Tintri

TECHNICAL WHITE PAPER

Ensuring Robust Data Integrity

Executive Summary

Purpose-built storage appliances serving enterprise workloads typically use commodity hardware with value-added OS and file system software. The components — hardware and software — of these appliances can and do fail from time to time. For example, the total failure of a controller in a dual-controller system is immediately user-visible, albeit without operational impact on serving data. Other failures, such as firmware errors, are subtle and can cause corruption that may only surface much later.

Tintri™ VMstore™ appliances are purpose-built to leverage cost-effective multi-level cell (MLC) flash solid-state drives (SSD) and high-capacity SATA hard disk drives (HDD) using a proprietary virtual machine (VM)-aware file system. The Tintri OS runs on VMstore appliances and has comprehensive data integrity and reliability features to guard against hardware and software component malfunctions. These data integrity features work in concert to provide optimal performance and system availability.

This white paper describes how the Tintri OS and its underlying file system provide unparalleled data integrity and intelligent performance for serving hundreds of VMs in a single appliance. It also examines Tintri's proprietary redundant array of independent disks (RAID) implementation for comprehensive data integrity.

Introduction

Data integrity is essential to storage, especially primary storage systems. A robust multilayered approach must be used to protect against all manner of hardware and firmware errors. Possible failures include:

- Complete failures, such as a controller failure or a drive failure.
- The many erroneous behaviors (short of complete failure) by individual components such as HDD and SSD. For example, drives can return corrupted data on read operations, or fail to write data, etc.
- Data is constantly moved even after it is written to stable storage. Data movements can potentially compromise data integrity. Examples include:
  - SSD's complex internal mechanisms such as garbage collection, due to asymmetric read and write granularities.
  - HDD's remap due to bad sectors.
  - File system garbage collection as a result of deduplication and compression.
- Advanced functionality such as deduplication and compression can turn otherwise small errors into major issues. Many files reference the same block of data as a result of deduplication, and an error with one block can affect all related files.

A comprehensive data integrity strategy must cover all of these cases and more, not just simple component failures. Data integrity features also must work in concert to ensure end-to-end integrity. The diagram below is a high-level overview of the architectural components of the Tintri OS.

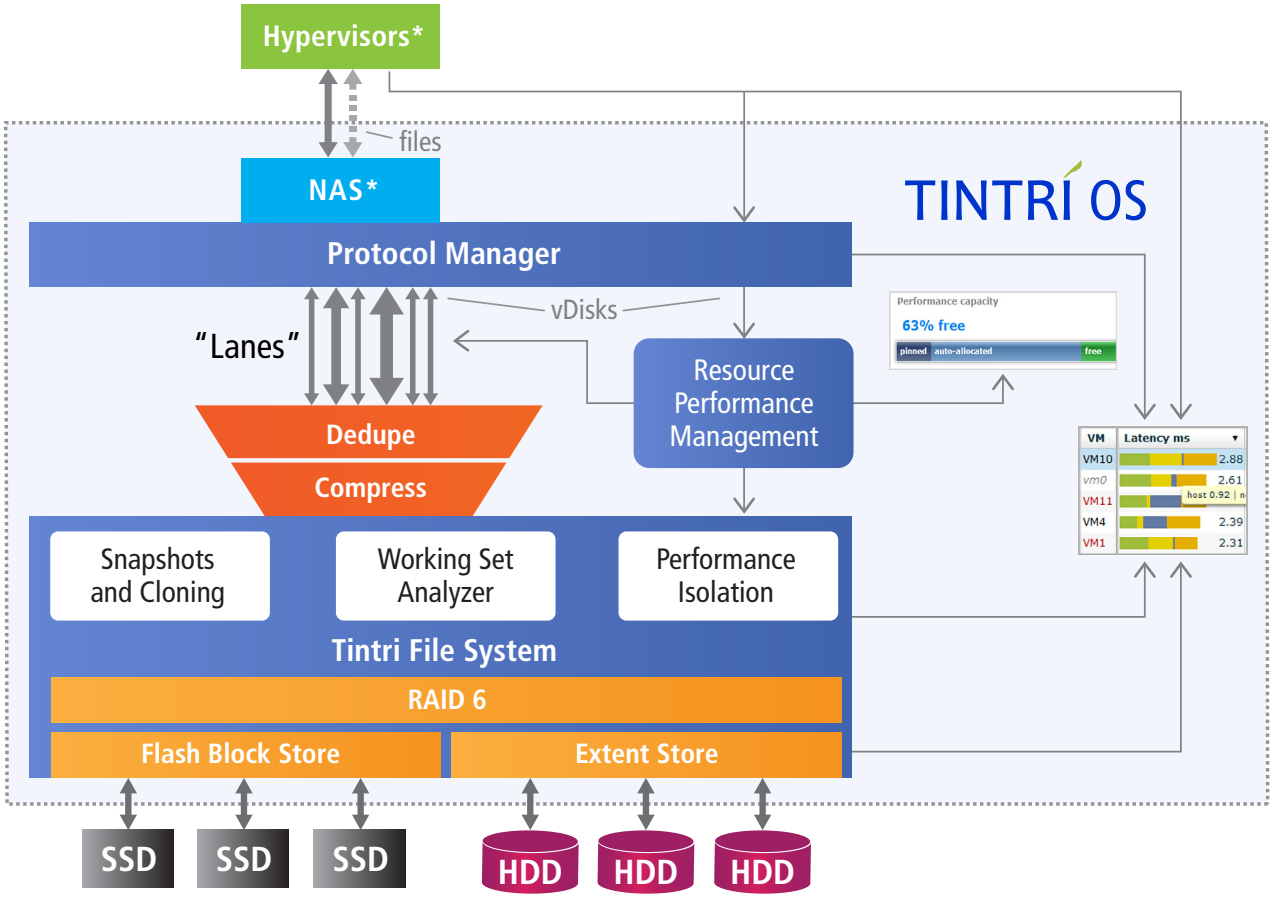

Figure 1: Tintri OS Architectural Overview
*VMware vSphere with NFS currently supported.

Data Integrity in Tintri Operating System

The following are major focus areas for the Tintri OS comprehensive data integrity approach:

- Purpose-built for VMs
- Proactive error-avoidance design
- RAID 6 with real-time error correction
- Inline integrity protection and verification
- Self-healing file system
- High-availability

Purpose-built for VMs

VMstore appliances running the Tintri OS, unlike traditional general-purpose storage systems, are purpose-built for understanding and serving VMs. Its design considerably simplifies the environments and clients the Tintri OS serves to just hypervisors, enhancing behavior predictability. Fewer moving parts and rigorous qualification ensures full compatibility between hypervisors and the Tintri OS and almost eliminates errors from environmental interaction, such as software and infrastructure compatibility. Currently, Tintri OS supports VMware vSphere hypervisor with NFS datastores, and also extends to other hypervisors, such as Citrix XenServer.

Proactive error avoidance design

One of the biggest risks to file system integrity is a software error when writing new data. Like garbage collection, this can accidentally overwrite data, and updates to metadata can mangle existing structures. Tintri OS and its file system avoid these errors by using nonvolatile RAM (NVRAM) for write buffering and by writing data directly to SSDs configured in a RAID 6 group. Major benefits include:

- **No partial stripe writes:** The Tintri OS file system never updates just one data block in a RAID stripe. All new writes go to new RAID stripes — both on SSD and on HDD — and new RAID stripes are written in their entirety. Compared to complex partial-stripe writes, this ensures disk reconstruction, in case of drive failures, is consistent and ensures no data is lost. The graphic below shows the full-stripe write scheme used by the Flash block store (SSD) and the Extent store (HDD) in figure 2.

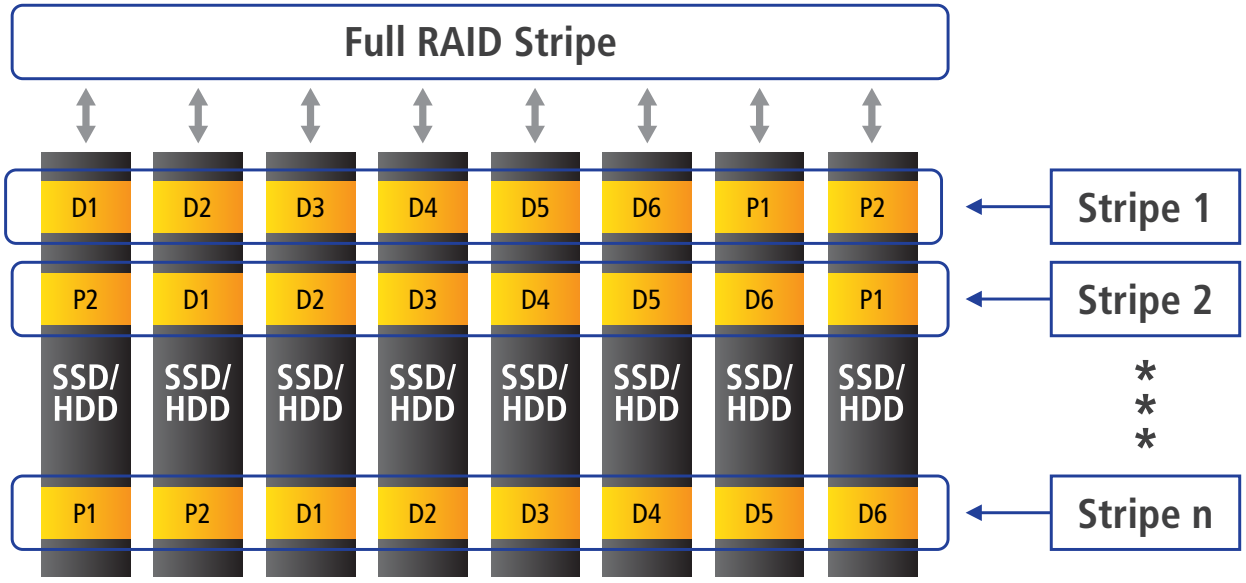

Figure 2: Tintri's RAID 6 writes full stripes to both SSD and HDD

- **NVRAM for fast buffering and fail protection:**
  - Tintri VMstore appliances include an NVRAM write buffer into which all data blocks are written before they are committed to SSD. NVRAM mechanisms protect data from power failure by writing it to an onboard flash medium. Data blocks are buffered in NVRAM and written to flash sequentially in large chunks to enhance endurance and reliability of SSD.
  - The Tintri OS file system leverages NVRAM to implement a safe file system restart. As it restarts, the Tintri OS file system verifies the integrity of the data in the NVRAM buffer before applying it to the file system, ensuring no data is lost due to file system restart. If the NVRAM fails, the file system ensures the integrity of data to ensure VMs can be restarted safely.

RAID 6 with real-time error correction

Double drive failure protection

RAID 6 is the basis for data protection on SSD and HDD, and provides continuous error detection and real-time healing. The dual-drive failure protection architecture of RAID 6 offers significant advantages over single-drive failure approaches like RAID 1 and RAID 5. Uniform RAID 6 simplifies configuration compared to storage systems that deploy multiple RAID levels in a single array.

RAID 6 ensures availability even during double-drive failures. Using this approach, VMstore appliances remain operational even if two SSDs and two HDDs fail concurrently. Unlike legacy disk-based systems, application performance impact is minimal: more than 99 percent of data is served from flash and SSD has outstanding performance for quick background reconstruction.

Many legacy storage systems employ single-parity RAID schemes, leaving them vulnerable to data loss in the case of two simultaneous disk failures. In single-parity RAID schemes, once a single disk fails, another disk failure causes data loss and possibly corruption. Storage systems that use flash as merely a read-cache without RAID protection can suffer a significant loss in performance. In these systems, all data on entire drives and the cache must be rebuilt after failed drives are replaced.

Real-time error correction

An individual data block may be unreadable if bits are flipped on the storage medium or if there are firmware errors. Drives generally suffer from many insidious failures such as silently losing a write, writing a block to an incorrect location, or reading data from an incorrect location. File systems must be able to detect and correct errors. Tintri OS' RAID 6 software detects and self-heals errors in real-time. Figure 3 shows the workflow for real-time error detection and correction using RAID.

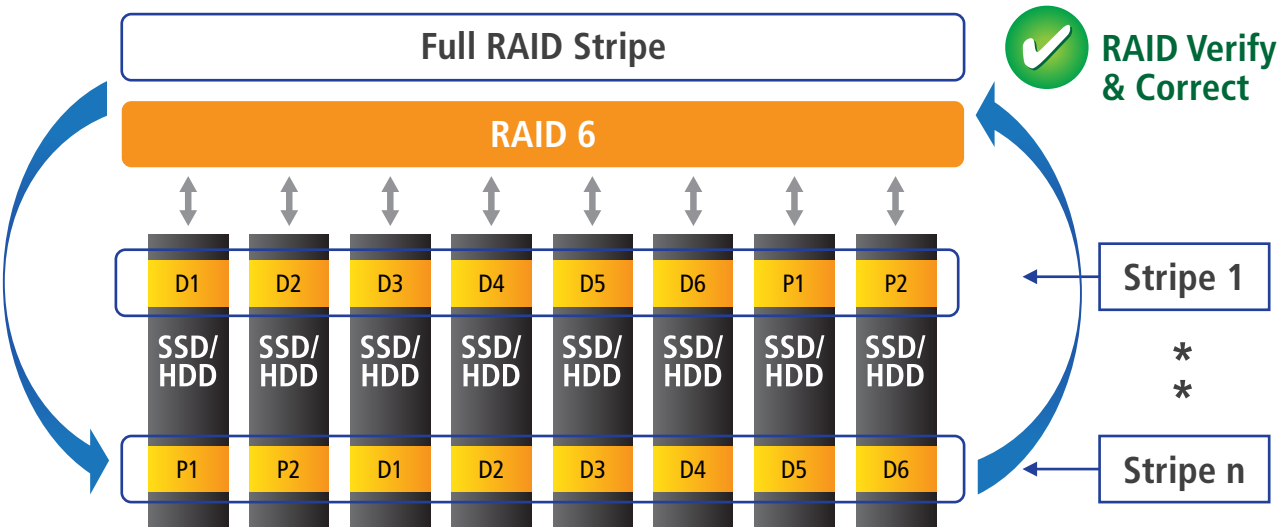

Figure 3: Real-time error detection and correction with RAID 6

The underlying file system of the Tintri OS stores all of the objects it manages — metadata and data — in blocks with strong checksums. On every read, the file system first verifies the object read from disk by computing checksums and ensuring the computed value matches what is retrieved. If an issue is found, RAID 6 corrects and self-heals the system.

Inline integrity protection and verification

Storage systems that use flash must protect against wear-out (MLC SSD) and reliability (all types). SSD adds complexity because of its asymmetric program and erase granularity and internal complexities, such as garbage collection. Primary storage systems that use flash as a bolt-on do not use even the most basic form of failure protection (such as RAID); this severely impacts cost and performance. These systems use expensive single-level cell (SLC) flash or use flash as merely a read cache.

When Tintri OS receives a write request, after analyzing for redundancy, it stores the data block along with its checksum on SSD. For a duplicate block, a read is issued for the existing block from flash to ensure a match. A checksum is computed and stored with each data object — both in flash and on disk — and verified whenever the object is read. A self-contained checksum may be valid if a drive substitutes one read for another because of DMA errors, internal metadata corruption, etc. So an inline checksum by itself cannot catch all device errors. A referential integrity check is needed to detect corruption, to avoid bigger issues by returning incorrect data. To ensure referential integrity, references to data objects contain a separate checksum that is verified against the checksum of the object being read.

These techniques ensure the data on SSD and HDD are readable and correct and the file system metadata used to locate data is readable and correct. Potential problems, such as a disk controller returning bad data, are caught and fixed on the fly. In most cases, problems can be corrected through self-healing as described in the “Real-time error correction” on page 5.

Self-healing file system

RAID-assisted real-time error detection works well for active data, but does not address errors with cold data, such as data blocks referenced by snapshots for long periods of time. To guard against corruption, VMstore appliances actively re-verify data integrity on SSD and HDD in an ongoing background process.

For data stored on HDD, there are two levels of scrub process to identify and repairs errors.

- As new data and its checksums are written, a background process reads entire RAID stripes of data written to disk and verifies checksums for integrity. If there is an error, RAID heals the system in real-time. This helps correct transient errors that may occur in the write data path.
- A weekly scheduled scrub process that requires no user intervention re-verifies all data stored on disk, ensuring any errors are detected and corrected. This helps correct cold-data errors.

For data stored on SSD, a continuous scrub process runs in the background to read full RAID stripes of data at fixed-time intervals, and compares computed checksums. If there is an error, RAID corrects errors in real-time. Checksums for each data object inside the RAID stripe are also computed independently and matched with what is retrieved from SSD. Figure 4 shows the workflow for real-time error detection and correction using RAID.

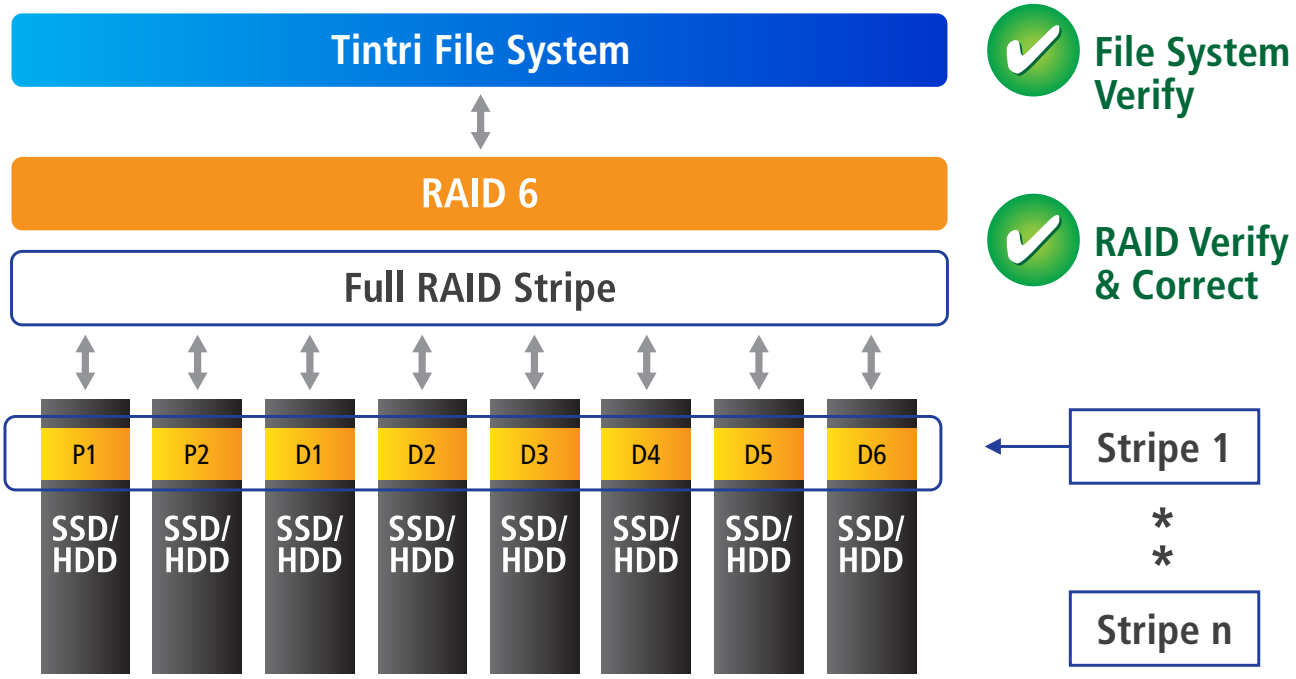

Figure 4: On-going scrubbing process for error detection and correction with RAID 6

Through RAID 6 real-time error correction and ongoing scheduled data scrubbing, most storage-medium generated errors are identified and fixed with no impact to file system or storage system operation.

Referential integrity and recoverability

Tintri's file system stores data on SSD (in blocks) and on HDD (in extents). The metadata that describes the data is stored on SSD (in pages organized in page pools). Every object – data block or extent and metadata page – has a checksum and a descriptor (the self-describing module of an object). The descriptor of a data object describes the file and the offset in that file the object belongs to and similarly the descriptor of a metadata page describes the page pool to which a metadata object belongs and whether it is the latest version. Tintri file system stores checksums that tie an object and its descriptor, so that lost writes, misplaced reads, or other such perfidious errors do not corrupt data. The self-describing nature of data and metadata helps recover from disk and firmware errors. Figure 5 shows the self-describing structure of Tintri file system.

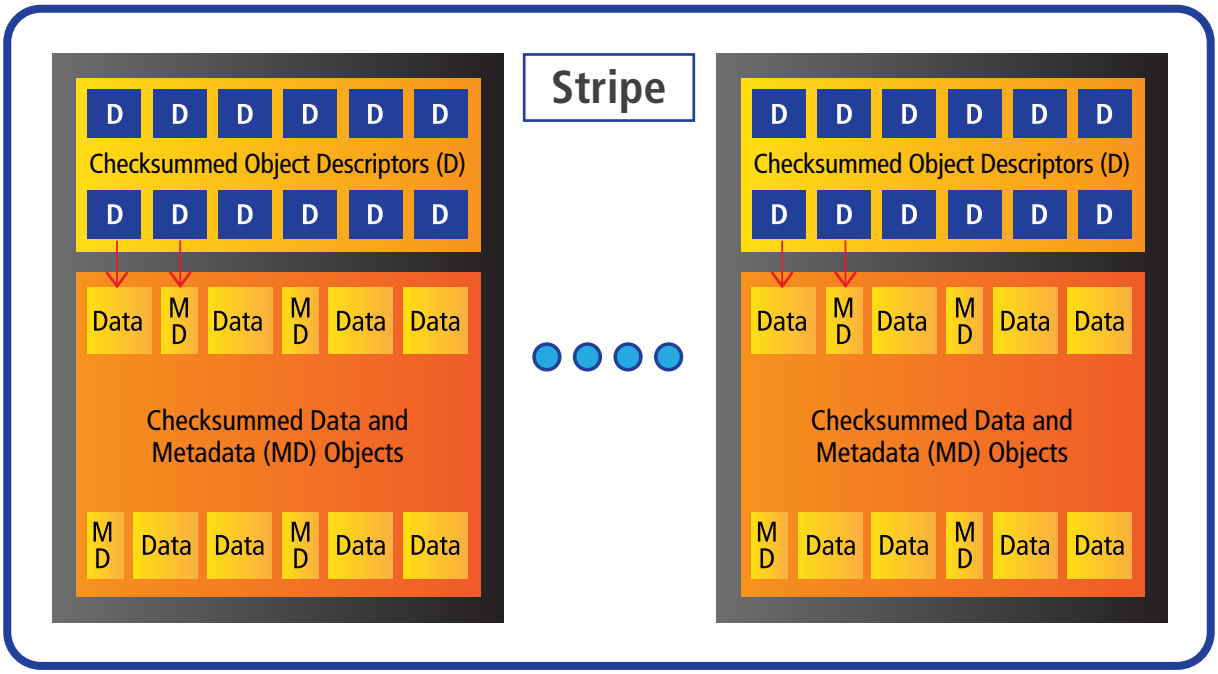

Figure 5: Self-describing structure of Tintri file system

The Tintri file system consists of a hierarchy of references with blocks and extents at the lowest level and metadata mapping them at higher levels. Referential integrity is maintained at each level using strong checksums to detect errors. The checksums defend against aliasing issues such as a file pointing to wrong data blocks. Further, metadata objects have version number in metadata pages to detect similar aliasing issues.

As described in the no partial stripe writes section, data blocks and extents are written to SSD and HDD respectively in full RAID stripe units. Techniques described in self-healing file system detect and correct errors with cold data. In the unlikely event that an unrecoverable disk error causes orphaned or corrupt objects, a scan of the self-describing objects helps detect and correct the problems.

High availability

Tintri VMstore appliances have dual-controller architecture to provide high-availability. Tintri OS incorporates a number of data integrity features in its high-availability architecture. The Tintri file system syncs data from the NVRAM on the active controller to that on the standby controller, to ensure file system contents match. A strong fingerprint is calculated on the contents of NVRAM on the primary controller as data is synced and verified with the fingerprint calculated independently by the secondary controller.

When Tintri OS receives a write request, data is buffered into the NVRAM device on the primary controller and forwarded to the NVRAM device on the secondary controller. After acknowledgement from the secondary controller that data was safely persisted on its NVRAM, the primary controller acknowledges the write operation to the hypervisor. This, combined with techniques discussed in NVRAM for fast buffering and fail protection ensures controller failures have no impact on data integrity.

Conclusion

Modern storage systems must employ multiple mechanisms at different layers — hardware, operating system and file system — to ensure data integrity. Robust coordination among these mechanisms guards against errors that could occur in various components of a storage system. Unlike traditional storage systems, VMstore appliances are purpose-built for serving VMs with data integrity as a key design consideration. The Tintri OS incorporates a number of techniques to provide data integrity, and a combination of these techniques is not available in many general-purpose primary storage systems.

The simplicity that comes from being purpose-built for VMs limits the scope of supported clients to a few types of hypervisors, and thorough qualification reduces any errors from environmental interaction. An error avoidance design philosophy and robust implementation minimizes the chances of software errors in the first place. The purpose-built file system and full-stripe writes ensure data is always safe and guards against potential errors, software or otherwise.

All data and metadata objects written to SSD and HDD are protected by checksums, which are verified on all reads. Data corruption is immediately detected and corrected using RAID 6 protection. Furthermore, metadata references contain referential integrity checks to detect software errors that may corrupt metadata.

Tintri's proprietary RAID 6 implementation for SSD and HDD protects against double-drive failures, rebuilds a failed drive even if there is a data read error, and corrects errors in real-time during read operations. Continuous scrub processes actively identify and self-heal latent errors.