The Fastest Global Cloud File System on the Planet

Table of Contents

Introduction

Legacy NAS storage was never designed to deal with the sheer volume of data enterprise is now producing, and it's clear that cloud storage is the only option with the capacity to handle it. The cloud has no limit to its storage capacity, it's cost effective, secure and accessible. It's no surprise then, that according to IDC, over 90% of all enterprises are already using 4 or more clouds.

However, adopting the cloud as a storage tier comes with immense complexity, driven by the lack of supportive technology to replace the network server functions you've always taken for granted. That is, while your file storage in the cloud looks like it always did in your NAS storage, the files themselves behave very differently.

As a result, the workflows your enterprise has spent years developing to support your business processes and to allow your teams to work as cohesive units are forced to change. This has a particular impact for organizations that have geographically distributed teams, who all need to access files centrally stored in the cloud.

Worse, legacy applications are often unable to be migrated to the cloud without rewriting to allow them to function, resulting in significant development cost and delay.

Panzura allows the enterprise to seamlessly move their data storage into the clouds of their choice by enabling enabling global file sharing, cloud-integrated NAS and data protection, allowing distributed users to work as if they're still in the same office. We take care of cloud-based file operations and file management to deliver the high performance NAS experience you need for every user you have, regardless of where they're located.

At the same time, Panzura consolidates storage and infrastructure to achieve a reduction in footprint of up to 70%, while also delivering active archive and disaster recovery, along with military-grade security including immunity to ransomware and other cyber threats.

Let's dig in to what Panzura does, and what it means for your business as you look to the cloud for the future of your enterprise data storage.

The Data Storage Challenge

The primary technology for storing unstructured enterprise data today is high-performance, high-cost network-attached storage (NAS). With this technology, the standard method for addressing data growth is to add additional expensive filers containing spindles or SSD based flash arrays. In addition to the hardware costs, more disks/ SSDs mean more datacenter space, power and cooling requirements, and added off-site capacity for replication.

Because capacity expansion takes time, IT managers must try to forecast storage needs and forward provision to accommodate these potentially incorrect forecasts. Unanticipated spikes in storage demand send IT managers scrambling for added capacity and overly aggressive forecasts result in investment in idle capacity. As enterprises adopt newer cloud based technologies like Dropbox and Box to meet the increasingly mobile workforce, these services lead to even more disparate islands of storage to be managed. Additionally, as different departments independently launch projects into the cloud for development, test and simulation workflows the enterprise is evolving into an ever increasingly complex multi-cloud vendor ecosystem.

When multiple sites are taken into consideration, using standard NAS can often result in over-provisioning at some sites and under-provisioning at others. Maintaining multiple sites also results in storage "islands", where data at one site is not visible to users and applications at other sites. The only way for users at one site to view files created or edited at another site is to save copies of those files to their own location, resulting in data sprawl, duplication of files, the commensurate overspending on expensive storage, and significant version control management challenges. Since backup and archiving also occurs locally, duplicate copies on islands of storage means greater storage needed for data protection as well. As an alternative to customer-owned NAS, some vendors offer what they call Cloud NAS. Unfortunately, almost all of these solutions suffer from limitations in performance, scale, or both - and none are yet enterprise-class.

What about the nature of the data itself?

On-site NAS today supports both structured and unstructured data storage. High-performance applications that utilize structured data, like databases, require block storage in order to provide adequate response times and synchronous replication speeds - storage that often uses high-performance drives or SSD storage. iSCSI is a common interface used for block storage to provide direct disk access to these applications. But, applications using unstructured data (which represents 80% of data under management) store it in filesystems (which provide the structure), not as blocks, and usually use interfaces like SMB or NFS unless the applications are rewritten for iSCSI. For the most part, block storage interfaces like iSCSI do not lend themselves to applications using unstructured data.

In addition to compatibility, block storage interfaces suffer other shortcomings relative to file-based protocols. Because block-based applications are primarily limited to single-node storage targets, their ability to scale can

be quite limited compared to file-based storage systems spanning multiple servers. And unlike unstructured data, replication of structured data requires that the disks at the replication site be identical to those at the source site.

Since applications using unstructured data are disk agnostic and can address multiple servers, they are particularly suited to solutions with SMB or NFS interfaces that target scalable storage. In particular, object storage. Thus, for optimal storage performance and cost control, administrators devote special attention to tiering storage according to the specific requirements of each class of user or application, as shown below.

	Block	**File**
Example Application Interface	iSCSI	Standard SMB/NFS
Application Types	Databases, Exchange	Departmental Shares, Home Directories, Application Logs, Video & Images, IoT telemetry and sensor data, Productivity Applications, etc.
Share of Data	<20%	>80%
Scalability	Limited	Massive
Replication Storage Type	Identical	Mixed OK

The Data Protection Challenge

Data protection is comprised of archival, backup, and disaster recovery (DR). The primary purpose is to maintain access to data that is no longer regularly used or to be able to recover data if it is lost or corrupted. For archiving, the authoritative copy of the data may be stored off site and recalled when needed. With backup, the data remains in use, or at least kept in primary storage, and a copy is made and stored for retrieval if the original data is lost. The most common method for protecting data is using a software application to direct data to disk or tape for backup. Magnetic tape has been used for data storage for over 60 years and the technology has not changed all that much during that time. Tape is still in use due primarily to inertia and its perception as being inexpensive on a $/GB basis. Using tape, however, is very cumbersome, time consuming, and prone to error, making it a much derided medium for data backups and archiving.

With steep reductions in the cost of disk and the development of deduplication over the last decade, disk-to-disk archive and backup has gained more and more of the data retention share from tape. Disk targets range from removable (very slow) optical disk and commodity magnetic disk to specialized backup and archiving appliances. But all disk-to-disk backup still suffers from one or more of the following major drawbacks: high

cost, limited functionality, vendor lock-in, limited scalability, and cumbersome deployment and management. Sometimes disk-to-disk-to-tape methodologies are also deployed.

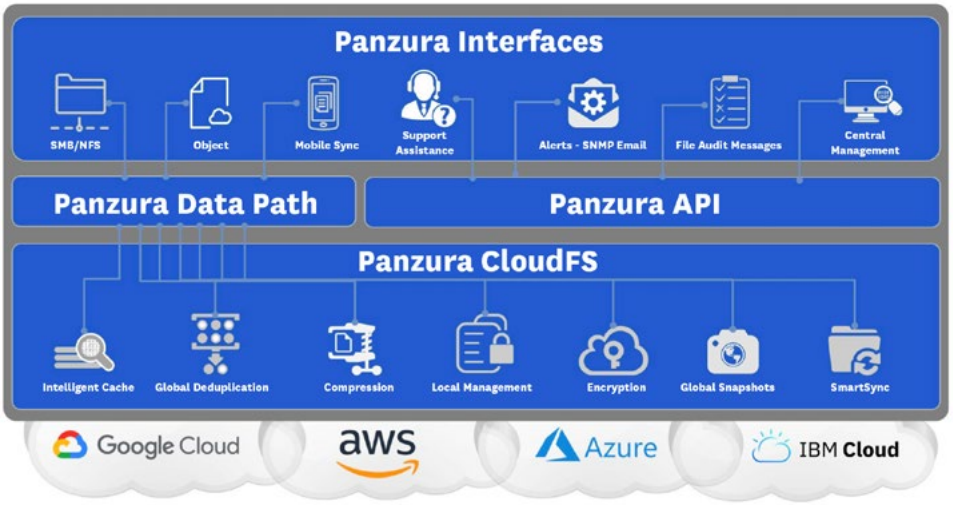
More recently, disk-to-disk-to-cloud data protection solutions have appeared, offering the scalability, availability, and consumption based cloud economy as a storage target. Theoretically, using the cloud is quite appealing but, in practice, integrating cloud storage into an established IT infrastructure can be problematic due to issues like latency, communication protocols, and data security. The resulting impact on restoration times due to limited bandwidth and highly latent links is particularly concerning.

DR (disaster recovery) can leverage both backup and archival while centering around restoring operations when a site fails. DR involves rebuilding site functionality as quickly as possible to minimize the impact on overall IT operations while maintaining business continuity. Rebuilding can occur at the same location or off-site at another location. Traditional reliance on tape, with its complex off- site logistics and slow performance, has made rapid tape-based DR unachievable. Replication (mirroring or moving backup data from one location to another) is a common but potentially expensive way to implement a DR strategy. This is especially true if your solution involves full hardware duplication, which doubles datacenter capital costs. DR planning is challenging, time consuming, and difficult to get right. For these reasons, DR is often put off or avoided as long as possible, with organizations adopting a “hope for the best” strategy.

Panzura CloudFS Overview

CloudFS is a distributed filesystem incorporating network acceleration technology, specifically designed to accommodate highly latent remote object stores, and able to overcome the limitations preventing enterprises from successfully integrating cloud storage into their infrastructure.

The Panzura architecture comprises four major component blocks: the Panzura Interfaces, the Panzura Data Path, the Panzura API and the Panzura CloudFS. Together, they provide a multi-cloud file services platform that enables high performance tiered NAS, global file collaboration, active archiving, backup, and DR across all enterprise locations.



Panzura supports public, private and dark clouds

The Panzura architecture enables the enterprise to consolidate their unstructured data and eliminate islands of storage.

File Based Storage

Panzura developed a high-performance file-based global storage platform for the cloud to address the 80% of current data that is unstructured. By supporting NFS and SMB transfer protocols commonly used by most applications, Panzura can plug into existing IT infrastructures without any changes and connect to all major cloud storage platforms, simplifying deployment and minimizing impact on operations. All data is managed under a single global filesystem, simplifying user interaction and system administration while tying into enterprise applications and targeting both local disk and the cloud.

Cloud Object Storage

Object storage, the typical storage system used in the cloud, breaks up data and stores it as flexibly-sized containers or chunks. Each chunk can be individually addressed, manipulated and stored in many locations - not tied to any particular disk - with some associated metadata.

Object storage can scale to billions of objects and exabytes of capacity while protecting data with greater effectiveness than RAID. In addition, due to the discrete scale- out architecture of object storage, drive failures have little impact on data and self-healing replication functions recover very rapidly (think weeks for large capacity legacy RAID systems). This combination of scale and robustness make object storage an ideal target for warehousing enterprise data.

Panzura interfaces directly with all major cloud object storage APIs and related storage tiers, avoiding vendor lock-in, and leverages object-based cloud storage as a data warehouse to provide scale and availability at a compelling cost structure.

Global Cloud File System

The heart of any storage system for unstructured data is the file system. Key file systems that have shaped the market include VxFS (Veritas), NTFS (Microsoft), WAFL (NetApp), and ZFS (Sun). A successful file system must be **highly scalable, high performing, flexible, and manageable**. NetApp built much of its success around WAFL and its ONTAP OS. WAFL combined RAID, the disk device manager, the file system, replication, and snapshots (limited per volume). Its primary target is HDD. ZFS took these elements, added encryption and deduplication into one stack. Additionally, it is massively scalable, natively targets HDD and SSD, but has no native cloud integration.

The Panzura CloudFS file system was engineered to closely manage how files are utilized and stored to provide seamless, high-performance, and robust multi-cloud data management. It improves on WAFL and ZFS while integrating cloud storage as a native capability.

Any user, at any location can view and access files created by anyone, anywhere, at any time.

The file system dynamically coordinates where files get stored, what gets sent to the cloud, who has edit and access rights, what files get locally cached for improved performance, and how data, metadata, and snapshots are managed. The structure of the file system has no practical limit for the number of user-managed snapshots per CloudFS. Panzura's innovative use of metadata and snapshots for file system updates, combined with unique caching and pinning capabilities in the Panzura Filers, which are deployed either locally or in the cloud, allows you to view data and interact through an enterprise-wide file system that is continually updated in real time. Support for extended file system access control lists (ACLs) empowers administrators to set file access and management policies on a per user basis.

Global Namespace

At the highest level, the Panzura global namespace is an in-band file system fabric that integrates multiple physical file systems into a single space and is mounted locally on each node. The entire global namespace has the root label of the distributed cloud file system.

As an example, the following 2 global namespace paths point to the same directory (\projects\team20) and are visible from both nodes as well as locally on nodes cc1-ca (California) and cc1-hi (Hawaii).

It is important to note some fundamental differences between the Panzura global namespace fabric and conventional global namespace (GNS) concepts. Conventional GNS architectures require a database process on each storage system (either in-band or side-band on a separate dedicated GNS system). These distributed databases own and manage all file system metadata transactions. File operations are intercepted in-band, processed and acted on by the distributed database instances before each file operation is allowed to complete.

Changes to file metadata anywhere within the GNS require complex out-of-band distributed database replication, synchronization, arbitration and resolution while simultaneously attempting to provide real-time access to the file with guaranteed consistency. The reliance on multiple distributed database processes could introduce complexities, in- band latencies and operation challenges that fail to scale at global multi-site levels. Examples of conventional GNS solutions are Microsoft DFS, F5 ARX and EMC Rainfinity. Additionally, these GNS architectures are somewhat limited in their ability to offer capabilities like global snapshots.

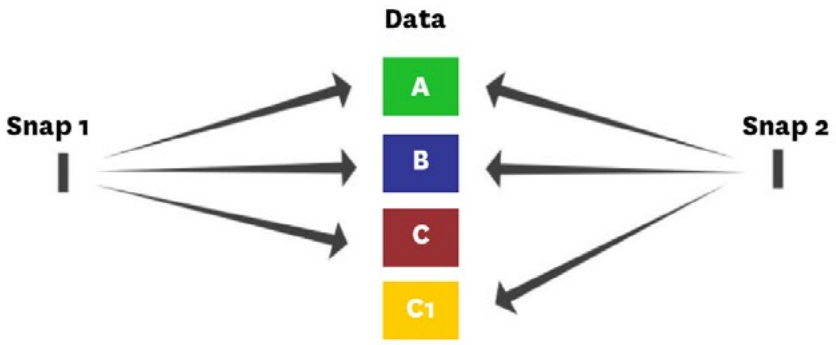
The Panzura global namespace has no reliance on underlying distributed databases and avoids common GNS limitations (e.g. speed, transactional data coherence, write order fidelity, open files, precision, in-band operation, global snapshots).

At a fundamental level, the file system fabric is the namespace engine. The global namespace is integrated into the metadata. Metadata is stored centrally in the cloud for durability in addition to being fully cached locally for enhanced performance. All filers in a single namespace or CloudFS synchronize updates to the metadata in parallel asynchronously every 60 seconds in a hub (cloud) and spoke (filer) configuration. This is further complemented by a peer to peer synchronization event that occurs in real-time when lock dynamically moves from one filer to another through the distributed global file locking.

This ensures that no two processes will be awarded locks at either the file or byte range level at the same time, thus avoiding write collisions. Furthermore changed blocks that may not have been written to the cloud yet are included in the lock exchange ensuring immediate consistency.

Panzura Snapshot Technology

Snapshots for Consistency
Snapshots capture the state of a filesystem at a given point in time. For example, if blocks A, B, and C of a file are written and snapshot 1 is taken, that snapshot captures blocks A, B, and C to represent the file.



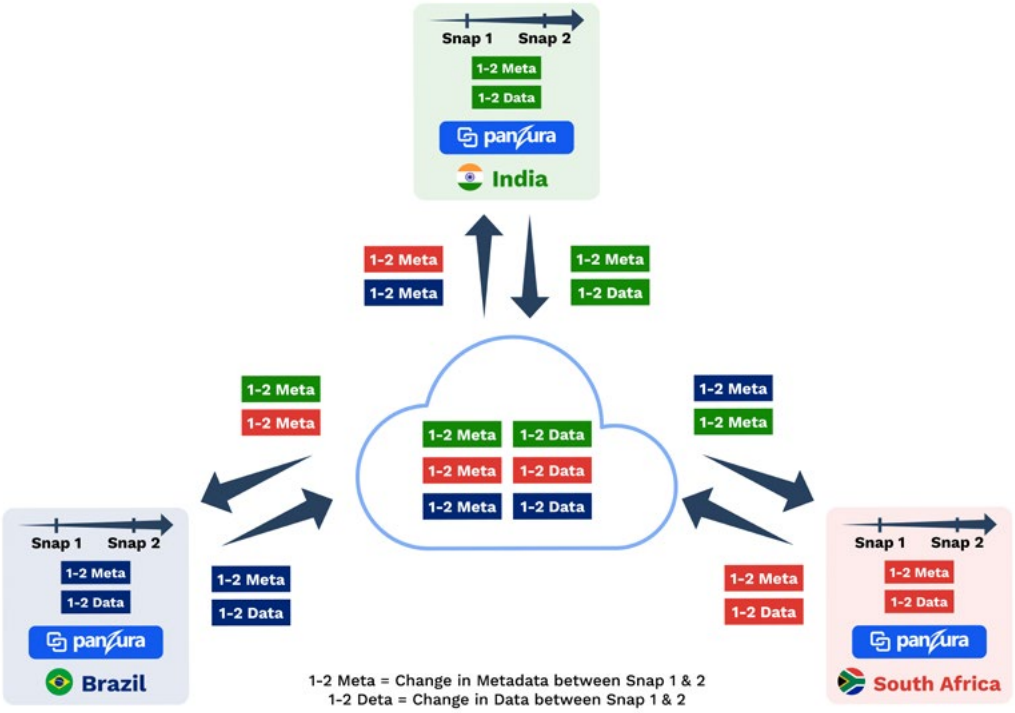
If someone then edits the file so that block C1 replaces C and snapshot 2 is taken, the data pointers in the snapshot file blocks A, B, and C now point to A, B, and C1. Block C is still retained but not referenced in snapshot 2. If someone wanted to recover to the original state, they can restore snapshot 1, then the system will point back to A, B, and C, ignoring C1.

By using snapshots for creating and saving an ongoing series of recovery points for different stages in data's lifetime, a consistent state of the filesystem can always be restored in the event of a data loss.

Snapshots for Currency
Panzura uses differences between consecutive snapshots both to maintain filesystem consistency as well as

to protect data in the filesystem. In a process called syncing, the Panzura filesystem takes the net changes to metadata and data between consecutive snapshots and sends them to the cloud. The metadata portion of these changes is retrieved from the cloud by all other Panzura Filers in the configured CloudFS, where they are used to update the state of the filesystem and maintain currency (see image below). This system updating occurs continuously across all filers, with each filer sending and receiving extremely small metadata snapshot deltas to and from the cloud in a hub and spoke configuration, using them to update the filesystem seamlessly and transparently.

For example, a controller in Brazil (red in the figure below) takes Snap 1 and then later takes Snap 2. The difference in metadata between Snap 1 and Snap 2 for Brazil is shown in red as 1-2Meta. The difference in data between Snap1 and Snap2 for Brazil is shown in red as 1-2Data. Brazil sends its 1-2Meta and 1-2Data to update the cloud, as do all other controllers in the infrastructure. Brazil also receives back metadata updates for all other controllers (shown as 1-2Meta in green for India and in blue for South Africa).



All of the changes in data and metadata are stored and tracked sequentially in time - should a data loss or corruption occur at the local filer or in the cloud, data can be restored to any previous state at which a snapshot was taken, without the need to follow a separate backup process.

It is important to reiterate that the size of these snapshot deltas (1-2Meta, 1-2Data) are exceptionally small relative to the data in the filesystem; thus they can be captured continuously and use bandwidth and capacity very efficiently.

The result is the Holy Grail of a global filesystem: a solution that requires almost no overhead but provides near real-time, continuous rapid updates across all sites for one global filesystem currency.

The key to a current global filesystem is accurate and efficient transfer of only the data needed to ensure the filesystem view of each Panzura Filer remain current. Every sixty seconds Panzura snapshot synchronization technology enables currency across a globally-dispersed filesystem with minimal overhead, providing local NAS responsiveness to a worldwide infrastructure.

Snapshots for Efficiency

In addition to synchronization, snapshots used to maintain consistency and currency, Panzura Filers also have no practical limit for user-managed snapshots. This category of snapshots allows users to recover data on their own by simply finding the desired snapshot in their inventory and restoring it. This self-service recovery greatly reduces demands on IT by allowing users to recover data on their own, without IT intervention. Policies around user- managed snapshots (frequency, age, etc.) are defined by IT administration.

For example, a Microsoft Windows user in India travels to Brazil and realizes she needs a file that she deleted 3 months ago. She directs her Windows Explorer to the local Brazil Panzura Filer and navigates to her snapshot folder, finds the date/time that corresponds to the filesystem view that contains the file she wants to recover, opens that snapshot, and navigates to the file or files she needs to recover, then just drags and drops the needed file(s) into her current filesystem location where she wants them restored. Within minutes, she has recovered whatever files she needs and can continue with her work, all without involving anyone from IT.

For ease of use, user snapshots have been integrated with the Windows Previous Version function allowing users to right-click on any file or folder and easily restore to any previous snapshot. IT administration can dynamically change snapshot policies as needed to satisfy data retention policies, balance frequency and duration for optimal system performance and user satisfaction.

Snapshots Benefits

Panzura snapshot technology provides three major benefits for the Global filesystem: consistency, currency, and efficiency. Continuous snapshots provide granular recovery points so that, in the event of a data loss, a consistent filesystem state can be restored with minimal disruption or delay.

Panzura snapshot technology provides all users in all locations with a current view of the entire filesystem.This is done by syncing all filesystem views globally in real-time, allowing users to experience cloud storage as if it were local, finally solving the key inhibitor to a true global filesystem.

By empowering users to recover their own data as needed, Panzura snapshot technology offloads a key aspect

of user support, freeing up time for strategic IT projects. The Panzura Filer brings the power of the cloud to enterprises without sacrificing the user experience.

Intelligent Caching at the Edge

SmartCache

Panzura CloudFS utilizes a user-definable percentage of the local storage as the SmartCache to intelligently track hot, warm, and cold file block structures as they are accessed. This form of caching dramatically increases the I/O performance of reads (and reduces cloud object storage access charges) by servicing them from local cached storage (both in memory and on persistent local flash) rather than from external cloud storage. The filesystem also buffers against variations in cloud availability to help maintain consistent read/write response times - performance AND availability at the edge.

SmartCache Policies

SmartCache is a dynamically managed caching technology that allows administrators to create intelligent caching policies based on defined rules. SmartCache policies provide a flexible method for the storage administrator to directly manage and influence the performance and availability of reads for explicit types of data via specific policies.

Caching policies provide two basic functions. The first function is pinned data, which keeps data available on local storage using flexible wildcard policy rules. Pinning is a forced action and executed against full files whereas SmartCache is a read-stimulated action executed against frequently accessed blocks within a file. Pinned data results in a 100% local read guarantee whereas SmartCache is deterministic based on previous I/O read patterns within the local filer. The second function provided by caching policies is Auto-Caching which automatically caches data locally based on defined rules. However, auto-cached data can be evicted for requested hot data, as needed.

The pinned, or auto-cached, data is a subset of the total SmartCache storage tier. Pinned data is considered high-priority cached data that is never evicted unless authorized by the administrator, whereas auto-cached (cached based on wildcard rules) or SmartCache cached (data blocks automatically cached based on observed usage patterns) can be evicted by the system if needed to make space for more frequently accessed data. The balancing of pinning and SmartCache is delicate as a pinning rule will force data blocks to be logically placed inside the SmartCache, consuming local space, which may affect the local cache utilization and efficiency in ways that the administrator may not have considered. Because pinned policies are of the highest priority and override caching rules based on observed behavior, careful attention should be given to those policies so as to not consume all of the local storage leaving little for actual hot data.

The Auto Pre-populate feature provides an even higher degree of automated caching capabilities called. If enabled, the filer will automatically pre-populate or pre- cache files based on ownership changes between filers in a CloudFS. This is particularly helpful in collaborative workflows where users at different sites are working on the same datasets. As the filer detects ownership changes between locations it will automatically cache data in the same directory in anticipation of user read requests on those files between sites.
The SmartCache policy allows administrators to define wildcard based rule sets. In this way, administrators can define matching rules for specific directories within the filesystem or even specific file types across the entire filesystem. For each defined rule the administrator can define caching actions for that given rule.

Rule actions are as follows;
• Auto Cache - This is the standard behavior of the filer. The filer will evict blocks of a file as needed to accommodate new data.
• Deny - When this action is selected, the creation of files with names matching the glob expression is not allowed for that filesystem.
• Pinned - Applies a 'last out' policy to data. It avoids evicting data unless the cache is full and new priority data is being ingested. Pinned data will be evicted only as a last resort after other data to maintain normal operations.
• Not Replicated - Causes the data not to be copied to the cloud. This creates an unprotected scratch or temp space and should be used cautiously since the data is persisted locally but considered temporary as it is not available in the case of a DR event.

Local Storage Usage

A portion of the local storage is allocated for SmartCache. This portion is configurable and is set to 50% by default. Over time and through general usage, the system dynamically populates the local cache with hot data blocks from all of the files being read by users and applications. The most optimal and efficient SmartCache configuration is to have most of the cache comprised of hot and warm blocks, with most cold blocks being evicted to the cloud. In this case, a high percentage of reads are serviced directly from the local cache rather than from the cloud. This is the optimal caching state, but is harder to achieve when more pinning rules are added.

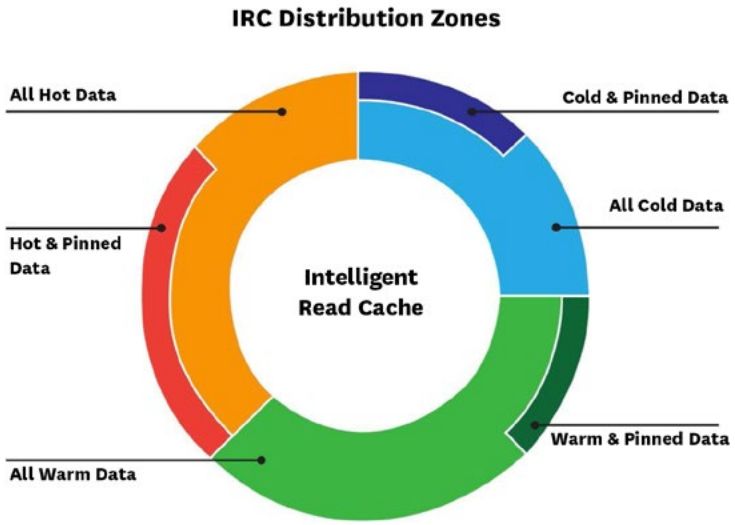
Blocks residing in local cache are characterized by a combination of 3 different temperature states, 2 modification states, and 2 protection states. These are:
• Pinned – Blocks that have been pinned receive the highest priority in the SmartCache and are the last to be evicted, but only if critical write space is needed.
• Hot – Blocks frequently being accessed for reads (from 0-7 days). The goal is to have mostly hot blocks in the local cache.
• Warm – Blocks that were recently hot but have not been read as recently as any of the hot blocks (8-30 days). They will be evicted after cold but before any hot blocks if extra SmartCache space is needed.

• Cold – Blocks that have not been accessed for 30 days or more. These are the first blocks to be evicted when SmartCache needs space for pinned, hot, or warm blocks. There should always be some cold blocks as this indicates that the SmartCache completely holds all pinned, hot, and warm blocks.
• Recently modified – Blocks that have been written to as part of updates to a file.
• Not modified – Blocks that have not been written.
• Protected in the cloud – Blocks that have been successfully uploaded to the cloud storage.
• Not yet cloud protected – Blocks that are pending upload the cloud.

Pinning consumes SmartCache space by forcing complete files into the local cache. The amount of space consumed by pinned data depends on the aggressiveness of the pinning policy and the number of rules. There are many objectives and use cases for pinning, not all of which can be documented in this paper. Pinning is a technology designed for the administrator to satisfy user or site needs. From a general perspective, pinning overrides the SmartCache's auto-caching logic to disable eviction indefinitely for specific blocks.

Because of this, careful attention to specific pinned rules should be given to prevent a rule that could cause thrashing of the local cache space (rotating eviction of data with new data due to reduced cache capacity). It is recommended that administrators utilize the Auto-Caching action or enable the Auto Pre- populate feature where possible.

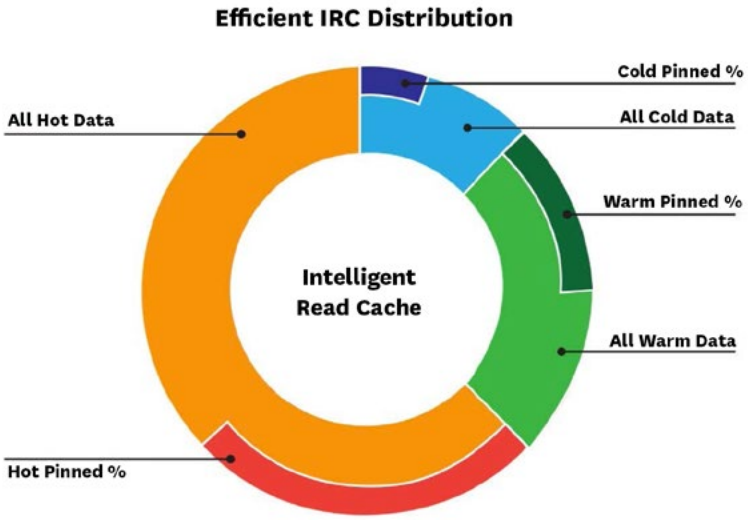
This image graphically represents how the SmartCache is logically divided into multiple zones. The main zones are the cold, warm and hot segments as classified by recent read activity. The small outsize curved bar shows how much data (as a %) within each zone is pinned while the main segment shows the total amount of SmartCache auto-cached data. As more data is written to the cache, it is highly unlikely that the IRC will contain "recently modified cold data," as a recent read operation would have automatically changed that data to a hot or warm state.



Panzura CloudFS is designed to transition all data into the cloud as quickly as possible. Data is always committed and uploaded to the cloud before becoming hot, warm, or cold based on any recent read activity. When data is pinned, that data is only evicted from SmartCache if the administrator changes the pinning policy or space is needed for writes and all other hot, warm, and cold data has been evicted.
Pinned data is considered high-priority cache data. Inversely, auto-cached SmartCache data is treated as

low-priority cache data that can be evicted automatically by the system as Smart Cache space is needed for new hot data. As more pinned data consumes the IRC, the usable auto-cache capacity is reduced. This will negatively impact the most frequently read data, causing it to be evicted and then re-read continuously. Therefore, aggressive policies that pin large amounts of data should be used sparingly as this could cause excessive local disk I/O and reduce performance.

Ideally, most of the data that applications need should be resident in the local cache. The diagram at right depicts a case where all hot and warm data is auto- cached with some cold data and some pinned data. Overall, most of the Smart Cache local-disk space is being used by active data (hot+warm). The amount of cold pinned files should always be monitored as this indicates a pinning rule that is no longer relevant and potentially no longer needed. Those rules should be removed from the system.



The Global Cloud Filesystem

The heart of any storage system for unstructured data is the file system. Key filesystems that have shaped the market include VxFS (Veritas), NTFS (Microsoft), WAFL (NetApp), and ZFS (Sun). A successful filesystem must be highly scalable, high performing, flexible, and manageable. NetApp built much of its success around WAFL and its ONTAP OS. WAFL combined RAID, the disk device manager, the file system, replication, and snapshots (limited per volume). Its primary target is HDD. ZFS took these elements, added encryption and deduplication into one stack. Additionally, it is massively scalable, natively targets HDD and SSD, but has no native cloud integration.

The Panzura CloudFS file system was engineered to closely manage how files are utilized and stored to provide seamless, high-performance, and robust multi-cloud data management. It improves on WAFL and ZFS while integrating cloud storage as a native capability.

Any user, at any location, can view and access files created by anyone, anywhere, at any time. The file system dynamically coordinates where files get stored, what gets sent to the cloud, who has edit and access rights, what files get locally cached for improved performance, and how data, metadata, and snapshots are managed.

The structure of the file system has no practical limit for the number of user-managed snapshots per CloudFS. Panzura's innovative use of metadata and snapshots for file system updates, combined with unique caching and pinning capabilities in the Panzura Filers, allows you to view data and interact through an enterprise-wide file system that is continually updated in real time. Support for extended file system access control lists (ACLs) empowers administrators to set file access and management policies on a per user basis.

Global Namespace

At the highest level, the Panzura global namespace is an in-band file system fabric that integrates multiple physical file systems into a single space and is mounted locally on each node. The entire global namespace has the root label of the distributed cloud file system.

As an example, the following 2 global namespace paths point to the same directory (\projects\team20) and are visible from both nodes as well as locally on nodes cc1-ca (California) and cc1-hi (Hawaii).

H:\ → \\cc1-ca\cloudfs\cc1-ca\projects\team20
J:\ → \\cc1-hi\cloudfs\cc1-ca\projects\team20

It is important to note some fundamental differences between the Panzura global namespace fabric and conventional global namespace (GNS) concepts. Unlike Panzura's global namespace, conventional GNS architectures require a database process on each storage system (either in-band or side-band on a separate dedicated GNS system).

These distributed databases own and manage all file system metadata transactions. File operations are intercepted in-band, processed and acted on by the distributed database instances before each file operation is allowed to complete. Changes to file metadata anywhere within the GNS require complex out-of-band distributed database replication, synchronization, arbitration and resolution while simultaneously attempting to provide real-time access to the file with guaranteed consistency.

The reliance on multiple distributed database processes could introduce complexities, in- band latencies and operation challenges that fail to scale at global multi-site levels. Examples of conventional GNS solutions are Microsoft DFS, F5 ARX and EMC Rainfinity. Additionally, these GNS architectures are somewhat limited in their ability to offer capabilities like global snapshots. The Panzura global namespace has no reliance on underlying distributed databases and avoids common GNS limitations (e.g. speed, transactional data coherence, write order fidelity, open files, precision, in-band operation, global snapshots).

At a fundamental level, the file system fabric is the namespace engine. The global namespace is integrated into the metadata. Metadata is stored centrally in the cloud for durability in addition to being fully cached locally for enhanced performance. All filers in a single namespace or CloudFS synchronize updates to the metadata in parallel asynchronously every 60 seconds in a hub (cloud) and spoke (filer) configuration.

This is further complemented by a peer to peer synchronization event that occurs in real-time when lock dynamically moves from one filer to another through the distributed global file locking. This ensures that no two processes will be awarded locks at either the file or byte range level at the same time thus avoiding write collisions. Furthermore, changed blocks that may not have been written to the cloud yet are included in the lock exchange, ensuring immediate consistency.

Global File Locking

Global file locking is at the heart of allowing geographically distributed users to work collaboratively, without overwriting each other or creating multiple file versions.

The following core concepts and symbols are used to describe the locking flow:
• The Origin – the node where the file was originally created.
• The Data Owner – Panzura Flash Cache grants ownership of the data by the Origin. The payload is built on this system and the authoritative data instance is managed from here.
• Ownership Metadata – the metadata showing the Data Owner state.
• The Authoritative Write Node – the Panzura system where locks for a data write operation are executed. This will normally be the Data Owner. Writes here always happen at LAN speed.
• Traditional File Lock – a lock issued against a file by a file system, a server or an application. This lock may consist of extensive application specific meta-information and be written into part(s) of the file payload and/or its file system metadata.
• File Coherency Locks – file system locks that are issued by applications in order to arbitrate guaranteed consistency between applications writing/ reading to a single file. An example is Microsoft Office application locks.
• Opportunistic Caching Locks – a delegated right issued by a file server protocol engine for a remote client to cache a file locally to increase client- side performance. This is not necessarily a guaranteed write lock. This delegation may be revoked by a file server at anytime. An example is Microsoft SMB Oplocks.
• Data Asymmetry Resolution – Panzura technology that efficiently resolves differences between remote sets of files and transports the required blocks to the Data Owner.

Data Ownership, Data Locking and Data Mobility
The Panzura Distributed File System was designed so that data and metadata are physically decoupled. This decoupling enables the file system to be highly flexible in referencing which physical blocks are used to construct a file. Global distributed file locking leverages this flexibility by assigning a Data Owner to all files.

This Data Owner state is held within the ownership metadata of each file and is easily transported via snapshots. A Panzura Filer that wants to be the new Data Owner communicates with the origin (the node where the file was originally

created), whose location is defined by a unique unified namespace path for each file when the file was originally created.

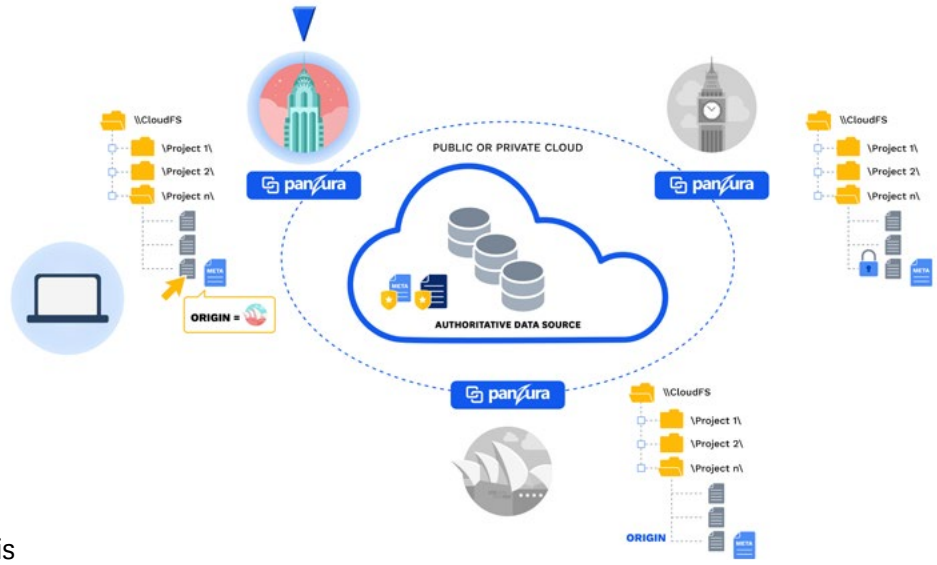
Within distributed file locking, Data Ownership status naturally flow from node-to-node. Data Ownership transitions are frequent events and are negotiated via small real-time peer-to-peer communications among Panzura systems. As the Data Owner flows to a new node, that new node instantly becomes the authoritative write node. All new writes to the file will now happen at the new Data Owner node at full LAN speed. Note: The origin is never involved in the I/O data-path during a write operation once the Data Owner successfully migrates.

The final step in assimilating blocks after a Data Owner transition is to resolve any data asymmetry. This involves a direct peer-to- peer communication between the origin and the new Data Owner, and possibly the current Data Owner (which might not be the origin). Within this peer-to-peer stream, the ownership metadata computes a final delta list of real-time changes that may have occurred since the Data Owner changed. This list, which can be as small as a single file system block, is streamed directly to the new Data Owner via a secure optimized data channel. The new Data Owner processes all remaining deltas, making the file current and consistent.

All file reads and writes from that Panzura system now happen as local I/O operations on the new Data Owner. The Data Owner retains full read/write ownership until a new Data Owner transition occurs.

A 2-User Transaction

In this transaction two users will open the same file for write, from one unique global namespace path. Each user will experience LAN speed I/O for the read and write operations as well as 100% data consistency via the Data Ownership locking and datablock mobility.



User 1 is in Sydney and creates a new file called “File.doc” on the Sydney node. This transaction defines the file Origin because it assigns the unique global namespace path to a new file. The metadata is updated and Data Ownership is assigned as pz-cc1(machine name) in Sydney.

The Sydney node will eventually write the metadata and payload for File.doc to the cloud. This will happen independently of the original write. All nodes in the Panzura CloudFS will independently read from the cloud the metadata of all other nodes from their private regions and update their file systems and namespace view.

Later in the sequence, User 2 (in New York) tries to write to the same file by accessing the unique global namespace path on his local system in New York (pz- cc2). This is the start of the global read-write locking transaction. Controller pz-

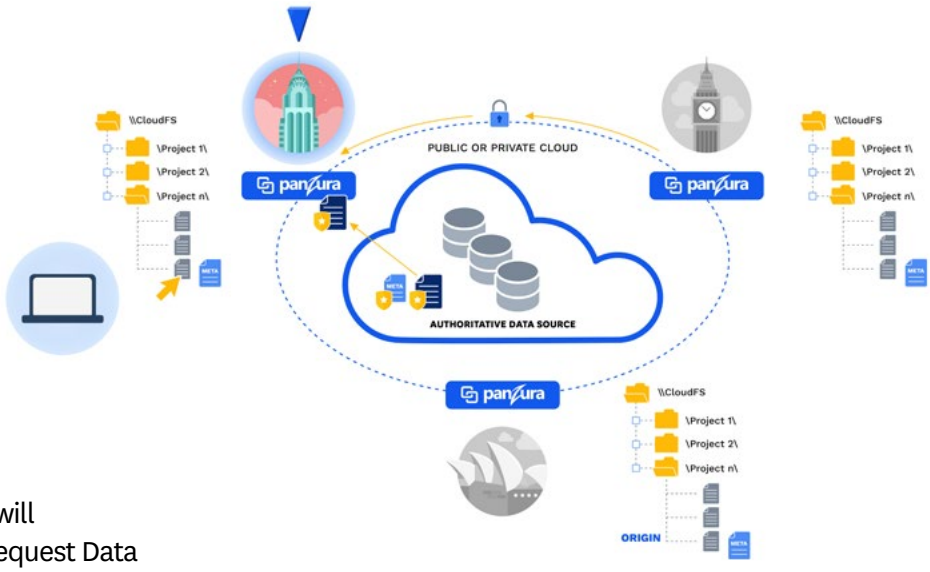
cc2 will request Data Ownership from the Origin. The Origin evaluates the Ownership Metadata state and discovers that the current Data Owner is itself. Controller pz-cc2 is granted Data Ownership and the Ownership Metadata is updated by the Origin.

From here, pz-cc2 will resolve the location of all blocks. Some may come from the local file system instance and some may be transported in from the cloud. As a final consistency state check, pz-cc2 asks the previous Data Owner if it holds any in- flight data blocks that have not yet been synced to the cloud. This resolves any data asymmetry. The processes occur in parallel to all other block reads and results in the entire state of the file being fabricated on pz-cc2 with guaranteed data consistency. At this point pz-cc2 will serve the file to the client at local LAN speeds. All other nodes are updated via the normal snapshot process.

A 3-User Transaction

In this transaction, 3 users will open the same file for write, from one unique unified namespace path. Each user will experience LAN speed I/O for the read and write operations as well as 100% data consistency via Data Ownership locking and data block mobility.

User 3 is in New York and wants to write to "File.doc". His application opens the path in the global namespace via his local view, which is consistent. During the initial local file open operation, the New York Panzura Flash Cache (pz-cc2) evaluates the global namespace path and identifies the Sydney Panzura Flash Cache (pz-cc1) as the Origin.



The New York Panzura Flash Cache will contact the Sydney appliance and request Data Ownership. Sydney evaluates the Ownership Metadata state and discovers that New York is not the current Data Owner; the London appliance is (pz-cc3). Sydney will now arbitrate a Data Owner transition. As London relinquishes Data Ownership, Sydney updates the Ownership Metadata with a new Data Owner (New York) and advises New York that it is now the new Data Owner.

At all times, New York knew that Sydney was not the Data Owner and that London was the last Data Owner but in order to enforce consistency, the Origin was leveraged as the authoritative arbiter of the transition since it must also update the Ownership Metadata.

New York is now the new authoritative Data Owner and will compute any missing blocks that are not localized in the New York file system. These blocks can come from multiple locations in parallel.

As part of this process, pz-cc2 (New York) will communicate directly with London (pz-cc3) to compute any in-flight blocks not yet in the cloud from the last Authoritative Write Node (i.e. the previous DO, which was pz-cc3, London). Any in-flight blocks are streamed directly between London and New York (peer-to-peer) to resolve data asymmetry. The result is that the full block structure in New York (pz-cc2) has guaranteed data consistency.

At this point, pz-cc3 will serve the file to the client at local LAN speeds. All other Panzura systems are updated via the normal snapshot process.

Global Deduplication

Unlike other deduplication solutions, which were designed to offset inherent data duplication in localized, inefficient file systems, Panzura designed an interconnected, global file system that stops file-level duplication before data gets stored. Since only unique copies of files across all sites are preserved by the filesystem, data is deduplicated before it is ever stored.

Capacity is optimized further by running advanced, inline block- level deduplication on any data that gets stored on the network in order to remove blocks common across different files.

Unlike any other deduplication provider, Panzura embeds the deduplication reference table in metadata, which is instantly shared among all Panzura Filers. This inline deduplication method removes data redundancy across all filers, rather than just based on data seen by a single controller. Thus each controller in the network benefits from data seen by all other controllers, ensuring even greater capacity reduction, guaranteeing all data in the cloud is unique, and driving down cloud storage and network capacity (and cost) consumed by the enterprise.

Cloud Mirroring

Using cloud mirroring, you can effectively double the availability SLA of any single cloud storage provider while providing uninterrupted service in the case of a cloud storage service outage. Introduced in Panzura CloudFS 8, cloud mirroring will automatically failover to a redundant cloud storage provider in the case of a failure of the primary provider **without disrupting any front-end file services for systems or users**.

This is only made possible because cloud mirroring delivers immediate data consistency. Failover at time of failure is not possible with eventual data consistency, which is what most other replication features offer. When the primary cloud object store is back up, Panzura will automatically synchronize both clouds to a consistent state – all without human intervention. Additionally, you are protected against accidental object or bucket deletion.

The cloud mirroring functionality addresses problems of auto-failover in case of cloud failure, provides a full backup beyond single cloud replication and automatically initiates syncing of clouds after failure. As enterprises increasingly employ multiple clouds for storage, cloud mirroring helps by eliminating dependency on any one vendor.

Military-Grade Encryption

One of the top concerns most frequently expressed by IT professionals about cloud storage is data security. Because data is being transmitted to and stored by a 3rd-party cloud storage provider outside the corporate firewall, some worry that their data will be exposed and at risk of theft. The perception is that keeping data inside the firewall is inherently safer. This concern must be overcome by any cloud storage solution before it can become mainstream within an enterprise.

Panzura addresses data security concerns directly by applying military-grade encryption to all data stored in the cloud. Each Panzura Filer applies AES-256-CBC encryption for all data at rest in the cloud. In addition, all data transmitted to or from the cloud is encrypted with TLS v1.2 to prevent access via interception. Encryption keys are managed by the enterprise, never stored in the cloud.

This complete, robust two-tier encryption solution is in addition to the typical multi-layer security provided by mainstream cloud storage providers. In some cases, enterprises find that the combined security of a Panzura+cloud solution is greater than they can reasonably achieve within their own infrastructure, making cloud storage safer than some private cloud deployments.

Summary

The cloud offers tremendous potential for enterprises to reduce storage costs, improve productivity, and reduce data availability risk. Tapping that potential fully and effectively can provide significant competitive advantage while reducing both business and technological risk.

To date, enterprises attempting to fully integrate the cloud as a storage tier have been faced with building their own limited solutions by kludging together different technologies from various vendors, many of which were never designed to be used with cloud storage. This approach to implementation fails to realize the full benefits of cloud storage while consuming precious IT resources in implementation and management.

Panzura lets organizations unify enterprise data, and scale as required by deploying cloud storage with confidence and ease, without sacrificing productivity or compromising existing workflows. With Panzura, you can break the unending cycle of on-site storage expansion, eliminate islands of storage that make it difficult for people in different sites to work together, increase your productivity, and enjoy real-time data protection.

See it For Yourself

See Panzura in action and ask our experts anything, by requesting a no-obligation demo at **panzura.com/demo**