When Implementing Object Storage Makes Sense

With its low cost per GB, high scalability, resiliency and durability, an object storage implementation may prove advantageous over file storage for long-term archives, data backup, big data projects and mobile applications.

Object Storage Preps for Starring Role

Object storage is getting the type of attention that plenty of data center technologies can only dream about. And it's doing what any technology in the spotlight must do to avoid being written off by end users as yet another technology trend. Slowly, steadily, object storage has been delivering as promised. First as a natural fit for archiving, where its metadata capabilities make it invaluable, and now in new roles beyond archiving.

Indeed, some shops have figured out that a well-planned object storage implementation, along with file sync-and-share technology, means they don't need a public cloud provider after all. For big data repositories, for mobile workforces, and for crucial data protection, object storage is gaining traction and fans. The ability of object storage to handle deduplication globally and avoid redundancies is just one way the technology is making inroads into today's data centers and helping to secure a spot in the data center of the future.

If you're like most IT pros, you already have questions—and some answers—about object storage. What's the best way to acquire and implement it? Should you start with software or a fully functional platform? Why choose an object storage system over file storage? What are the best use cases in your shop? How can it aid your big data projects?

Those questions and more are answered in this collection of expert technical content, designed to help you make the most of this compelling technology. ■

Ellen O'Brien
Associate Editorial Director
SearchStorage

Object Storage Poised to Break Away From Just Archiving

THERE ARE TWO reasons object storage is considered a good match for the archive storage tier. First off, it delivers only modest performance and a low cost per GB. In addition, object storage has the capability to extend data durability, and that makes it even more of a natural fit for archiving. But the use cases for object storage are growing. Today's object storage vendors continue to improve the technology and object storage is preparing to move out of its traditional role as an archiving technology—and into more of a production role in the data center.

To understand the latest trends in object storage, it's helpful to understand why the technology is such a great fit for data not frequently accessed. That data almost always can benefit from the richer metadata attributes that object storage allows. These richer attributes allow for things like multiple versions to be tagged, expiration dates to be set and descriptive context to be applied. All of this makes searches easier to conduct, and the results of those data searches and more relevant with metadata guiding the way.

Also, object storage systems use either replication or erasure coding for data protection, a scheme better suited for high-capacity hard disk drives (HDDs). The challenge is that object storage's rich metadata capabilities add latency that impacts performance. And since most object storage systems are scale-out in nature, it is difficult for them to use power efficiently; rich metadata requires that all the nodes are powered and available.

So, where does that leave us? The good news is that object storage vendors are addressing these shortcomings. For example, the impact of metadata latency is minimized by storing metadata either in RAM or flash storage. Storing metadata in DRAM—or on flash—also enables vendors to address power consumption by powering down off nodes and their HDDs. These improvements also let object storage move beyond its traditional archive role.

OBJECT STORAGE FOR DATA PROTECTION

A next logical step for object storage is to be a data protection storage area. Like disk backup appliances, object storage systems are very cost effective from a dollar-per-Gb basis. With DRAM and flash storage to speed up metadata processing, these systems can ingest data significantly faster than before. In the past, ingestion rates were a considerable challenge to using these systems as backup targets.

While the ingest speed of an object storage system still may not be as fast as a disk backup appliance, they do offer other advantages. The first is scalability. Object storage systems are typically more scalable than most disk-based backup systems. Object storage systems also tend to be more efficient because they can apply deduplication globally. That means the software will compare all files on an array and it will only store the redundant data once. The deduplication occurs across arrays so redundant data won't be stored—even if those redundancies occur on different arrays. That's a big bonus when it comes to efficiency. Finally, object storage systems are often cost effective since the organization can use commodity, off-the-shelf servers and HDDs.

Now consider snapshots—object storage is ideal for storing snapshot replication jobs from production storage. An increasing number of production storage vendors (that is, storage vendors who specialize in production storage and not backup and archiving) have direct integration to object storage. This combination allows these systems to keep an updated copy of production data on the object storage system.

OBJECT STORAGE AS PUBLIC CLOUD REPLACEMENT?

Another use case for object storage is as a replacement for file serving. Files are a form of objects, so object storage systems are ideally suited for this task. When combined with file sync and share (FSS) software, these systems can help jettison file serving to the modern era, one where users want access to data from any location and any device.

In this use case for object storage, the speed of ingesting data, or even retrieval, is not a primary concern. Remember that most devices are accessing via broadband, or Wi-Fi. So object storage systems are easily able to keep up with those speeds. Reliability and durability are what matters, and object storage delivers.

Most importantly, the combination of FSS and object storage enables IT to eliminate the unauthorized use of public cloud services, commonly known as "shadow IT." Shadow IT puts an organization's data at risk while giving users what they are demanding: on-demand access to data across multiple devices with the ability to share that data with people outside of the organization. In this scenario, a private object storage system combined with FSS can eliminate the use of a public cloud, and that is an appealing option to many companies.

It is true that most organizations can't justify object storage just for FSS. However, when used in combination with the traditional archive and backup use, the argument for integrating object storage becomes very compelling because the FSS problem can be addressed with almost no additional investment in hardware.

OBJECT STORAGE AS A DATA LAKE

Another use case for object storage is as a data lake. It's helpful to think about a data lake as an archive, but one that is specifically designed for big data. However, a data lake has to support more protocols than just object storage. Big data is often generated by machines or devices, such as cameras and sensors, which continuously feed small amounts of data to the data center. Typically a data lake storage repository has to support CIFS and NFS and occasionally iSCSI. The good news is that many object storage vendors have added these protocols. The addition of multi-protocol support makes object storage an ideal data lake because a data lake also needs scalability and durability while being cost effective.

To be clear, object storage will play a key role in data centers of the future. In the future, we can expect storage infrastructures to consist of two tiers: an all-flash tier for very active production data, and an object storage tier for secondary data. This means storage pros will be using object storage for archive, backup and file sharing—essentially any data that won't benefit from flash storage. There are even some all-flash array vendors that are integrating with object storage so that their snapshot replication jobs are sent directly to an object store, providing a more real-time backup and recovery option for this most mission-critical data.

In summary, there is no question that object storage is ready to demonstrate how useful it can be beyond archiving, but it is also important to remember that it can fulfill its archive role better than ever. Today's object storage systems have the right protection scheme for high-density drives and are becoming more power efficient. But extending the investment in object storage to do more than just archive allows the system to pay huge dividends.

—George Crump

Object Storage vs. SDS, File Storage

OBJECT STORAGE IS suitable for mass quantities of unstructured data and can replace file storage and large, secondary storage arrays. The system assigns unique identifiers to enable objects to be retrieved without knowing their physical location. Potential advantages over file storage include more granular metadata stored with the object and better use of disk capacity, but object storage may be slower in some cases.

Check out these expert answers to key technical questions to find out more about the unique characteristics of object storage.

WHY CHOOSE OBJECT STORAGE SYSTEMS OVER FILE STORAGE?

Object storage systems have been promoted as "the next big thing" for nearly a decade, though they have yet to develop a sufficiently compelling rationale to displace conventional file system-based storage in most environments. However, as storage becomes increasingly populated with large and small files—so-called "unstructured data"—wrangling all of this data in a hierarchical structure of volumes, directories and subdirectories supported by complex structures of nodes and inode extensions is becoming more and more unwieldy.

In some cases, businesses report storing hundreds of millions of very small files in a single directory used to build a webpage. In other cases, firms have digitally recorded medical images or surveillance video as files that need not only to be searched for and retrieved quickly, but modified with the addition of highly granular metadata to enable their efficient use. In short, the file system is beginning to fray at the edges, as storage requirements change.

Object storage involves the replacement of traditional file system structures with an object-oriented framework. Basically, instead of organizing files in a hierarchical file tree, data objects and files are organized into containers or "buckets," each with their own unique ID or key. Access is made simply via key or ID. Most metadata about the object or file is stored with the object or file, reducing the metadata management burden found in most file systems.

What exactly is object storage? It is a storage architecture that addresses and handles individual units of storage. It then assigns the objects a unique identifier, so the objects can be retrieved without knowing their physical location.

Basic object storage systems are simple to implement and operate, and rather minimalist in terms of functionality. Users can store, retrieve, copy and delete objects, and specify which users have what permissions to perform which functions. Increasingly, World Wide Web Consortium REST standards are being paired with object storage to facilitate integration with other applications and traditional file system-based data access methods. RESTful APIs and primitives are used to interact with containers and objects, enabling the use of external search engines—in the case of popular public cloud object stores—and other application software functions to interact directly with objects—in the case of some video editing systems.

Object storage systems also let users define the metadata stored with the object to a much greater and much more granular extent than is possible with most file systems. This is very good news for archival applications, big data analytics and in other cases where data volumes are large and detailed metadata searching can yield great value.

What exactly is object storage? It is a storage architecture that addresses and handles individual units of storage.

Object storage is also thought to make better use of disk capacity than a traditional file system, since space is not reserved for incomplete files and because journals of directory paths and file names typically consume little space on disk. In fact, in the more elegant implementations of object storage systems—such as Caringo Swarm—only objects are written to disk, while a small stateless journal is maintained in RAM, which considerably speeds up operations.

With control over object structure and metadata, better use can be made of data protection services. Data copy operation can be applied to objects that actually require multiple separate copies based on metadata-centric policies. Files that change less frequently may be delegated to processes such as erasure coding rather than mirroring.

Some vendors are working to find ways to organize better data in caches and buffers by recoding it initially as an object, then using object metadata to assist in cache management. Amazon has implemented object storage in its Simple Storage Service, or S3, and supports it with specialty REST-based protocols that have become as much a part of the storage interconnect lexicon as Fibre Channel, iSCSI or NFS.

Remember, if you are storing data to cloud services, chances are you are already using object storage systems today. If you aren't, it's time to consider how object storage can benefit you.

WHAT ARE THE MOST OBVIOUS OBJECT STORAGE USE CASES FOR MY SYSTEM?

Object storage is really software. It's a software layer that sits on a commodity x86 server, which is why a lot of vendors call it software-defined storage. But it sits on an x86 server, takes advantage of embedded storage inside that server and creates a node.

And then that software interlinks multiple nodes to create an object storage system across Ethernet. Now, object storage adds a lot of metadata to every object. In fact, you have a lot of flexibility in what you can describe about each object. And the metadata is actually stored with the data versus, let's say, a file system where the metadata is separated from the data and is usually put in some kind of database.

Now because of that extra layer of processing—more metadata—you're adding latency. You're processing between the application and the actual media you're writing on. That means you're adding latency, and whenever you do that, it's not as fast as other types of storage.

In fact, it tends to be a little slower than let's say, file storage, so it's definitely slower than SAN storage. There are ways to speed that up, but generally speaking, the best object storage use cases include archiving data, data backup and secondary data applications, but not applications that you will be accessing frequently or any kind of interactive or transactional type of application.

Replacing large, secondary storage arrays or file-based storage arrays are also object storage use cases. You can replace tape libraries because it makes a highly resilient, highly durable technology for both active archive and even a passive cold archive.

Here are the four chief things object storage can provide:

1. Low cost per GB of storage
2. Exceptional scalability (into the zettabytes)
3. High data resilience and durability against disk, node, or site failures, as well as bit rot
4. Easy tech refreshes

—Jon Toigo and Marc Staimer

When and How to Implement an Object Storage Platform

When deciding whether to implement an object storage platform into your environment, it makes sense to first outline what kind of data you're storing and how it's typically used. I would suggest you start by answering the following questions:

- Is it time to develop a genuine content store? In other words, do I need to keep a lot of data online for compliance or other regulatory reasons or for its historical value?
- Am I dealing with massive amounts of data that is overwhelming my file shares? Does this data become inactive shortly after it is created but stays on primary storage for long periods of time? When I need it, does it need high-speed access or would slower access be OK?
- Do I use backups as my archives today? Is that causing backup window issues? Are most of my applications disaster recovery (DR) protected with geographic separation?
- If not, do I want them to be? Do I have a large amount of content that is born static, such as photos or videos, that needs to be kept online for extended periods of time? Does my company have a desire to run serious analytics on this data today or tomorrow?

If "yes" is the answer to most of these questions, then you need to seriously look into object storage. Given the characteristics of object stores, it is easy to see why the use cases that have surfaced to the top include content stores, long-term archives, the back end of backup applications, backups with geographic separation for DR purposes and Web 2.0 applications. Web 2.0 apps had the distinct advantage of being written for object storage from the get-go. But then a large majority of these were written by the likes of Facebook, Twitter, Google, eBay and others for their own use and they developed their own object storage architectures that are not available to the outside world. Fortunately, many vendors specialize in object storage targeted directly at enterprises.

OBJECT STORAGE PLATFORMS: TYPES, VENDORS THAT PROVIDE THEM

You essentially have four choices:

1. Purchase a fully functional object storage platform from a variety of vendors today.
2. Purchase object storage software and install it on hardware (servers and storage) of your choice.
3. Install software on select file and block storage arrays that add an object interface to existing storage.
4. Use a gateway product that interfaces your existing application to a public cloud.

Fully functional object storge is available from EMC (Atmos-based), Cleversafe, Compuverde, Data Direct Networks (DDN), Dell (DX, based on Caringo), HP StoreAll, NetApp StorageGrid, Quantum Lattus (Amplidata OEM), Scality, Tarmin and others. Object storage software (or a virtual machine version) is available from most of these vendors as well. Open source software (OpenStack Swift, Ceph, Gluster) is available for free in a typically open source, unsupported manner. But it is also available in commercial, fully supported versions, from the likes of Red Hat and SwiftStack.

Software in option three (above) is available primarily from major players such as EMC Isilon or ViPR, HDS and HP 3PAR. The fourth category is interesting in that many backup and archival platform vendors have modified their software such that the back end can be a public cloud, such as Amazon Web Services or Microsoft Azure. In this case, you essentially get the benefits of an object storage platform without having to build a system yourself. It may be the best way to get in the game, as you learn more about the capabilities and limitations of object storage.

A large number of small backup and DR

HOME

EDITOR'S NOTE

OBJECT STORAGE POISED TO BREAK AWAY FROM JUST ARCHIVING

OBJECT STORAGE VS. SDS, FILE STORAGE

WHEN AND HOW TO IMPLEMENT AN OBJECT STORAGE PLATFORM

players, too numerous to list here, have sprung up in the past three years that specialize in these areas and use the public cloud as the repository. Of the major players, NetApp offers AltaVault (formerly SteelStore) appliances that provide deduplicated backup data on premises for immediate restores and uses the public cloud back end for storing older backups and to enable DR in the cloud or at a third site.

IBM also offers a way for backups to be stored in private or public clouds (including SoftLayer), via the IBM Cloud Managed Backup service. Microsoft, via its acquisition of StorSimple, has an appliance that sits in the data center and presents an iSCSI interface to the application while optimizing the data, which includes performing all the protocol conversions for Microsoft Azure. There are no changes necessary to the application in this scenario.

If you choose option one or two, the work involved is not trivial. You will need to decide which applications will run on object storage and how these applications will be modified to make REST-based calls to object storage. If you have no control over changing the source code for these applications, your options are then limited to using a gateway. However, if you do have the ability to modify these applications, then go ahead and survey the object storage products in the market to see which ones make sense for you.

START WITH THE PUBLIC CLOUD

My suggestion is to start playing with the public cloud first before embarking on building a large private cloud. Backup applications lend themselves to such ideas. Perhaps start with those. In that process you will end up getting off-site DR for "free." Then consider large content stores, perhaps using a gateway so that no change to the application is required.

When you do modify the application you will be able to use the metadata "magic" that object storage enables. And that would open up all kinds of opportunities for analysis that you never dreamt were possible.

Of course, if you are developing a Web 2.0 application yourself, I say you dive right into implementing full-blown object storage right from the start.

OBJECT STORAGE CAN NO LONGER BE IGNORED

It is time to start seriously exploring object storage if you haven't already. Exactly what you select and how you go about implementing it varies.

I suggest you let the use case determine the best method and start small. If possible, learn the idiosyncrasies of object storage before you make the big strategic decision that will last you for five years or more.

Another alternative is to survey the unified offerings, mostly from the larger legacy vendors, where you get all three methods of access (file, block and object) and don't have to worry about building a separate object storage box. Keep in mind, however, that such a unified system ultimately has either file or object underpinnings and will perform, cost and scale accordingly. But for convenience it is unbeatable. —*Arun Taneja*

GEORGE CRUMP *is president of Storage Switzerland, an IT analyst firm focused on storage and virtualization.*

MARC STAIMER *is the founder, senior analyst and CDS at Dragon Slayer Consulting in Beaverton, Ore.*

ARUN TANEJA *is founder, president and consulting analyst of the Taneja Group, an analyst and consulting group focused on storage and storage-centric server technologies.*

JON TOIGO *is the CEO and managing principal of Toigo Partners International and chairman of the Data Management Institute.*

STAY CONNECTED!

Follow @SearchStorageTT today.

When Implementing Object Storage Makes Sense
is a SearchStorage.com e-publication.

Rich Castagna | *VP Editorial*

Ellen O'Brien | *Editorial Director*

Ed Hannan | *Senior Managing Editor*

Andrew Burton | *Executive Editor*

Sarah Wilson | *Site Editor*

Dave Raffo | *Senior News Director*

Linda Koury | *Director of Online Design*

Neva Maniscalco | *Graphic Designer*

Jillian Coffin | *Publisher* | jcoffin@techtarget.com

TechTarget
275 Grove Street, Newton, MA 02466
www.techtarget.com

About TechTarget: TechTarget publishes media for information technology professionals. More than 100 focused websites enable quick access to a deep store of news, advice and analysis about the technologies, products and processes crucial to your job. Our live and virtual events give you direct access to independent expert commentary and advice. At IT Knowledge Exchange, our social community, you can get advice and share solutions with peers and experts.

COVER ART: FOTOLIA