
# Cohesity Architecture White Paper

Building a Modern, Web-Scale Architecture for Consolidating Secondary Storage

COHESITY

## The "Band-Aid Effect": The Hidden Cost of Trying to Make Legacy Solutions Work

The combination of explosive data growth and traditional siloed architectures has presented a major challenge to many companies. Over the years, these organizations have tried to address new business requirements by adding, swapping, and integrating new solutions into legacy architectures. While this approach has succeeded in providing temporary relief, it has also created a "band-aid effect," in which each stopgap solution makes it harder for organizations to adapt to future data demands. It continues to create multiple copies of data, which further accelerates data growth. Consider three common examples of modern business initiatives that have resulted in massive data sprawl across the enterprise: business continuity, general purpose workloads, and data analytics.

**Business Continuity:** Business continuity and disaster recovery strategies are absolutely critical to ensure data is always available, even in the event of a disaster or total loss at the primary site where applications reside. Oftentimes, organizations have invested in an entire replica of their production stack devoted to maintaining business continuity that sits idle until disaster strikes. This requires major capital investment in redundant solutions, ranging from target storage to backup and management software, along with the management complexity and overhead associated with maintaining the disaster recovery site. Significant investments made in disaster recovery and business continuity solutions often are seen as an expensive insurance policy, constantly adding additional costs but rarely providing value outside of the occasional restore operation.

**General Purpose Workloads:** In addition to designing, managing, and supporting production and business continuity solutions, system administrators must also address their developers' requirements. To help facilitate agile software release cycles, developers often request a replica of the production environment to stand up sandbox environments, test new releases, and iterate on them until they are ready to be delivered out to production. While organizations with large IT budgets can swallow the enormous upfront costs of a high-performance production storage solution that is architected to handle these additional development workloads, those with smaller budgets are forced to invest in a cheaper alternative, resulting in another silo of infrastructure along with its own copy of the production data.

**Analytics:** As IT organizations manage the difficult transition from a cost-center to a business partner, investing in a strong data analytics strategy becomes even more imperative for CIOs. Providing the ability to derive real-time insight from raw data that enables business owners to make better-informed decisions requires hefty investments. Companies can achieve this either through the initial investment in a high-performance storage solution that is capable of handling analytics workloads in addition to running production applications or through a dedicated data lake infrastructure.

It comes as no surprise that with each new solution that is added to address a new business initiative, the costs and complexity associated with managing data continue to expand. Along with the growing costs of protecting, storing, and managing these independent solution silos, it becomes more and more difficult to provide visibility into the data sprawl.

## It Doesn't Have to Be This Way: Bring Order to the Data Chaos with Cohesity

Cohesity was founded with the core vision to eliminate the fragmentation in data storage and put an end to the decades-long "Band-Aid effect" that has plagued data storage solutions. Architected and designed from the ground up to be the world's most efficient, flexible solution for enterprise data, the Cohesity Data Platform couples commercial off-the-shelf (COTS) hardware with intelligent, extensible software, enabling organizations to spend less time worrying about how to retrofit their legacy solutions with future needs, and more time focusing on the core functions of the enterprise.

## Introducing the C2000 Series Data Platform

Cohesity provides a starting point for infinite scale with either the C2300 or the C2500. The former providing 48 TB of raw hard disk capacity and 3.2 TB of flash storage in a dense 2 Rack Unit container. While the latter packs 96 TB of raw hard disk capacity and 6.4 TB of flash storage in the same 2 Rack Units. Each appliance is called a Block, which can support up to four Cohesity Nodes.

These Nodes can be joined together to form a cluster. Clusters can expand from a minimum of 3 Nodes to as many Nodes as necessary regardless of the series. Customers can add additional Nodes one at a time to linearly scale their capacity and performance as needed, eliminating the guessing game required with traditional scale-up solutions.

Each C2300 or C2500 node contains two 8-core Intel v3 processors and 64GB of RAM, in addition to 12 TB or 24 TB (C2300 or C2500) of raw hard disk capacity (three 4 TB or 8 TB SAS drives) and 800 GB or 1.6 TB PCIe SSD C2300 or C2500). Each Node also has its own discrete set of networking hardware, which is comprised of two 10Gb SFP+ interfaces, two 1Gb Ethernet interfaces, and an out-of-band management interface for remote configuration.

## Cohesity Open Architecture for Scalable Intelligent Storage (OASIS)

The Cohesity Data Platform is built on the Open Architecture for Scalable Intelligent Storage (OASIS), the only file system that combines infinite scalability with an open architecture flexibility that can consolidate multiple business workloads on a single platform. With built-in, native applications to support data protection, copy data management, test and development, and in-place analytics, customers experience the benefits of consolidation right out of the box.

OASIS was built from the ground up to be the most robust and fully distributed system in the market. Distributed systems are inconsistent by nature: operations that are performed on a distributed system are not atomic, meaning operations could complete on some but not all nodes, resulting in data corruption. The notion of 'Eventual Consistency' was created to address this by stating that data written to a distributed system will eventually be the same across all of the participating nodes, but not necessarily the moment the data is written. While this tends to be fine when immediate access to that piece of data is not required, in the case of enterprise file systems, where a user can very easily write a new piece of data and then request it right back in the subsequent operation, all pieces of data need to be consistent across all participating nodes. Unlike traditional distributed file systems that are 'Eventually Consistent,' OASIS leverages a purpose-built noSQL store, combined with Paxos protocols, that delivers full consistency with the ability to make decisions rapidly, at massive scale, and without performance penalties.

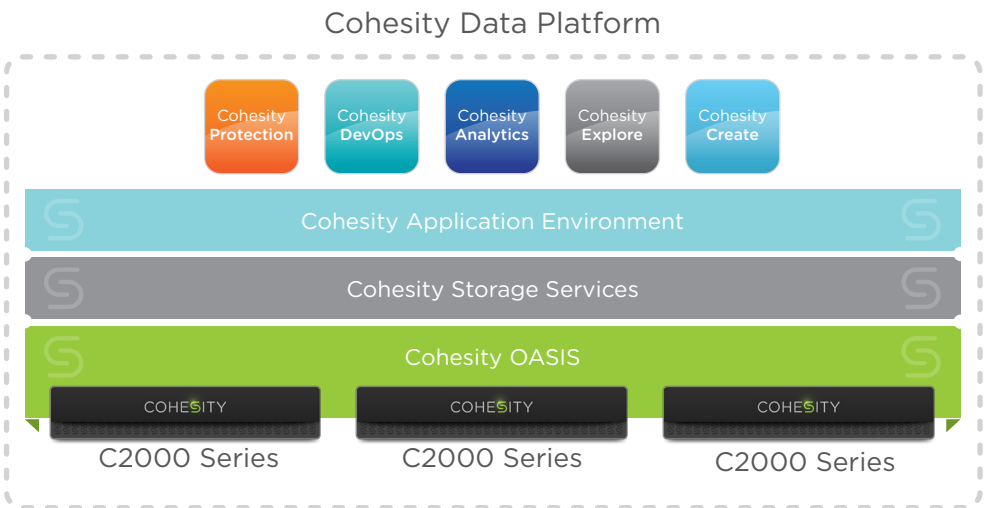



Figure 1

OASIS is comprised of several services, each one handling a key function to provide an infinitely scalable architecture while optimizing performance to enable the consolidation of multiple workloads.

**Cluster Manager:** The Cluster Manager controls all the core services that run on a Cohesity Cluster. This layer is responsible for maintaining all configuration information, networking information, and the general state of all other components in the system. This was purpose-built to provide better performance and a higher level of fault tolerance than any other existing open-source solutions on the market.

**I/O Engine:** The I/O Engine is responsible for all read and write operations that take place on the cluster. It is comprised of the write cache, which lives in SSD, and the tiered storage layers that span across both SSD and spinning disk. For write operations, as data is streamed into the system, it is broken down into smaller chunks, which are optimily placed onto the tier that best suits the profile of that particular chunk. The I/O Engine also ensures that all data is written to two nodes concurrently, providing write fault tolerance. This enables completely non-disruptive operations, even if a node were to become unavailable during a given operation. For read operations, the I/O Engine

receives the location information of the data from the Distributed Metadata Store and fetches the associated chunk(s). If a particular chunk of data is frequently requested in a short period of time, that chunk is kept in SSD to ensure quick access and optimized performance on subsequent requests.

**Metadata Store:** The Metadata Store is a consistent key value store that serves as the file system metadata storage repository. Optimized for quick retrieval of file system metadata, the Metadata Store is continually balanced across all nodes within the cluster (accounting for nodes that are added or removed from the cluster). The Metadata Store ensures that three copies are maintained at any point in time, so that data is always protected, even in the event of a failure.

**Indexing Engine:** The Indexing Engine is responsible for inspecting the data that is stored in a cluster. On its first pass, the Indexing Engine grabs high-level indices for quick data retrieval around top-level objects, such as Virtual Machine (VM) names and metadata. On its second pass, the Indexing Engine cracks open individual data objects, such as Virtual Machine Disks (VMDKs), and scans individual files within those data objects. This native indexing enables rapid search-and-recover capabilities to quickly find and restore files stored within higher-level data objects such as VMs.

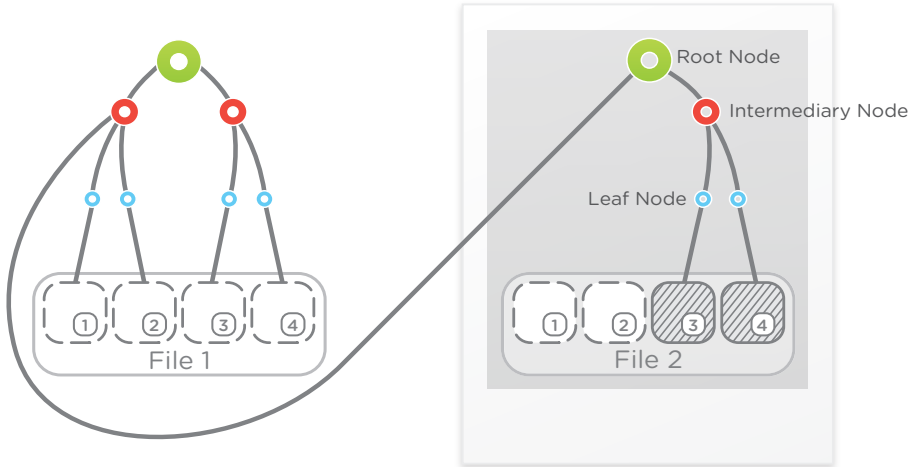
**Integrated Data Protection Engine:** The Integrated Data Protection Engine provides the basis to deliver a native, fully integrated data protection environment right from the Cohesity Data Platform. This engine is interoperable with third-party services, such as VMware APIs for Data Protection (VADP), to provide end-to-end data protection for customer environments. The Integrated Data Protection Engine is the core engine supporting Cohesity Protection.

## Cohesity Storage Services

The next layer in the Cohesity Data Platform architecture consists of the Cohesity Storage Services, which provide the storage efficiency capabilities that customers depend on at a scale that no other solution can achieve.

**Snapshots:** In legacy storage solutions, snapshots (of a file system at a particular given point in time) form a chain, tracking the changes made to a set of data. Every time a change is captured, a new link is added to the chain. As these chains grow with each and every snapshot, the time it takes to retrieve data on a given request also grows because the system must re-link the chain to access that data.

Cohesity takes a different approach with it's patented SnapTree™ technology which create a tree of pointers that limits the number of hops it takes to retrieve blocks of data, regardless of the number of snapshots that have been taken. The figure below shows how data is accessed using SnapTree.




This SnapTree shows a snapshot of File 1 called File 2, but with two blocks of data updated. Blocks '3' and '4'. The new tree containes an intermediary node and leaf nodes to the new blocks, as well as a link to the intermediary node in the original file. All blocks can be accessed within three hops in either file version.

Figure 2

Because SnapTree limits the number of hops it takes to retrieve blocks of data when it is requested, customers are able to take snapshots as frequently as they need - without ever experiencing performance degradation.

This provides the ability to create custom snapshot schedules, with the granularity of taking a snapshot every couple of minutes for near continuous data protection, to meet a wide range of data protection requirements.

**Data Deduplication:** Data deduplication is a common storage efficiency feature that frees up storage capacity by eliminating redundant data blocks. Different vendors implement deduplication at a file-level and/or a block-level of different sizes, which only works well across a single storage pool or within a single object (e.g. application or VM).

Cohesity leverages a unique, variable-length data deduplication technology that spans an entire cluster, resulting in significant savings across a customer's entire storage footprint. In addition to providing global data deduplication, Cohesity allows customers to decide if their data should be deduplicated in-line (when the data is written to the system), post-process (after the data is written to the system), or not at all.

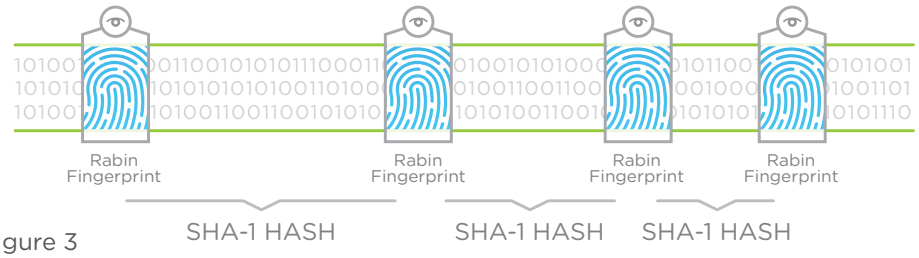



Figure 3

By using Rabin Fingerprinting algorithm, a variable-length window is created. This optimizes the level of deduplicaiton no matter the type of file

**Intelligent Data Placement:** Intelligent data placement ensures that data is always available, even in the event of a node failure. When data is written to a Cohesity Cluster, a second copy of that data is instantly replicated to another node within the cluster. For customers who have multiple Cohesity Blocks (a chassis with one or more Cohesity Nodes) or racks, Cohesity will always optimize the placement of the data by placing the second copy on a different block or in a different rack, providing an even higher level of fault tolerance. For customers with stricter fault tolerance requirements, the Replication Factor (RF), or number of copies of data that are replicated within a Cluster, can be adjusted to fit their needs.

This intelligent placement, or sharding, of data also enhances the performance characteristics of the data placed on the cluster. As the data hits the cluster, it is broken down into smaller bite-sized chunks (typically 8K to 24K). By spreading data across all nodes of a cluster the I/O load is shared across all available resources and eliminates the notion of a 'Hot Node' or 'Hot Disk' which would get accessed more frequently and would create an I/O bottleneck.

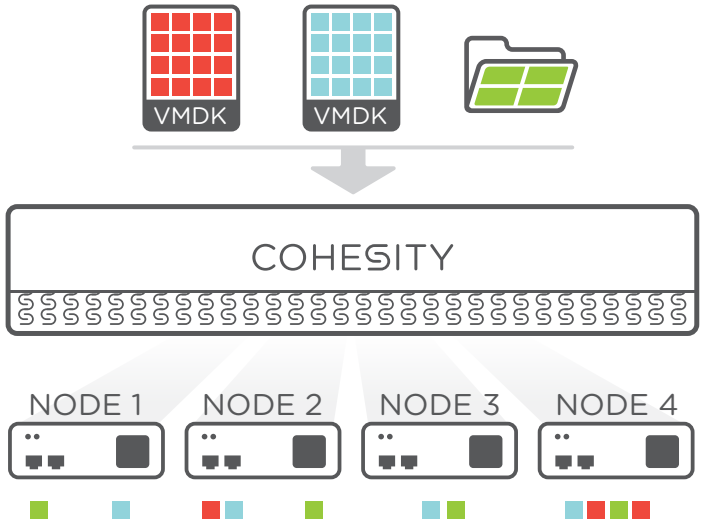



Figure 4

As data is ingested into OASIS it is evenly distributed across the available nodes of the cluser. This reduces the notion of 'Hot Nodes' or 'Hot Disks' which plague systems that keep an entire copy of the object on a single node.

## Cohesity Application Environment

To facilitate the move from fragmented silos of infrastructure, Cohesity has created the application environment, an extensible way for customers to leverage a single platform to deliver multiple applications. The environment provides a consumer-like experience to deliver business-specific applications and services, with built-in, native applications to address common operational workloads like data protection, test and development, and analytics, as well as the ability to develop custom applications through its open Analytics Workbench archecture. The first three applications are Cohesity Protection, Cohesity DevOps, and Cohesity Analytics.

## Manage and Protect Data Seamlessly with Cohesity Protection

Cohesity Protection delivers a fully integrated, end-to-end data protection environment right out of the box. Unlike traditional data protection applications that require external media and management servers, Cohesity Protection runs directly on the Cohesity Data Platform. Cohesity Protection works seamlessly with virtual environments running VMware. Cohesity leverages VMware Virtual APIs for Data Protection (VADP) to seamlessly connect to a vCenter environment and discover existing resources (e.g. VMs and ESX hosts). Once inventoried, the Cohesity Protection app triggers a VMware snapshot for objects that are designated for protection, and quiesces the Virtual Machine before taking a snapshot to ensure it is consistent. Once the snapshot is taken on the host, OASIS ingests, sorts, and stores the initial baseline copy (the first time it is protected) and will continue to protect that virtual machine with incremental snapshots, based on the deltas from the previous snapshot, for future backups. These snapshots are powered by Cohesity's patented SnapTree data structure.

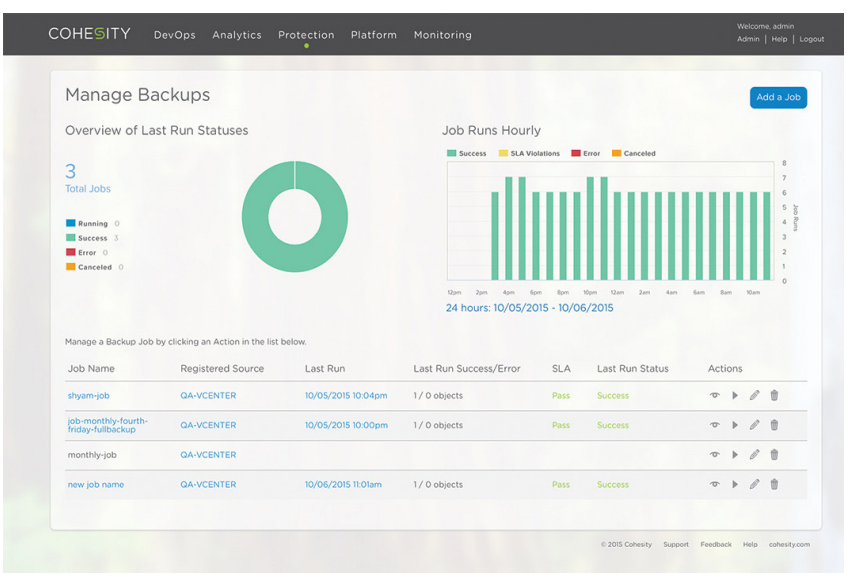


Figure 5

At its core, SnapTree uses a variant of a B+ tree data structure[1], which is optimized for storing large amounts of data efficiently on disk and in memory. This optimized search tree breaks away from the traditional link and chain methodology for organizing and storing snapshot data. Using the tree structure, access to any point in the tree takes exactly three hops no matter how many snapshots there are, without having to rebuild any chain linkage. This provides instant access to a given file system at any point in time.

Moving away from the traditional linked architecture of snapshots, in which snapshots form long metadata chains, SnapTree provides access to any block of data within three pointers of reference, no matter how many snapshots are taken of a given file. Conversely, the methodology that legacy storage vendors use requires the user to collapse chains of snapshots, or drastically reduce the frequency of snapshots, in order to avoid the performance penalty associated with long metadata chains. SnapTree allows organizations to protect their data as frequently as they would like, and save those delta snapshots for any period of time without any performance penalty. Unlike traditional data protection technologies that require users to restore a full initial backup and each subsequent incremental backup in order to restore a particular file or object, SnapTree provides a virtual, fully hydrated image for every snapshot, enabling instant search-and-recover operations, regardless of the snapshot version in which that file or object resides.

When configuring the Cohesity Protection app, users have a few options when it comes to data reduction. Cohesity Protection provides a policy-driven, variable-size data deduplication engine, which is configurable to support inline, post-process, or no deduplication for a particular dataset. The benefits of this deduplication are shared globally across the entire cluster, maximizing storage efficiency.

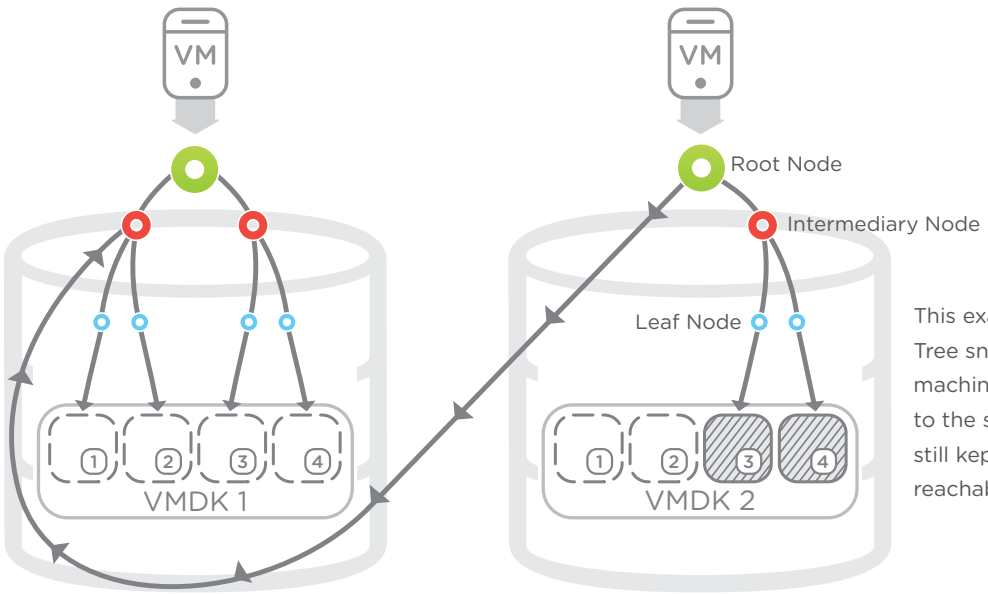
*[1] A B+ tree is an n-ary tree with a variable but often large number of children per node. A B+ tree consists of a root, internal nodes and leaves. The root may be either a leaf or a node with two or more children. (Wikipedia 10/2015)*

As the data is being streamed into the Cohesity Cluster, a background process automatically indexes the virtual machine environment along with the contents of the filesystem inside of each VM. This index powers a Google-like search box, enabling instant wildcard searches for any VM, file, or object protected by Cohesity Protection. This index is fully distributed across the entire cluster and is served from the cluster's flash memory, ensuring extremely fast access to the data in the index collections.

This Instant Search powers two key components of the Protection environment: File Restore and Virtual Machine Restore. In the case of a File Restore, users are presented with a list of filenames that match the search string. From this list, they can select the individual file or object they would like to recover from a particular VM. They can then select the point-in-time snapshot from which they would like to recover the file. In the case of a full Virtual Machine restore users search for a particular VM by name and are then presented with a list of snapshots associated with that VM. Once chosen, the instance of the VM is recovered and can then be cloned back to a given Resource Pool within the vCenter environment.

## Use Data Efficiently with Cohesity DevOps

In order to effectively leverage data that is stored on the Cohesity Data Platform, Cohesity provides a SnapTree-based, instant cloning capability. This requires no additional capacity and has no performance impact, so that users can quickly spin up an environment for test and development. These clones are created by taking another snapshot of a given VM, creating a new VMX file, and registering it with vCenter. In addition, a REST API is exposed to enable application-consistent SnapTree-based clones for other workflows.




This example of SnapTree shows how a Snap-Tree snapshot can be used to power a virtual machine. Only newly created blocks are written to the system, and the reference pointers are still kept in place. Again, all blocks of data are reachable within three hops.

Figure 6

## Gain Powerful Insight from Data with Cohesity Analytics

Leveraging Cohesity's powerful indexing capabilities, Cohesity Analytics provides organizations with intelligent analytics capabilities to derive business intelligence from their data. Native reporting capabilities include storage utilization trends, user metrics, and capacity forecasting, providing businesses with the information they need to anticipate future growth. Reports and real-time graphs of ingest rates, data reduction rates, IOPS, and latencies provide a holistic view of the performance and storage efficiency of a particular Cohesity Cluster.
In addition to the native reporting and analytics, Cohesity Analytics also includes Analytics Workbench, which allows users to inject custom code to run against a particular data set in the Cluster. This code leverages all the available compute and memory resources available, as well as the abstracted map/reduce functions, to quickly answer any query.

One of the first tools written for Analytics Workbench offers the ability to pattern-match across any file type for a pattern or phrase that exists inside of the file, providing a distributed GREP command for a Cohesity Cluster, regardless of the size.

## One Platform. Infinite Possibilities.

For far too long, organizations have been forced to deal with the complexity, cost, and overhead associated with managing multiple solutions from multiple vendors to achieve their business needs. Now, with Cohesity, organizations are able to eliminate data sprawl across their environment, reduce the complexity and cost of managing multiple solutions, and immediately benefit from the consolidation of multiple workloads onto a single platform. It's time to move to move away from legacy architectures, modernize enterprise IT, and bring order to the data chaos.

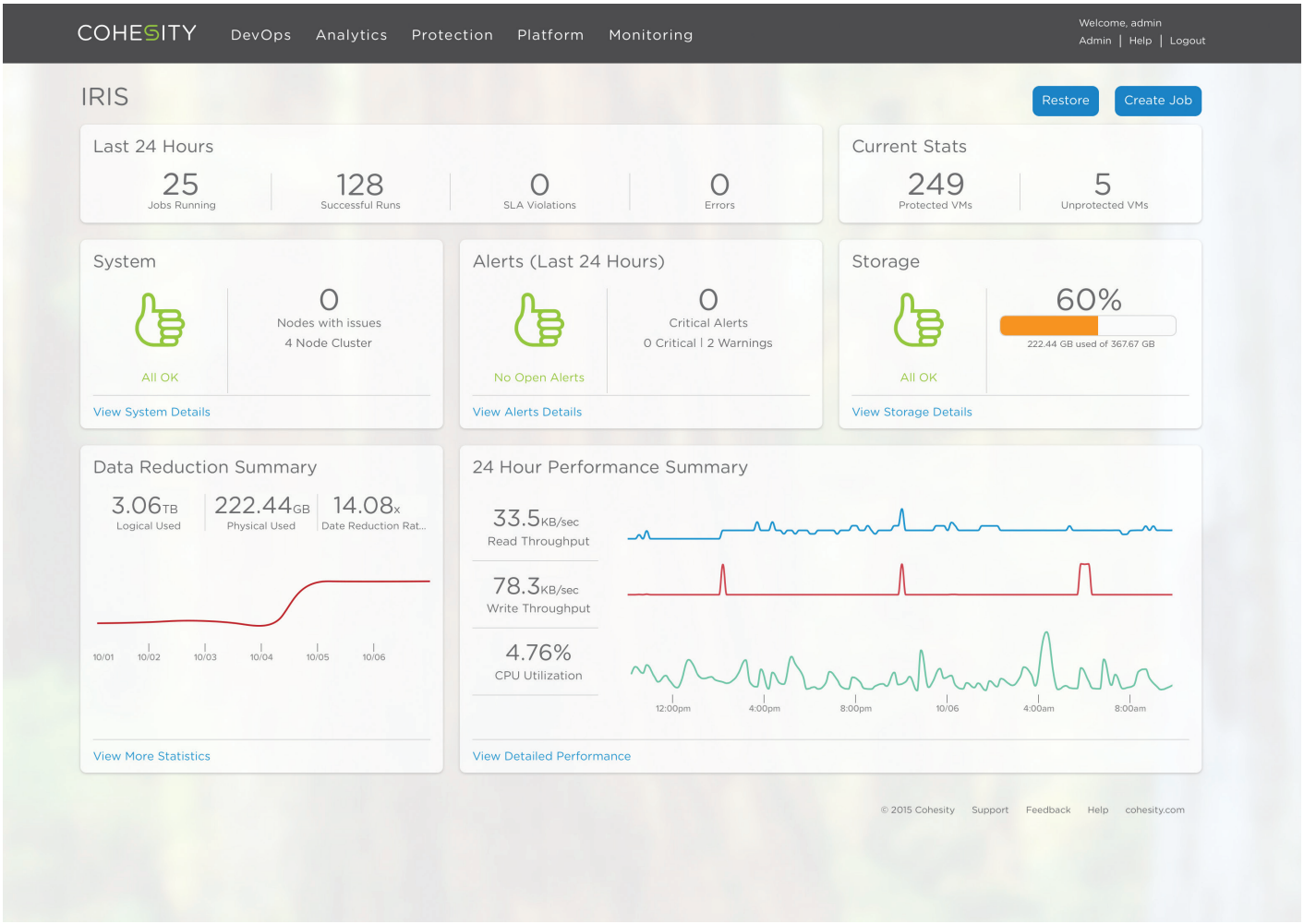



Figure 7