. Tests are run, including "what if" scenarios that simulate changes in the workload, or in the configuration, such as the effect of load balancers. A simple diagram of this can be found below in figure 3.

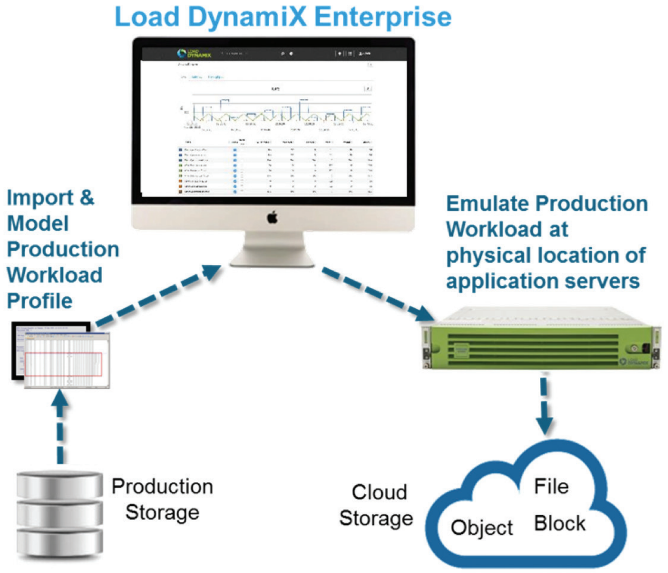

Load DynamiX Enterprise
Import & Model Production Workload Profile
Emulate Production Workload at physical location of application servers
Production Storage
Cloud Storage
File
Object
Block

Figure 3: Diagram of cloud data performance testing methodology

An example may be an object-based workload utilizing an Amazon S3 target. The AWS S3 workload model allows users to characterize access patterns with as much detail as needed, and Key Performance Indicators (KPIs) offer the analysis you need to assess the viability of running your workload on the cloud storage targets. An example showing PUT response times follows below:

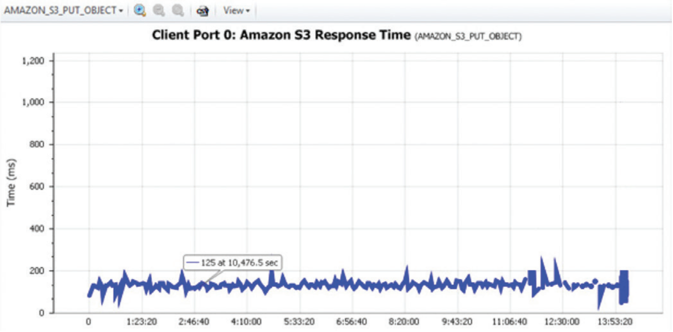


Figure 4: AS3_PUT response time

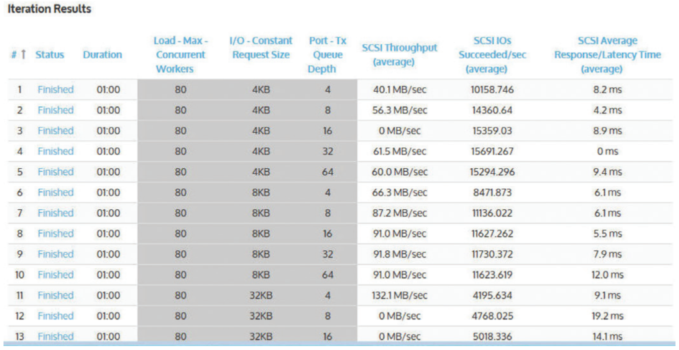
Performance Profiling

Another related question concerns the performance limits of potential cloud offerings. For instance, you may ask "how much does my cloud / tiered performance degrade with dedupe, compression, snapshots, etc?" To answer these sorts of questions, we use the Performance Profiling methodology.

This methodology is sometimes called "performance corners testing" or "multi-dimensional benchmarking". The objective of performance profiling is to characterize the behavior of a storage system under a large set of workload conditions. It provides a very useful outline of the workload-to-performance relationship. This provides the storage engineer with a map of the behavior of the cloud storage system: it makes it easy to understand where sweet spots or bottlenecks may be, or what workload attributes most directly affect the performance of the system. Engineers can then use this information to optimally match their workloads to tiers of their storage systems, and divide data between on-prem and cloud-based offerings.

In the Load Dynamix Enterprise application, this methodology is enabled by an iteration workflow that allows the user to iterate on any of the many workload characterization attributes exposed by Load Dynamix workload models (areas such as load profile and command mix, etc). This workflow can result in tests that stress the cloud storage system under hundreds or even thousands of workload configurations, with automated test execution, aggregation of data and presentation of results.

In figure 5 below, we see a very small portion of the results of a test we ran to examine the number of concurrent workers the cloud target could support with different I/O request sizes and queue depths. These are results with 80 workers. Similar results were obtained with over 8,000 workers.



Iteration Results

# ↑	Status	Duration	Load - Max - Concurrent Workers	I/O - Constant Request Size	Port - Tx Queue Depth	SCSI Throughput (average)	SCSI IOs Succeeded/sec (average)	SCSI Average Response/Latency Time (average)
1	Finished	01:00	80	4KB	4	40.1 MB/sec	10158.746	8.2 ms
2	Finished	01:00	80	4KB	8	56.3 MB/sec	14360.64	4.2 ms
3	Finished	01:00	80	4KB	16	0 MB/sec	15359.03	8.9 ms
4	Finished	01:00	80	4KB	32	61.5 MB/sec	15691.267	0 ms
5	Finished	01:00	80	4KB	64	60.0 MB/sec	15294.296	9.4 ms
6	Finished	01:00	80	8KB	4	66.3 MB/sec	8471.873	6.1 ms
7	Finished	01:00	80	8KB	8	87.2 MB/sec	11136.022	6.1 ms
8	Finished	01:00	80	8KB	16	91.0 MB/sec	11627.262	5.5 ms
9	Finished	01:00	80	8KB	32	91.8 MB/sec	11730.372	7.9 ms
10	Finished	01:00	80	8KB	64	91.0 MB/sec	11623.619	12.0 ms
11	Finished	01:00	80	32KB	4	132.1 MB/sec	4195.634	9.1 ms
12	Finished	01:00	80	32KB	8	0 MB/sec	4768.025	19.2 ms
13	Finished	01:00	80	32KB	16	0 MB/sec	5018.336	14.1 ms

Figure 5: Extract of output report of Iterator function, showing effect of changing parameters on performance

Conclusions

By following these outlined methodologies, our customers have realized the following benefits:

- Performance assurance: Ensure cloud storage solutions will meet performance SLAs under specific workloads and confidently choose the optimal solution for those workloads.
- Reduced storage costs: Choose the lowest cost cloud-based systems for specific workloads; quantify the benefit and effects of every cloud-based system.
- Increased uptime: Identify performance bottlenecks prior to production deployment; validate all infrastructure changes against workload requirements and troubleshoot more effectively by re-creating failure-inducing workload conditions in the test environment.
- Acceleration of new application deployments: Accelerate time to market by validating new applications on cloud-based systems, making deployment decisions faster and more confidently.

Sales
sales@virtualinstruments.com
1.888.522.2557

Training
training@virtualinstruments.com

Website
virtualinstruments.com