Lab Insight

Performance for Next Generation Workloads with IBM Cloud Object Storage (COS)

Author: Russ Fellows

October 2019

Evaluator Group

Introduction

Enterprises understand that the value of their organizations are increasingly measured based upon the amount and quality of information they retain. Information is more than data, consisting of context and meaning along with the underlying raw data. Merely retaining data in off-line repositories does not allow a company to access and utilize their information. In order to derive value, data along with contextual information must be stored in online repositories that facilitate analysis and processing, thereby enabling decisions and inferences.

As the amount of available data grows, the sheer quantity requires new approaches to be able to manage, store and analyze this data while also providing resiliency and protection features that scale in a cost-effective way. Traditional storage systems were designed for processing comparatively limited amounts of data for transactional databases or file workloads at a single datacenter, with additional locations used for disaster recovery. These primary storage systems were not designed to support multiple Petabytes of data split between multiple sites.

It is within this context that object storage systems have evolved to support nearly unlimited scalability across multiple physical locations, while still providing cost effective retention and processing of data to support modern enterprise business operations. An object storage system can serve as a repository for inactive data, or for active systems that process data to gain insight via big data analytic applications or by using artificial intelligence and deep learning.

In this paper, Evaluator Group outlines some typical workloads and uses for object storage systems along with the features and requirements of these systems. We then review the characteristics of IBM's Cloud Object Storage (COS) system, specifically with respect to how it can facilitate modern applications and workloads that require access to large of amounts of data.

All testing was performed in IBM facilities by IBM engineers and reviewed by Evaluator Group to ensure consistency and accuracy of the data presented. The primary focus of testing was to explore how the IBM COS system scales with additional storage nodes and disk drives.

Object Storage Use Cases

Evaluator Group works with IT users and performs primary research about how particular technologies and concepts are being adopted and for what purposes. In an attempt to describe data repositories for emerging workloads, terms such as "data lakes", "big data" and others have been used. Object storage systems were designed to address the problems these terms describe, namely to store data more efficiently and at significantly larger scales than traditional storage systems. Specifically, object storage is designed to manage and protect large amounts of data efficiently while also enabling active processing of data retained within them.

Protection of data in object storage systems is fundamentally different from other storage systems. Due to the large capacity requirements and the need for low cost capacity retention, it is imperative that object systems use more efficient protection mechanisms than making multiple complete copies of data.

For systems at the petabyte and larger scale, making complete copies of data across multiple locations is not feasible due to scale, cost and network bandwidth limitations. However, with the importance of data, mechanisms must exist to ensure that data is protected from loss, due to malfunctions or human errors.

Systems utilized for online repositories must support scaling to large capacities across multiple geographic locations while providing self-protecting mechanisms to ensure data is durable. The notion of durability includes resiliency (the ability to protect without making multiple copies), support for multiple distributed locations along with security and longevity concerns.

Next Generation Workloads – Online Repository

Object storage systems are being used as online content repositories for a wide variety of next generation workloads, both for active data processing and as a long-term retention solution. Many modern, or next-gen workloads require access to large amounts of data as sources of information but also for storing the processed or intermediate data sets.

Object storage is utilized with these workloads for several reasons including the ability to store large amounts of data cost effectively, as well as the use of large files or objects for retaining data. Since many disk access methods for backup evolved from tape, object systems ability to support large data sets, stream data sequentially at high rates and retain information cost effectively, are all advantages compared to using traditional block or file-based storage designed for primary storage.

Applications using object storage for data protection include backup applications, many of which now support sending data directly to on-premise or cloud-based object storage systems. Additionally, many archiving applications also support object storage.

Information Retention – Analytics, AI and Big Data

Data analytics is a broad area that covers multiple uses, applications and vertical industries. This can include artificial intelligence (AI), with deep learning and machine learning (DL-ML). Other categories include analytic tools such as Splunk or Apache Spark, which operate by streaming data and performing an analysis on very large data sets. These applications require access to vast quantities of data and meta-data, which then generates additional meta-data, which may add classification or further refinements and information to the underlying data sets.

Other uses such as oil and gas exploration or bio-science research may also be included more generally under the heading of data analytics.

Some workloads may have relatively small objects, under a megabyte in size, while others may have very large objects that are measured in the 10's or hundreds of gigabytes. While processing this data, many applications need the ability to read data quickly, while other portions of the application may require the ability to write data at high rates.

Moreover, in order to support a variety of workloads, object storage systems need to be flexible enough to provide the ability to read and write data at high sustained rates.

Industry Applications

There are a wide variety of industry applications, with specific development dependent upon each industry's needs. Many workloads have evolved to include processing of significant amounts of data, that previously may have been retained offline in either electronic, or in some instances, paper forms. Medical and life sciences, media and entertainment, and energy production are a few industries that now have the need to retain and process significant amounts of data (often measured in the petabyte), even for smaller companies that are working in these fields. The amount of data a company must manage and process is less dependent upon their size than the fields in which they operate.

As discussed, AI is gaining traction across many verticals and particularly for DL. The amount of data available has direct implications in the quality and results produced via AI, and specifically for DL. Additionally, many industries have specialized workload, retention or other regulatory requirement issues that dictate the retention of large amounts of data independent.

Moreover, an increasing number of companies across broad segments desire a method to cost effectively retain large amounts of data that enables them to both process and gain insight while still adhering to regulations and security requirements they may have. Object based storage systems have become popular for IT consumers looking to solve exactly these challenges, due to the unique capabilities and design characteristics of object storage.

Object Storage Requirements

Although requirements vary depending upon individual companies, workloads, and industries, there are several general considerations that are important when considering an object storage system. They fall under categories such as storage efficiency, security, resiliency, scalability along with performance and in some instances, regulatory compliance capabilities.

Questions typically include some of the following:

- How is data stored and what are the implications for using one vs many locations?
- How are small vs. large objects managed ?
- What are the performance requirements for my applications?
- How is indexing performed and is both metadata and content searchable?

- How is data protected, via copies or with forward error correction?
- Do you require compliance or other retention features?

Requirement Details

Data Efficiency – This includes details of how the data is stored and managed. Architectural differences include making multiple copies of data or other more efficient methods for storing data. The ability to support multiple locations efficiently and for a variety of object sizes are also important aspects.

Data Integrity and Resiliency – These features are important to ensure that data is not lost due to system failures or error conditions. In some instances, these features can protect against human errors, although they will not prevent deliberate or accidental removal of data.

Regulatory Compliance - Additionally, systems should provide protection against data removal through retention locks, versioning or other methods. The most complete certification for compliance is SEC 17a4 which specifies legal holds, audit trails and object versioning.

Data Security – Many companies are subject to regulations or corporate governance that dictate the use of encryption as one means of providing data security. Although encryption itself is not sufficient to ensure security, it is one important aspect of an overall plan to provide security of retained data.

Object System Performance – The performance of IT systems is always an important consideration, and typically one of the most important after ensuring data is protected and available. If performance needs are not met, application owners may view the application as failing. Thus, cost effectively achieving performance required by applications is important.

Cloud Object Storage – IBM COS

COS Features

The IBM Cloud Object Storage system is based upon a shared nothing architecture, which not only eliminates single points of failure, but also eliminates potential bottlenecks anywhere within the system. By separating access from information dispersal and storage, each may be scaled independently as required by the application environment. Additionally, with access via standard S3 API's, it enables multiple applications residing at different locations to utilize the COS system without limitations.

> ***Evaluator Group Comments: IBM COS derives its capabilities from its shared nothing design together with a unique, flexible dispersal method that together eliminate single points of failure and performance bottlenecks, while providing flexibility to protect and scale small deployments to extremely large geo distributed deployments. This provides the customers utilizing COS the ability to maintain security and data integrity cost effectively, while scaling capacity and performance to cloud scale operations.***

One of the most unique aspects of the IBM COS system is the method used to split data into multiple segments, adding resiliency and fault tolerance while simultaneously adding I/O throughput. The term for this technology is information dispersal algorithms, or IDA which are related to RAID and forward error correction codes, but with more flexibility. Storage systems designed for primary application workloads utilize local data protection, providing resiliency from a local system failure. However, in order to provide backup and disaster recovery, multiple additional systems and technologies must be added to provide the necessary ability to withstand human, system and environmental errors.

IBM COS is designed instead to utilize IDA technology to provide reliable access to data from multiple failures including network outages or complete system failures. The ability to divide an object up to as many as 24 systems or as few as two systems provides extreme flexibility. Additionally, with two different components of the IBM COS system active for data access and retrieval, the overall system can be scaled based on object type, size and application specifics.

The IBM COS system has the following characteristics:

- Object storage system
  - Nearly unlimited number of objects supported, from 72 TB to nearly unlimited Exabytes in size
  - Available in hardware systems and as software only
- IBM COS Information Dispersal Algorithm (IDA)
  - Parallel access for performance, global dispersion possible
  - Data checking and correction on slice basis
  - Flexible dispersal, from as few as 3 up to 36 slices per object across 1 to 24 sites
  - Forward Error Correction with Cauchy – Reed Solomon erasure codes
- Multiple Access Methods
  - S3 REST API
  - Underlying objects stored with metadata on storage nodes
  - Available on premises, in IBM Cloud and via hybrid access
  - NAS gateway for file access available to COS back-end
- Data Protection Features
  - Security with encryption for data and communications – self-contained keys
  - Multi-tenancy support
  - Integrity verification and correction – on access and as background scan
  - Regulatory compliance capabilities - WORM mode, legal hold, audit trails for SEC 17a-4
- Scale and Performance
  - Scalable front end “Accesser” and back-end “Slicestor” nodes
  - Read caching to SSDs for performance accelerations – SmartRead
  - Ability to complete write operations in distributed, lossy environments – SmartWrite
- Application Support
  - Ability to support standard S3 object interface API’s and calls

— Custom application via language SDK's (C, Java, JavaScript, Python and others)

IBM COS Performance

As described previously, object storage system performance is dependent upon multiple factors and has greater variability than is seen with block storage system workloads. Performance for object storage systems will change depending upon the workload or access types, geographic deployment and distances along with the size of the system. For these reasons, it is difficult to compare performance results for an object storage system to that of another vendor unless the same workload and deployment constraints are utilized.

To date, no object storage benchmark has been developed that can rationalize these permutations into a confined set of configurations and tests cases. In lieu of an existing benchmark or standard, Evaluator Group worked with IBM personnel to review their internal performance testing for multiple IBM COS configurations. The results are intended to show the performance capability of smaller configurations while showing the scalability of IBM's COS. These results are designed to help IT consumers understand how object system configurations may impact various performance such as throughput and access latencies as configurations are scaled.

For the testing performed, the IBM COS systems utilized high capacity rotating disk drives, as is typical for the majority of object storage system deployments currently. Additionally, IBM COS now supports solid-state media for scenarios where sustained high I/O rates and throughput for multiple workloads is important. Due to the impact of multiple workloads, access patterns to the underlying storage media will become highly random in nature, similar to primary storage systems supporting multiple applications.

Measures such as latency are relevant for object workloads, although latency is typically less important than other measurements such as throughput, and time to first byte of data. In particular, streaming and batch applications such as Apache Spark, backup repository and media are all examples of applications that begin operation on streaming data, rather than requiring all data transfers to complete before application processing. Moreover, measures of latency for entire transfers may be irrelevant for object-based workloads.

Finally, object storage operations differ from block storage, utilizing "Put" operations rather than a block "write". A "Get" operation is used rather than a block storage "Read", and thus these terms "Put / Write" and "Get / Read" may be used interchangeably when describing operations for object storage systems. As discussed, scaling capacity and performance are both important characteristics, with linear scaling of performance seen as the ideal goal. Due to the IBM COS system design and workload, the number of front-end "Accessor" nodes can impact performance in addition to the number of storage or "Slicestore" nodes.

For simplification, all configurations shown utilize 4 Accessor Nodes to reduce the number of test cases, even though they could be a limiting factor on performance for the larger object sizes and for the larger 1,272 disk configurations. Moreover, the results shown below were not designed to be optimal, but instead representative of the systems performance for several smaller configurations. For all testing, all of the nodes were collocated, thus eliminating delays from network latency.

Small Configuration Scalability

One of the advantages of the IBM COS systems design is the flexibility of configurations. While there are literally hundreds of configuration options, a set of configurations were tested that are designed to showcase how a small environment could be scaled, using a small number of Slicestore nodes, each with just 12 disk drives. In this testing, the initial configuration consists of only 3 nodes, and thus will utilize an IDA dispersal that is best suited for smaller deployments. This is known as "Concentrated Dispersal" and ensures that data is dispersed across the 3 nodes so that no data is lost when a single node becomes inaccessible. Details of the IDA are provided in Appendix A.

Put / Write Workload

A total of four configurations were tested, each adding another 3 Slicestore nodes with 12 disks, thus providing 3, 6, 9 and 12 node configurations. As shown below in Figure 1, write or "put" performance scales with the number of storage nodes and also improves with larger object sizes. It should be noted that due to how small objects of less than 1MB are stored, performance for slightly larger objects can degrade slightly, as shown below by the lines for 900KB and 3.5MB objects. Large objects as represented by the green 100MB line typically have the highest throughput.

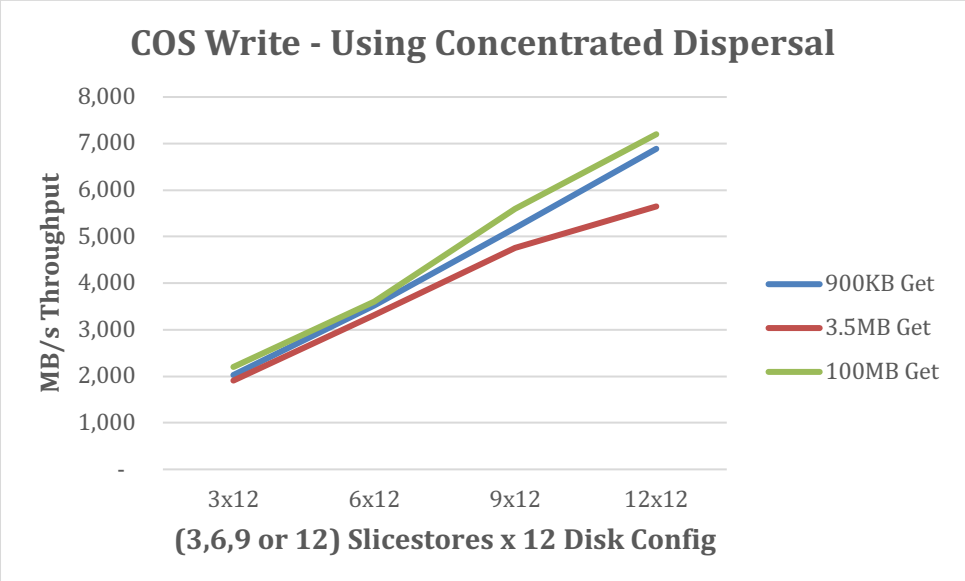

Figure 1 : IBM COS – Slicestore Scaling

Get / Read Workload

Using the same dispersal algorithm and hardware configurations, a set of read or "get" operations were performed, with the corresponding performance as shown below in Figure 2. It should be noted that read performance is more dependent upon the number of devices available in order to provide multiple I/O operations. This is due in part to the use of standard, high capacity hard disk drives. In order to improve read performance, additional disk drives are required, as will be shown in subsequent testing.

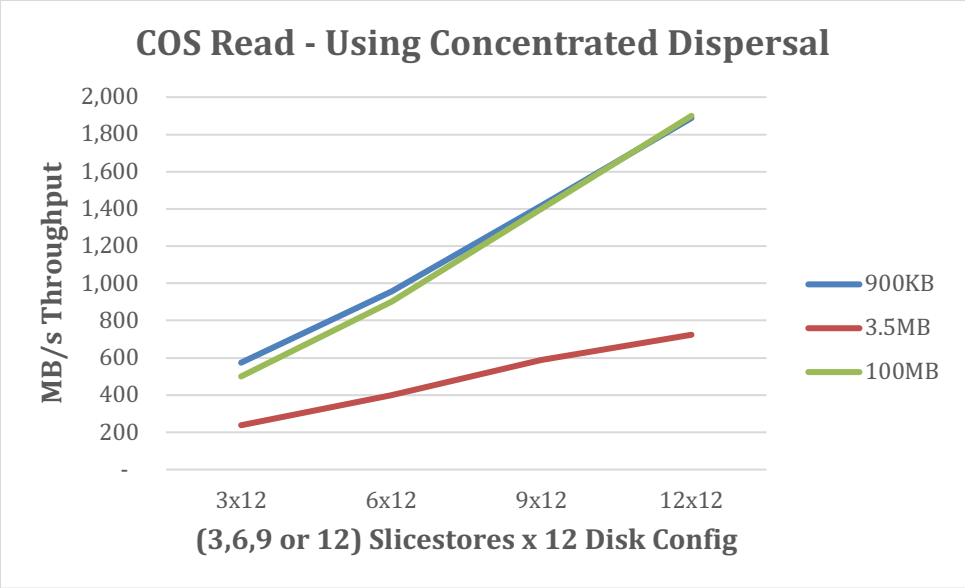

Figure 2 : IBM COS – Slicestore Scaling

The results for write/put workloads show the ability to scale nearly linearly when going from an entry-level 3 node by 12 disk Slicestore configuration up to a larger 12 node by 12 disk configuration. Both Read (get) and write (put) operations scaled nearly linearly with better overall write throughput due to the limited number of disk devices in the tested configurations.

Large Configuration Scalability

Scaling performance is dependent not only on the number of storage nodes, but also how many devices are in each storage node. Using a standard, wide scale dispersal configuration (IDA details are in Appendix A), the IBM COS system is able to linearly scale Read operations as the number of drives scales from 144 total drives up to 1,272 drives. Additionally, write throughput scales, although not as significantly due to the already high write throughput levels with a smaller configuration.

Write performance improves for all object sizes by using Slicestore storage nodes with more devices, growing from much already high throughput even with only 12 drive enclosures to even better results with larger, 106 drive chassis. These results are shown below in Figure 3.

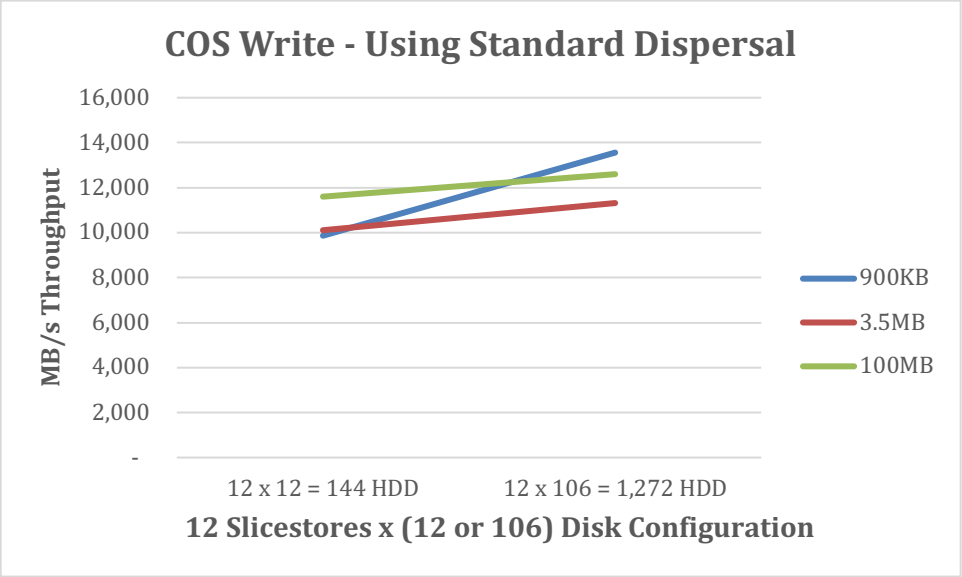

Figure 3 : IBM COS – Slicestore Scaling

Read performance is provided in Figure 4, which shows significant throughput performance improvement for all object sizes by moving from 144 total devices to 1,272. Testing utilized both 12 and 106 device configurations as shown. A twelve node Slicestore with twelve devices consists of (12 * 12 = 144) HDD's. Similarly, the twelve node Slicestore's with 106 devices had (12 * 106 = 1,272) HDD's

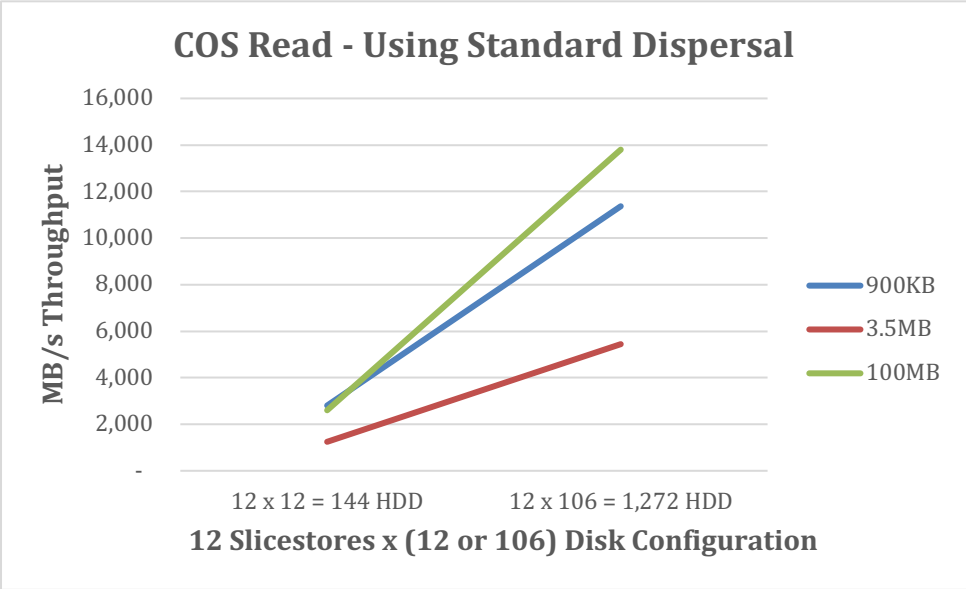

Figure 4 : IBM COS – Slicestore Scaling

Both read and write operations show performance gains when using larger disk nodes, with read operations in particular showing benefits due to the ability to perform multiple read operations on more devices. This is due primarily to the characteristics of spinning media devices, rather than any inherent limitation in the IBM COS design, or node configurations. The use of large, high capacity devices provides cost effective capacity, albeit with a limitation on the number of operations that can be completed. As with primary storage, the use of solid-state media or a large number of rotating media devices can overcome these limitations.

Final Thoughts

As the role of data within organizations shifts, there will be an increasing role for applications and systems that can store, manage and process large amounts of data, typically measured in Petabytes or more. Traditional storage systems are not able to scale capacity or performance cost effectively, preventing organizations from using these systems to gain insight into much of the data being retained. Object storage systems were designed to address these needs by cost effectively storing nearly unlimited amounts of data, while enabling the data to be processed and analyzed. An important consideration is how a performance scales performance as additional components are added to an object storage system.

Shared nothing designs have many advantages, particularly when attempting to provide storage for modern, cloud-scale applications. By eliminating single points of failure, individual deployments of any size are able to provide reliable scalable resources. The ability to scale components independently is due in part both to its shared nothing design along with the unique IDA methods utilized. It is highly inefficient to create multiple copies of data when capacities approach the petabyte scale. IBM COS's use of a custom IDA method enables scaling performance and capacity while providing reliability and security without creating multiple copies of data.

> ***Evaluator Group Comments: IBM's COS is one of the leading systems designed to solve the challenges posed by storing large amounts of data while still enabling business processing. By utilizing one of the most flexible information dispersal algorithms available, IBM COS can operate with as few as 2 or 3 nodes at a single site, or scale up to multiple sites or hundreds of nodes with both on premise and cloud deployments.***

As shown in this paper, the IBM COS is able to scale performance, somewhat independent from the amount of capacity. Adding additional Slicestore nodes adds performance, with the higher density storage Slicestore nodes adding both more capacity and performance than the Slicestore nodes with fewer disk drives. IBM's Cloud Object Storage system provides highly flexible IDA based data integrity to ensure data is not lost, while still efficiently storing data for on-line application utilization.

Appendix A - Test Setup Overview

Provided below are the specific components and configuration details of the test environment.

Accessor Nodes:

- Model A4105 Accessors (4 nodes for tests unless otherwise noted)
- OG workload running on Accessor nodes
- Cleversafe OS: 3.14.4.110

Slicestore Nodes:

- Model: Slicestore12 or SliceStore106
- Cleversafe OS: 3.14.4.110
- Each Slicestore utilized 12, or 106 HDD's – ea. HDD = 7.2K, 8 TB

Other Infrastructure:

- Ethernet Switch: 10Gb/sec ethernet switch

Workloads

- IBM OG tool, running on "Accessor" nodes to generate synthetic workload operations
- Standard Workloads: utilize "Standard Dispersal Mode"
    - IDA: (12/8/10 IDA) (12 wide, 8 reads, 10 writes), with 1.5:1 data expansion
- Scalability Workloads, utilize "Concentrated Dispersal Mode"
    - IDA: 18/9/11 (18 wide, 9 reads, 11 writes), with 2:1 data expansion
- Indexing was disabled as indicated for reported test results

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