

Panzura Snapshot Technology

Panzura Snapshot Technology

There have been many attempts to design a true global file system but those efforts have historically foundered in one or more of three areas: Consistency, currency, and efficiency. Panzura has leveraged snapshot technology to address each of these issues in unique and effective ways.

A core requirement of any file system is **consistency**. Consistency in a global file system ensures that data is recoverable anywhere, anytime in the event of a loss. Consistency could be achieved through complete reliance on cumbersome and slow standard backup processes that consume time and storage capacity and can delay recovery significantly. Alternatively, inherent data protection processes can be integrated into the core of the file system to provide data protection and rapid recovery without a separate backup process.

Another major challenge in deploying a consistent and functional global file system is making sure any changes to the file system at any location or time are disseminated to the rest of the infrastructure as quickly and accurately as possible for currency. A global file system must ensure that all users have current views of the entire file structure. When a sales manager in India uploads a purchase order, that file should be visible in the file system views of the executive team in Brazil and the finance team in South Africa with little to no perceptible lag. The goal is to mimic as closely as possible the experience that users would have if they were all in the same location using standard NAS.

Finally, an effective global file system must provide efficiency through enough user control to minimize the impact of managing that file system on IT. Enabling users to perform their own simple self-service file recovery can significantly shorten recovery times and reduce the workload spent by IT on mundane storage tasks.

Panzura has addressed all three requirements by innovating and applying new continuous snapshot technology that provides both consistency and currency across the global file system along with ongoing inherent data replication to supplement or replace standard backup methodologies. In addition, efficiency from user-managed snapshots gives users more flexibility in recovering lost files without involving IT.

Snapshots for Consistency

Snapshots capture the state of a file system 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 (Figure 1). 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.

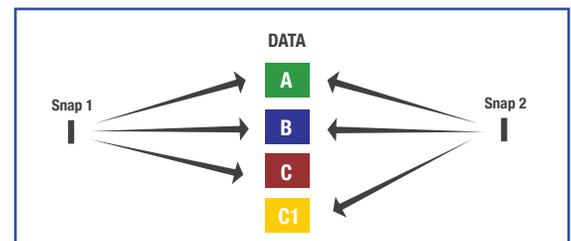


Figure 1: Snapshots maintain file system consistency

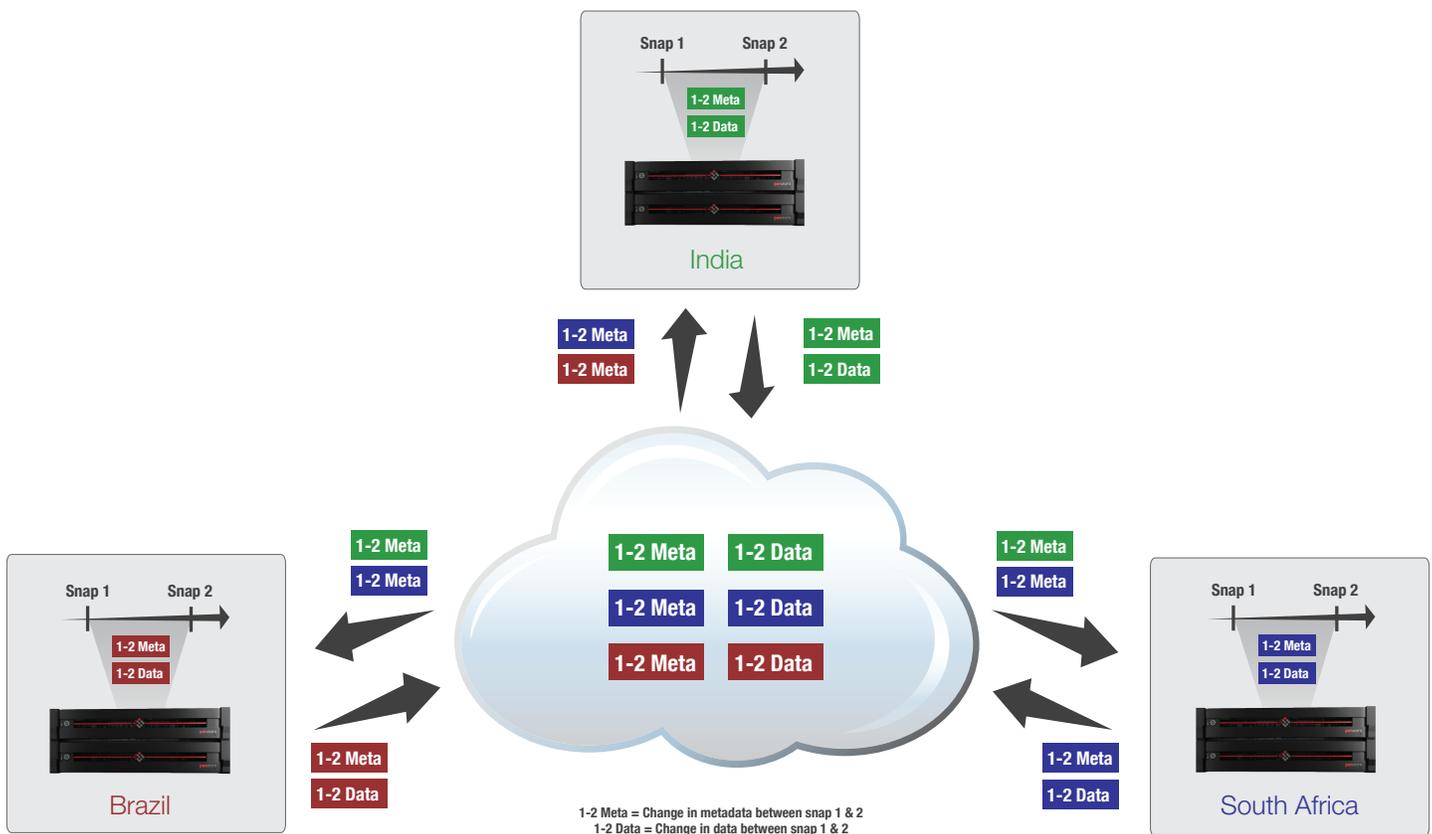
By using snapshots for creating and saving an ongoing series of recovery points for different stages in data's lifetime, a consistent state of the file system can always be restored in the event of a data loss.

Snapshots for Currency

Panzura uses **differences** between consecutive snapshots both to maintain file system consistency as well as to protect data in the file system. In a process called syncing, the Panzura file system 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 Quicksilver controllers in the infrastructure, where they are used to update the state of the file system and maintain currency (Figure 2). This system updating occurs continuously across all controllers, with each controller sending and receiving extremely small metadata snapshot deltas and using them to update the file system seamlessly and transparently.

For example a controller in Brazil (red in Figure 2) 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 in Figure 2).

Figure 2: Snapshots' role in global replication



All of the changes in data and metadata are stored and tracked sequentially in time such that should a data loss occur at a controller 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. This inherent data protection is explained in more detail in the white paper *Inherent Data Protection – How the Panzura System Protects Information Without Backups*.

It is important to reiterate that the size of these snapshot deltas are exceptionally small relative to the data in the file system; thus they can be captured continuously and use bandwidth and capacity very efficiently. The result is the Holy Grail of a global file system: a solution that requires almost no overhead but provides near real-time, continuous rapid updates across all sites for global file system currency.

The key to a current global file system is accurate and efficient transfer of only that data that is needed to make sure the file system views of each controller remain current. Panzura snapshot technology enables currency across a globally-dispersed file system with minimal overhead, providing local NAS responsiveness to a worldwide infrastructure.

Snapshots for Efficiency

In addition to system snapshots used to maintain consistency and currency, Panzura systems also provide the ability to capture and save up to 10,000 user-managed snapshots per volume vs 255 with NetApp. This category of snapshots allows users to recover data on their own, up to a year in the past depending on capacity, by simply finding the desired snapshot in their inventory and restoring it (Figure 3). 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 Quicksilver controller and navigates to her snapshot folder, finds the date/time that corresponds to the file system 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 file system 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.

IT administration can dynamically change snapshot policies as needed to balance frequency and duration for optimal system performance and user satisfaction.

Conclusion

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

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

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 Global Cloud Storage System brings the power of the cloud to enterprises without sacrificing the user experience.

