File Access for Object Storage

Tom Coughlin, Coughlin Associates, Inc.
26 November 2019

All the data that drives our civilization, records our history, and entertains us is kept in digital storage. The amount of data created and stored keeps growing due to the advent of the digital economy, sensor networks, and the Internet of Things (IoT), big science, astronomy, geological exploration, use of mobile devices, connected vehicles, and cameras with higher resolution, higher dynamic range, and higher frame rate images and video. All of these contribute to the generation of big data and the consequent use of artificial intelligence to understand and monetize it all.

To be useful, this ever-increasing quantity of data must be stored, processed, and distributed. Organizations must continuously evolve their management approaches, and address important considerations including business needs, cost, complexity, performance, security, availability, compliance, backup, and disaster recovery. Data must be managed and secured, while at the same time shared with disparate parts of an organization, anywhere in the world.

Achieving these goals generally requires maintaining legacy and current applications and workflows while adopting new paradigms that exploit a variety of storage devices, systems, and architectures.

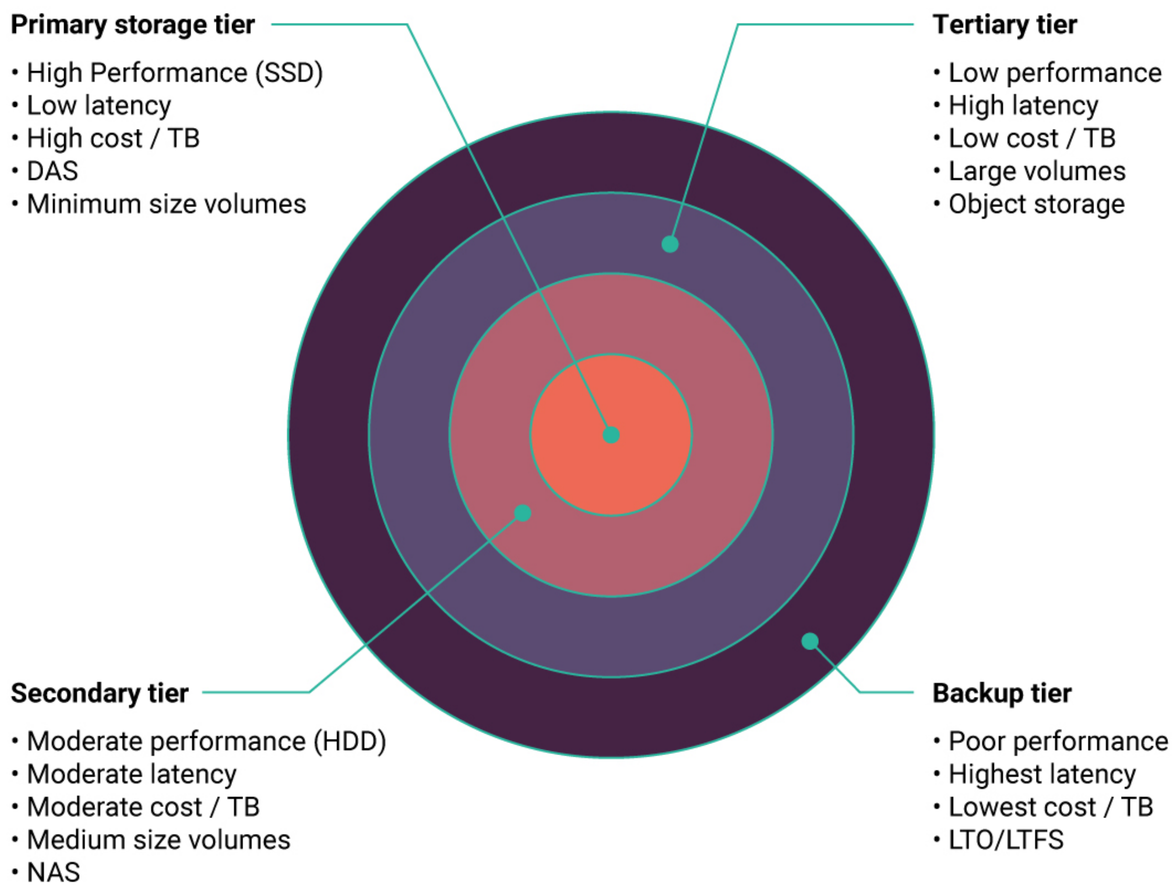
The market today offers a variety of different visions and approaches to solving these data-related challenges, most suited to just a particular use case, and many using proprietary formats, additional hardware, and with an underlying philosophy of "cloud first, on-premises second." This concept bears further consideration given the local generation of data and the need for low latency data access.

Storage Devices and Storage System Architectures

Today, digital data is stored on a variety of storage devices and architectures with different performance characteristics and pricing to match. The choice of what device to use for which application at a given point in time depends upon trade-offs between frequency of access (performance) and file size (capacity cost). These devices include solid state drives (SSDs) that use non-volatile NAND flash memory chips to store information, hard disk drives (HDDs) that store information on rotating disks and magnetic tape where information is stored on magnetic tape cartridges.

SSDs provide the lowest data latencies and these are the fastest generally available digital storage devices. HDDs have higher latency but are much less expensive than SSDs per TB. An array of hard disk drives with a large storage capacity can even work together to provide system data rates that equals that of an SSD storage system, although latencies will be higher for the HDD storage system. Magnetic tape is the least expensive storage media and consumes the least energy, since magnetic tapes don't consume energy unless they are reading or writing. However magnetic tape has the highest latency and thus has the longest delays in providing data, once requested.

Often, particularly in large data centers, SSDs, HDDs and magnetic tape may be used together to store data in different types of storage devices depending upon how often that data is accessed. In such a hierarchical storage system, the most frequently accessed data is in the SSDs. The HDDs store data that is a little less current, and magnetic tapes stores data that has long term value, or stored to meet regulatory requirements, but doesn't need fast access.



For example, concepts such as active archiving are driving the use of HDD storage for more frequently accessed "archived" data, supplementing tape for long term archives. Consequently, by 2024 we expect about 52% of archived content to be in mostly HDD-based near-line and object storage, up from 43% in 2018[1].

Data is stored on these storage devices as data blocks. When these storage devices are built into storage systems, they can access the data as blocks and transport these blocks to the connected host devices, where they can be reconstructed as data files. These data files combine blocks of data and include stored metadata that allows organizing the files into directories and sub-directories.

[1] 2019 Digital Storage in Media and Entertainment Report, Coughlin Associates, https://tomcoughlin.com/product/digital-storage-for-media-and-entertainment-report/

External storage devices or storage systems can be directly connected to individual computers and not shared with other servers in a network. An individually connected storage system is called direct attached storage (DAS).

If a block level storage system is connected to a network and is shared between multiple servers on the network, this connected storage system is called a Storage Area Network (SAN). SANs require a file system in the host that reassembles the blocks into files. A file system can also be built into the storage system, allowing the blocks of data from the storage devices to be assembled into files before they are shared with connected hosts. Such a connected storage system with direct file access (e.g. with NTFS) is called Network Attached Storage (NAS).

However, these traditional ways to store and share data have their drawbacks. File-level storage systems (NASs) are easy to implement and use and are generally less expensive storage options. However, their use becomes more difficult as the total number of files increases. Block-based networked storage systems (SANs) provide fast consistent movement of data. However, high performance SAN infrastructure is generally more expensive than NAS.

Object Storage and the Cloud

In order to store large amounts of unstructured data, object storage was developed. In object storage, data is stored in isolated containers called objects. Objects have a global unique identifier (GUID) and are stored in a flat rather than a hierarchical model. The objects can be accessed using this GUID, which makes finding data in object storage easier. Object storage can thus scale to hold a much higher number of objects than files in a file system.

Scaling of an object storage system only requires adding more nodes to a storage cluster. The flexibility and scalability of object storage has made this type of storage popular for large storage repositories, such as in large data centers (cloud storage is usually object storage). Object storage is becoming increasingly important as the number of data objects increase. For instance, overall object storage capacity is expected to grow about 5.6X between 2018 and 2024[1].

Object storage keeps metadata with the data and can pair objects with specific applications. Objects can be moved to different areas of storage and can be deleted when you no longer need them. Object storage can offer advanced data protection techniques like self-healing, automated replication, including replication between geographically dispersed data centers, high availability and improved search capabilities.

In addition, since each object in an object storage system has a unique identifier, data can be found and recovered from a local server or a remote server in a data center (the cloud). The table below gives some details about the difference between hierarchical file storage and object storage.

	Hierarchical Storage	Object Storage
Price	$$$	$
Data type	Structured	Semi Structured + Unstructured
Typical use	Millions of files/Terabytes	Billions of objects/Exabytes
Protocols	Local file system, SMB, NFS	HTTP REST API
Optimized for	Fast random access Reads and writes on small records, and ability to modify data in place.	Capacity Scalability

	Hierarchical Storage	Object Storage
	Storage for hypervisors File operations – move, rename, copy, etc Hierarchical browsing	Geo-distribution Web accessibility Storage density Searching
Typical apps	Word processors, content creation apps, CAD systems...	Social media, video, photos, log files, web content, large data sets
Approach	**File System Entry** = Datapath + file metadata - data stored at fixed location	**Object** = GUID (Global Unique Identifier) - data + associated metadata stored somewhere

Object storage platforms usually rely on Representational State Transfer (REST) Application Program Interfaces (APIs) for access to data. REST is a stateless, client-server architecture that is usually implemented with HTTP. Commonly used REST APIs are those of Amazon Simple Storage Services (S3) and Open Stack Swift. These APIs make objects accessible via HTTP and facilitate management functions for authentication, permissions and file properties.

Generally, local object and cloud object storage are not compatible with traditional operating systems (OSs), such as Windows or Mac OS. Incompatibility of common file systems with object or cloud storage can be an issue for many workflows. Since many legacy and current applications are based upon file systems, these applications cannot directly access data in object storage. This is a challenge for creating a modern hierarchical storage system that can be managed and controlled with a single system that incorporates SSDs, HDDs, tape, object storage and the cloud.

Hybrid and Public Cloud Storage

Some common uses of object storage are storing big data, access to data using web apps and maintaining backup archives. Object storage is often implemented in large data centers but can also be built into local storage systems. These local object storage systems can be connected to the internet, providing a restricted access private cloud. It is becoming increasingly common to store data in both private and public clouds with data moving between these clouds (this approach is often called multi-cloud).

Hybrid and public cloud storage are useful for many applications. These include enabling workflow collaboration between remote facilities with a shared dataset and generally remote access to such datasets. Public cloud access enables the use of specialized services that are expensive to build on-premises but can be leased as public cloud services such as AI training, content delivery networks and video rendering.

Many people are also copying data or archiving data into the cloud for disaster recovery or long-term retention purposes. Hybrid clouds also enable cloud bursting to utilize public cloud resources to scale operations when local resources become saturated.

Many organizations use public cloud service leasing to turn capital expenses into operating expenses or to minimize the need for local IT support. Other services supported by hybrid and public cloud include data replication and disaster recovery, live server migration and expansion, performance tiering, surveillance video storage, cloud archiving and tiering, geo-synchronization of data, file sharing, migration of data from tapes to the cloud and remote office access with a shared dataset.

Much data will continue to be generated outside of the cloud – on edge devices, on-premises, or from the multiple sensors in Internet of Things (IoT) devices (digital endpoints), including security cameras, field sensors and vehicles. Hybrid clouds will be commonly used for the foreseeable future; full cloud workflows will be few.

While the cloud offers near infinite scaling of storage, the migration of legacy servers and applications has proven to be slow, difficult and challenging for many users. Moving that much data to the cloud requires a lot of time and effort. In addition, once the data has been moved, retrieving it proves to be expensive due to egress charges. There is a need for a new hybrid architecture that can facilitate the integration and migration of large-scale Windows NTFS cloud deployments while reducing cost and complexity.

Many applications were designed to work with hierarchical file systems built with direct attached storage (DAS), NAS or SAN. These file systems have hierarchical indexes, containing folders, sub-folders and files. Files are accessed using a fixed pathname. File systems rely on specific addressees and locations. Consequently, it can be difficult to move data around without generating broken links.

Hierarchical file systems may also have less means to protect themselves from corruption. If a part of the file system hierarchy gets corrupted, it may not be possible to access some of the data. Businesses need a robust way to allow file access to cloud-based object storage.

IT managers need to find ways to cost effectively manage data placement, storage consumption and cost. To do this organizations currently deploy various systems – dedicated cloud gateways, file sync and share, NFS/SMB, object abstraction layers, backup/archive, geo sync, disaster recovery, etc. The advantages of these approaches stem from their singularity of purpose, the disadvantages often include no system-wide integration, multiple vendors, proprietary formats, vendor lock-in, disrupted workflow, app incompatibility etc.

For many organizations who want to access the power of cloud storage, their pathway involves the use of a cloud gateway. A cloud gateway is a hybrid cloud connectivity platform and is an example of a Platform as a Service (PaaS). The gateway securely connects local file-based networks with multiple cloud service providers, creating a hybrid cloud that includes local storage and public cloud storage.

There are many cloud gateway products available in the market. Most gateway servers present the object storage as a new mount point for the file system which must be integrated into existing processes and workflows. In this case, performance and compatibility to the SMB/CIFS/NFS protocol is often sub-optimal because of the added overhead and complexity this creates. These cloud gateways differ in their ability to provide ready access to local and remote data, but generally require additional dedicated hardware, and are built on the idea that data permanently moves away from file system storage to object storage.

Some important requirements for a useful cloud gateway are that it work with established cloud providers (Amazon, Azure, IBM, Google, Backblaze, Wasabi, etc.), including integration with any storage layers in these public cloud offerings and cloud storage management capabilities. It should also help keep servers in synchronization, whether local or across the globe, so content written on one server will appear as a local file on all the other servers. A cloud gateway should also help organizations transition their legacy infrastructure to the cloud at their own pace.

Storage Challenges Summary

- Most the world's file servers use Windows OS and NTFS and support legacy Windows-based applications and workflows
- Increasing data loads require new approaches to storage management
- Cost, scale, and flexibility make object and cloud storage compelling for IT managers
- Object and cloud storage are incompatible with these legacy applications and workflows
- Full migration to cloud and service-based infrastructure is disruptive to business and operations and requires re-training

If all these challenges can be met by one storage management system it can help transform a company, providing it with competitive advantages, reducing costs and complexity while increasing capabilities.

Tiger Bridge

Tiger Bridge is a software-only data and storage management system, cloud gateway and HSM manager that allows hybrid and full cloud operation and provides many other services. It is a policy-based connector that blends NTFS with cloud, NAS and tape into a single global file system. Tiger Bridge is unique in that it enables conventional file level access to object storage. Bridge also enables continued file-based access to data with an existing file system, utilizing current workflows using existing infrastructure, and replicating and tiering data across multiple locations. Tiger Bridge is a single storage management product that enables multiple use cases, such as file synchronization and sharing, capacity expansion, HDM, disaster recovery and others.

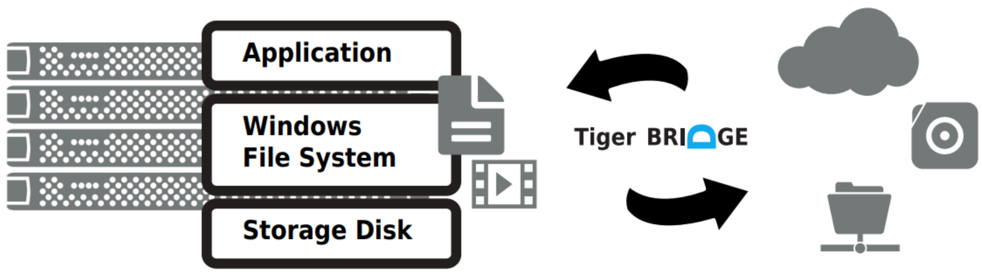
Tiger Bridge runs directly on a Windows server and integrates into the existing NTFS file system at the kernel level. It ensures strict adherence to Active Directory security. When connected to a target over a RESTful/S3 API, data is encrypted directly on the server via https before being sent to the cloud to ensure data security.

When files are migrated from the local server to the object storage, stub files are created to represent any files that were moved. These stub files contain all metadata from the original file and are accessed by users and applications just like any other file. In many cases, the stub file provides enough metadata to allow many processes to function normally despite the absence of file data, lowering latency. When a stub file is called, the original data is automatically and transparently restored back to the local file system and handed to the application.

Tiger Bridge integrates with the main cloud providers' platforms, including intra-cloud tiers such as hot, cool, or archive, and can save on egress charges by browsing local folder stubs rather than the full cloud file.

Some Tiger Bridge Applications

Several applications are enabled by file level access to object and cloud storage. Tiger Bridge provides data and storage management across multi-cloud hybrid storage deployments as shown below.

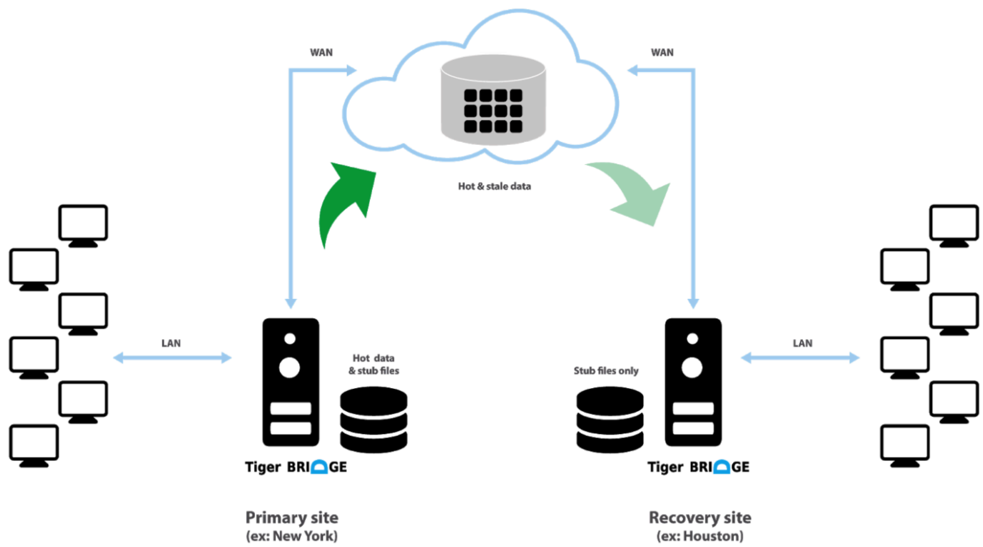


Tiger Bridge also supports DAS, tape, on-premises object storage as well as hot, cold and archive tiers in Azure and AWS. It dynamically tracks locally stored data, keeping active files local so they can be accessed at the highest data rate and lowest latency. Inactive files are migrated to lower cost storage tiers (e.g. in a public cloud) either automatically or manually. Small stub files are maintained on local storage where they are visible to applications and the file system. When one of these stubs is opened, data is automatically retrieved on demand.

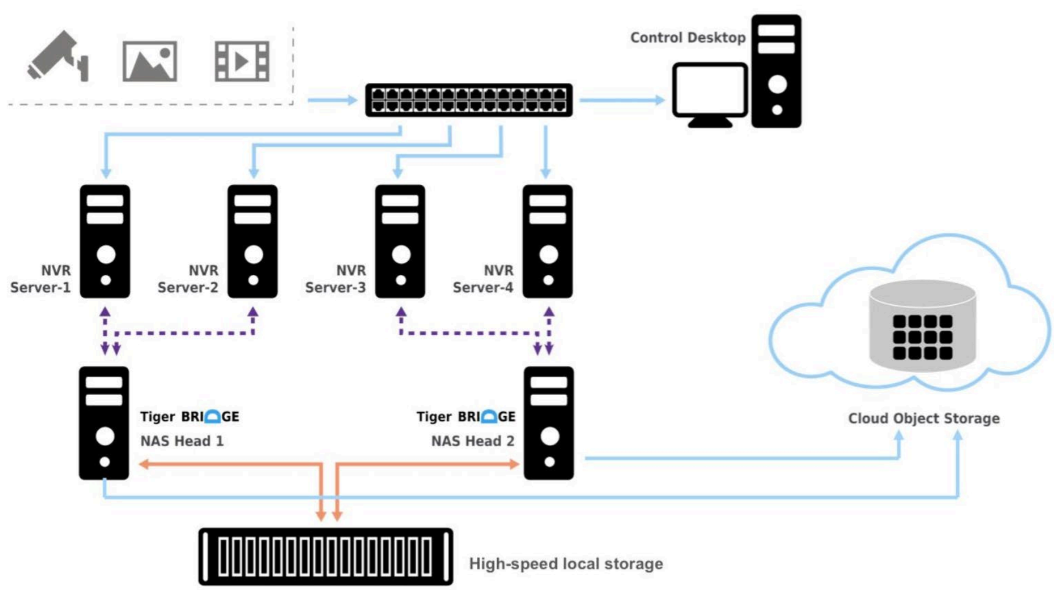
Because of the migration of inactive data, free space is made available locally. The stub files make sure that file-based workflows can access files stored in the cloud and, consequently, petabytes of data are available for local and collaborative projects. Tiger Bridge provides storage optimization, hierarchical storage management, backup and archive, geo-location, file synchronization and sharing.

Tiger Bridge can provide downtime disaster recovery. Because of its Geo-synchronization features, stub-files are already created and ready to run on a DR server. Tiger bridge running on the primary site automatically replicates newly created content to the cloud. Corresponding stub-files appear on the Tiger Bridge file system that is running at the recovery site (if both sites are connected to the same object storage).

Since the recovery site can immediately access the file metadata and thus access the file data, a complete restore of all data from the cloud may never be needed at the recovery site, reducing the size of the storage infrastructure required at the recovery site. The image below shows this Zero-Downtime Disaster Recovery application.



Tiger Bridge can also provide low-cost storage of video surveillance data. It is now common to find thousands of cameras in a single location sending all their data feeds to a few recording servers. The bulk of this video data has no value, until critical evidence needs to be retrieved from the videos. By presenting the object storage as an extension of the local NTFS file system, Tiger Bridge enables surveillance applications to interact with object storage transparently. A visualization of this application is shown below.



Tiger Bridge allows traditional network video servers to write video feeds to a local drive (or SAN) before being replicated to the object storage. When this object storage is off-premises (in the cloud), this buffer ensures that all video evidence is recorded prior to being encrypted and transferred to the cloud. Since Tiger Bridge runs directly in the server's memory, https encryption takes place before any critical data goes on the network, ensuring that data is secure. Once in the cloud, virtual machines can manipulate the data, performing tasks such as facial and object recognition.

Summary

Digital storage requirements are growing, driven by the growth in the number as well as the size of digital objects. Industries require strategies to cost effectively manage as well as protect the data from their operations and workflows. This has led to the development of object storage that can scale to huge numbers of data objects and object storage is the basis of most online (cloud) storage.

However, many popular applications are based upon the use of files, with their associated hierarchical data structure. Using these applications with object storage requires a cloud gateway that can provide file level access to object storage in the cloud. Tiger Bridge provides a cloud gateway data and storage management system that blends NTFS with cloud, NAS and tape into a single global file system. In addition, Tiger Bridge enables useful approaches to backup, archive, and disaster recovery using geo-synchronization and various levels of migration of data to the cloud.

About the Author

Tom Coughlin, President, Coughlin Associates is a digital storage analyst and business and technology consultant. He has over 37 years in the data storage industry with engineering and management positions at several companies. Coughlin Associates consults, publishes books and market and technology reports and puts on digital storage-oriented events. He is a regular storage and memory contributor for forbes.com and M&E organization websites. He is an IEEE Fellow, President of IEEE-USA and is active with SNIA and SMPTE. For more information on Tom Coughlin and his publications and activities go to www.tomcoughlin.com .