PURESTORAGE®
Uncomplicate Data Storage, Forever

komprise

TECHNICAL WHITE PAPER

# Transparent data-tiering between FlashBlade//S and FlashBlade//E with Komprise

Technical white paper to manage data intelligently

# Contents

**Executive Summary** 3

**Objectives and Scope** 3

**Data Management Challenges** 4

- Unstructured data explosion 4
- Data Compliance and retention requirements 4
- Legacy storage solutions 4
- Not all data are same 5
- Data Tiering challenges 5

**The Solution** 5

- Pure Storage FlashBlade 6
- Komprise 7

**Solution Architecture** 8

**Solution Benefits** 10

- Different tiering policy for different types of data 10
- Eliminates rehydration 10
- Access data in native format 10
- Access data from any tier without going to the source 10

**Solution Validation** 10

- Tests Overview 11
- Tiering NFS data on FB//S to NFS on FB//E 12
- Tiering NFS/SMB data on FB//S to Object on FB//E 14
- Accessing tiered data on NFS and Object on FB//E 15
- Recalling tiered data from FB//E to FB//S 16

**Conclusion** 17

**Appendix** 18

- Appendix A: FlashBlade configuration within Komprise 18

**Additional Resources** 25

**About the Author** 25

## Executive Summary

Unstructured data has been growing at an unprecedented level for the past several years. Industry analysts expect this rate of growth to continue exponentially in the years to come. This ever-growing data poses a variety of challenges, including data management, degraded application performance, increased backup time and size, and rising storage usage resulting in higher costs which don't match up to most IT budgets. Most organizations are spending 30% or more of their IT budget on data storage and backups, according to the Komprise 2023 State of Unstructured Data Management[1]. More than ever, the notion of "do more with less" resonates in IT organizations across sectors today.

In addition, this level of data growth is unsustainable using the current disk-based storage architecture. It simply requires too much space, energy, and resources to support. Legacy storage platforms have costly and disruptive refresh cycles, complicated data migration and poor data management capabilities.

Pure Storage FlashBlade® is a consolidated storage platform for the unstructured data, be it file or object, that is built for unlimited scale. FlashBlade//S™ is designed to deliver the efficiency, density, and top performance that modern unstructured data needs at scale. FlashBlade//E™ is designed to deliver the environmental, ease of use, and reliability benefits of all-flash storage for unstructured data workloads, but at a cost competitive with disk-based storage solutions. While performance-optimized FlashBlade//S and capacity-optimized FlashBlade//E can scale-out to petabytes of data, organization can be most cost-effective by keeping data at the most relevant FlashBlade based on where data is in its lifecycle.

Komprise, a SaaS for unstructured data management and mobility, was designed with the understanding that data is always in motion and should not be treated the same. Komprise gives enterprises the ability to intelligently manage their data by identifying rarely accessed data from FlashBlade//S and transparently tiering it to FlashBlade//E without any changes to the user or application access. With Komprise Transparent Move Technology (TMT™), users and applications access data in the same location as before with Komprise use of standard symbolic links (symlinks). The combination of Komprise Intelligent Data Management with the FlashBlade line of high-performance, resilient storage ensures the optimal cost/performance ROI in the industry.

Read this white paper to further understand the need for transparent data tiering, suggested architecture, solution validations and the benefits.

## Objectives and Scope

The purpose of the document is to design a solution for transparent data tiering between the performance-optimized FlashBlade//S and capacity-optimized FlashBlade//E using Komprise and demonstrate the benefits. While Komprise can support both Pure Storage FlashBlade and Pure Storage FlashArray™ Files as a source, the scope of this paper is limited to FlashBlade//S as the source and FlashBlade//E as the target.

# Data Management Challenges

## Unstructured data explosion

Analysts agree that the unstructured data growth is on the rise. The total volume of data created, captured, copied, and consumed globally is forecast to increase rapidly, reaching 180 zettabytes in 2025[2]. Another research from Enterprise Strategy Group[3] (ESG) indicates that unstructured data capacity will grow 10x by 2030. A variety of use cases are contributing to that growth. Regardless of whether it is data protection, video footage, documents, images or file and object repositories, unstructured data capacity is expected to be in high demand over the next several years. Burgeoning data growth is putting ample stress on the systems that manage the data and organizations are struggling to handle the unpredictable growth with existing budgets.

IT organizations are realizing the need for better data management capabilities, given that at least 80% of data becomes inactive and rarely accessed within a year of creation. This inactive data does not need to be managed on the same high-performance file storage such as FlashBlade//S and nor does it require the same level of data protection and backup strategy as the active data.

## Data Compliance and retention requirements

Data Compliance is the process of following various regulations and standards to maintain the integrity and availability of regulated and/or sensitive data. Adhering to ever-changing data compliance regulations and standards is a high priority for most CIOs. Data retention is an attribute of data compliance that defines the practice of storing and managing data and records for a designated period. Not all regulations have similar data retention, and it varies based on state and federal laws and industry regulations.

For example, the following well known regulations have varying data retention requirements based on the type of data:

- Health Insurance Portability and Accountability Act (**HIPAA**) requires covered entities and business associates to maintain required documentation for a minimum of six years but allows the state requirements (which varies from state to state) to preempt if the state has a shorter retention period.
- Sarbanes-Oxley Act of 2002 (**SOX**) was modified in 2003 to require relevant auditing and review documents to be retained for seven years after the audit or review of the financial statement is concluded.
- General Data Protection Regulation (**GDPR**) regulates how long sensitive data can be retained and how it will be disposed of when it is no longer in use.

Varying data retention based on the type of data requires a system to manage it seamlessly as the regulation requirement varies for every organization. Numerous organizations are using FlashBlade to host these data but based on the retention requirements, they want the flexibility to keep them either in the performance-optimized FlashBlade//S or capacity-optimized FlashBlade//E.

## Legacy storage solutions

Most unstructured data, almost 90%, lives on disk as that was the default option due to cost and a lack of viable, affordable alternatives. Given the projected growth rates of the data, leveraging legacy storage solutions using disks is not sustainable as it would require a massive data center footprint, significant energy to power the drives and numerous resources in terms of employees/contractors to manage them.

These legacy storage solutions are not evergreen and have a limited rack life which means they are refreshed periodically which is generally disruptive to the business. As well, the renewal contracts are extremely expensive and these storage solutions offer limited or zero data services such as compression to reduce the size of the data or encryption that protects data at rest.

### Not all data are same

Enterprise data value is defined by the frequency of its access by the business applications and users. For example, in the case of Security Information and Event Management (SIEM), the main functionality is to detect threats, anomalies real-time through collection and analysis of security events. Hence, the derived business value of the data is based on the recently ingested data or hot data, which is frequently accessed. As the data ages, most data is rarely accessed barring a security event such as data breach or audit.

A common problem in most enterprise environments, one which leads to high costs, is that most enterprise data is accessed infrequently over time but is still part of the regular backup and replication cycles along with the active data. This situation persists because there is no easy way to identify, move, and access data without impacting its availability to users. .

### Data Tiering challenges

As not all data are the same, they are managed through data tiering which is a technique of moving less frequently used data to cheaper levels of storage or tiers. Data tiering used to be in-built within storage systems to reduce the cost of data storage by tiering aged data within the storage array to cheaper but less performance media like SATA disks while the often-used data is kept on SSD or Flash drive which are expensive with higher performance characteristics. The tiering has now expanded to archiving where the data from storage systems are tiered to other cheaper storage systems or to cloud storage.

Some of the legacy data tiering solutions can address the data tiering functionality but they all come with challenges like cost, complexity, and operational challenges. Most of them require expensive enterprise licenses and upfront infrastructure investments. These legacy data tiering solutions have numerous components like storage agents, databases, hardware, and software components making the management very complex. Most of the legacy data tiering solutions are disruptive in nature for not maintaining transparent access to the tiered data or inducing performance slowdowns due to management overhead.

Finally, tiering solutions within storage systems are proprietary in nature. They tend to move the data at the block level aka block tiering, and they can only move data from the same storage operating system. Even when storage vendors extend this block tiering to tier data to the public cloud or to on-premises object storage, they still need the proprietary filesystem with metadata to access the file contents.

## The Solution

Pure Storage FlashBlade is a consolidated scale-out storage platform for unstructured data at scale designed specifically to handle the explosive data growth while addressing all the challenges with the legacy storage including: complexity, performance, lack of data services, and high total cost of ownership.

Komprise is an analytics-driven unstructured data management solution that moves the right data to the right place at the right time. Komprise, with its Global File Index, analyzes data and efficiently executes transparent data tiering between FlashBlade//S, performance-optimized scale-out storage array and FlashBlade//E, capacity-optimized storage built to handle exponential data growth.

Komprise Intelligent Data Management software complements the FlashBlade portfolio of products by providing transparent data tiering. Komprise software can intelligently identify data across FlashBlade//S and transparently move infrequently accessed data to more cost-efficient FlashBlade//E without disruptions to user or application access.

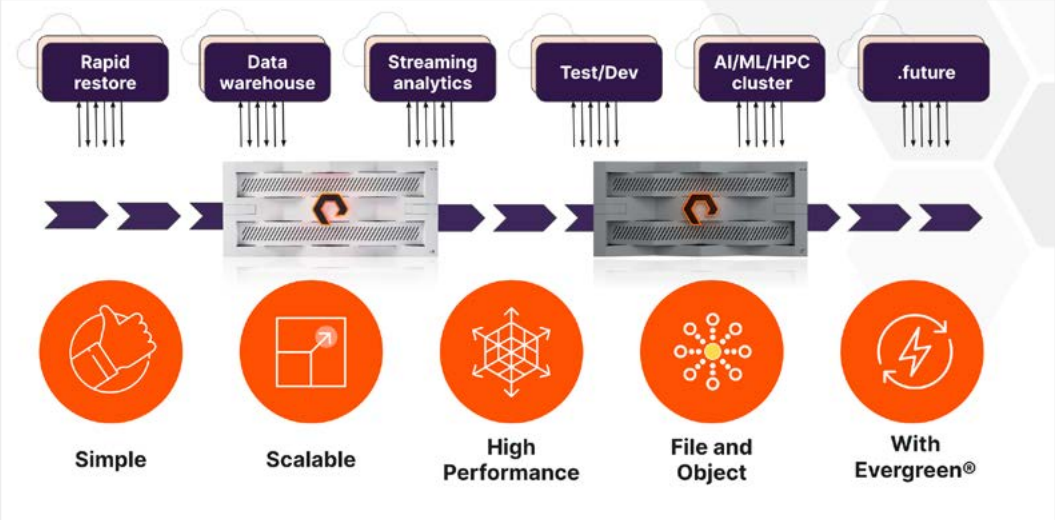
## Pure Storage FlashBlade

FlashBlade was built for unlimited scale that includes both file and object workloads delivering a simplified experience for infrastructure and data management while reducing the data center footprints, energy usage, and human resources to manage the storage systems.

FlashBlade was built with better engineering in mind. The chassis itself was built for long-life, or multiple generations of hardware, in mind while including maximum rack density along with power efficiency. It incorporated unified fast file and object storage with always-on encryption and data compression.

**Single Platform for Unstructured Data**

Built for Modern Fast File and Object Workloads.




The blade architecture of FlashBlade provides a modular unit of scale for both capacity and performance by using DirectFlash modules and Purity//FB, the operating system for FlashBlade. This allows a stateless, scale-out architecture for maximum resilience and scalable performance. FlashBlade is equipped with simplified software-defined high-speed networks and "set it and forget it" uplink configurations which means less cabling, less IP addresses requirement, a scalable uplink bandwidth, and out-of-band management.

True to Pure Storage's design traits specifically on "simplicity", FlashBlade can be managed very easily with intuitive GUI, interactive CLI, and REST-API that is both automation and DevOps friendly. The simplicity also eliminates any "nerd knobs" that other legacy storage vendors like to publish with gloat.

### FlashBlade//S

FlashBlade//S™ is a market-leading flash solution, designed to solve the complexity of data storage. It lets organizations use and grow cutting edge data capabilities—without growing complexity—through one seamless solution that brings together fast file and object data through hardware, software, and service. Its unique design allows capacity and performance to be updated independently and scale flexibly, alongside needs. IT teams can upgrade the solution as needed to protect investments, maintain availability, reduce waste, and keep data secure to match the organization's goals and market strategies.

### FlashBlade//E

FlashBlade//E™ is an all-flash scale-out unstructured data repository built to handle exponential file and object data growth. Delivered at a comparable price to disk-based solutions, its all-flash architecture delivers a denser, more energy-efficient and highly scalable unstructured data storage solution that can better support everyday repository workloads.

Designed for seamless and non-disruptive hardware and software innovation with predictable support, cost and experience, FlashBlade//E improves data access and meets changing business needs. With optimized workloads, lower energy consumption, denser footprint, and a best-in-class user experience, customers can drive down long-term costs and easily scale to support growth without additional floor space, power, cooling, and administrative resources.

## Komprise

Komprise Intelligent Data Management is a standards-based, modern data management solution built to help IT and storage managers get fast insights on data and create plans for automated policy-based movement of data. Komprise is unique in the market with its patented Transparent Move Technology (TMT) for non-disruptive transparent data tiering and native access to data at every tier without proprietary lock-in. The analytics-driven approach works across any file and object storage, cloud and on-premises, to give holistic visibility and a single consistent way to manage data.

Following are some of the key benefits of Komprise while supporting tiering between FlashBlade//S and FlashBlade//E.

- **Simple**—Komprise is simple to deploy and operate and it requires no proprietary interfaces or complex infrastructure setup.
- **Open**—Komprise works using open standards like NFS, SMB/CIFS and REST/S3, without the use of proprietary stub files or agents.
- **Analytics-Driven**—Komprise shows analytics on data volumes, data usage, data owners, file types and data growth to optimally manage data for cost and value based on changing requirements and needs. Komprise Storage Insights is a new console introduced in 2023, combining data-centric and storage-centric metrics to support faster decision making.
- **Transparent**—Komprise Transparent Move Technology (TMT) moves data transparently which means the data is fully accessible from the source as files, exactly as before, and the data is accessible as files or objects from the target. This gives owners control of their, no matter where it resides.
- **Elastic Scaling**—Komprise scales elastically on-demand with no bottlenecks, databases, or servers to limit scalability.
- **No Lock-in**—Data is always accessible from the source and target storage, even if Komprise is taken offline.
- **Non-Intrusive**—Komprise analyzes and manages data in the background, with no impact to storage and network performance, and outside the hot data and metadata paths.
- **Adaptive**—Komprise throttles back as needed when the storage or network are in active use eliminating the need to monitor or schedule when Komprise runs.

### Komprise Transparent Move Technology (TMT)

Komprise, with its patented Transparent Move Technology (TMT), moves data transparently using standard file system constructs. To provide a storage and cloud agnostic architecture, Komprise avoids the use of static stubs and agents. It tiers files such that users can access tiered files from their original location as files, while the data resides as objects in the target system. Komprise provides this file-to-object translation without requiring rehydration back to the source. Files tiered by Komprise are also accessible as native objects from the target system without going through Komprise or the original file storage, so there is no lock-in. Finally, Komprise does not get in front of hot data access.

## Solution Architecture

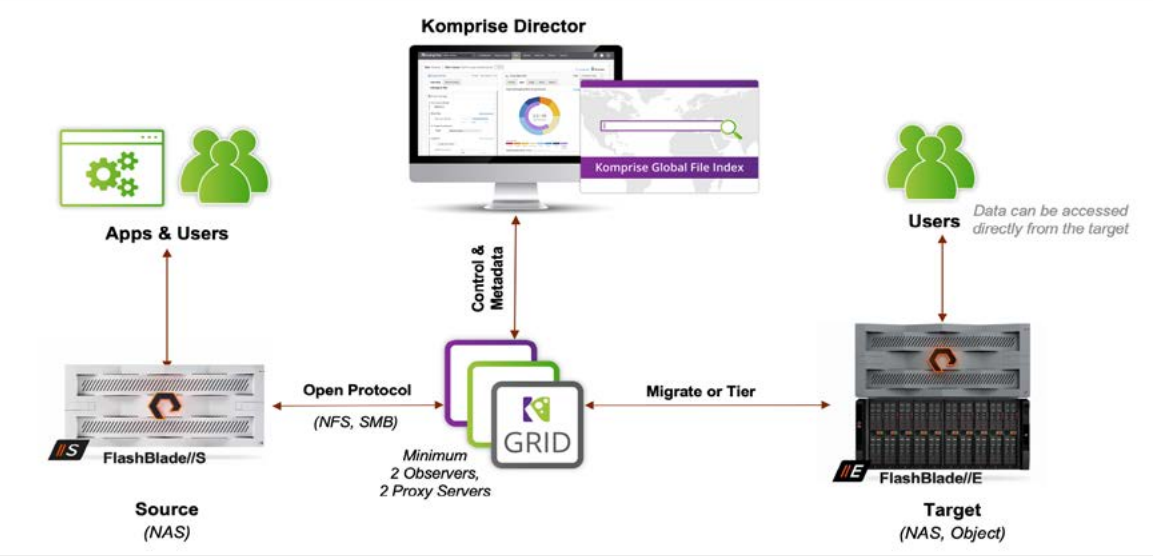
Our solution was designed to take advantage of Komprise Intelligent Data Management that can analyze, tier, and manage data at scale between FlashBlade//S and FlashBlade//E without any proprietary interfaces.

Following are the key software and hardware components and technologies included in this solution:

- Komprise Intelligent Data Management
- Komprise Grid
  - Director—Administrative interface to manage the file/object storage and data movement policies (as-a-service or on-prem).
  - Observer—Virtual appliance that performs data analysis, data movement and manages Proxy system transfers.
  - Windows Proxy Service—Assist with SMB data movement.
- Pure Storage FlashBlade//S as the source
- Pure Storage FlashBlade//E as the target

The following diagram illustrates the architecture to tier the infrequently accessed data out of FlashBlade//S to FlashBlade//E automatically through Komprise.

**Solution Architecture**




### Software Configuration

| Component | Details |
|---|---|
| **Komprise Intelligent Data Management** | 5.1.0 |
| **FlashBlade Purity Software** | 4.2.1 |

### Komprise Grid Architecture

The Komprise Elastic Grid is a distributed, shared-nothing, scale-out software architecture to manage data at massive scale. The architecture consists of the following components:

- **Director**: The user interface console where administrative users configure and monitor their unstructured data management policies, and which manages the Komprise Grid to perform the desired data movements. The Komprise Director can be deployed as a Komprise-hosted service on the cloud or as a stand-alone virtual appliance within the data center.
- **Observer**: Observers are virtual appliances that perform data analysis, data movement and manage Windows Proxies. Observers communicate with the Director to provide summary Plan Analysis information, maintain data management policies, and report error information. When the Deep Analytics feature is enabled, Observers also send metadata to the Deep Analytics query and indexing services.
- **Windows Proxy**: Assist with the SMB data movement. Komprise utilizes a set of services installed on a virtual Windows Server to handle Observer communication and data movement. The Windows Proxies ensure fast and correct transfers of SMB ACLs, extended attributes, and data.

### Komprise Analysis

Komprise Analysis provides meaningful insights into the unstructured data hosted on FlashBlade//S. Komprise quickly analyzes data residing in NFS, SMB or Dual/Mixed shares and provides useful information such as: how much data there is, how fast data is growing, the types of files and sizes as well as information on how much data is actively used and how much is inactive.

Organizations can use these insights to build actionable data management plans. These include reducing data costs significantly by transparently tiering infrequently used data to FlashBlade//E. The ability to tier data without disrupting users or applications is essential to ensuring uninterrupted business continuity.

During the source data store setup, Komprise provides preliminary estimated results within minutes after enabling shares. The estimated results are continually refined until the scanning completes. Komprise scans only the metadata of the file systems. File size and overall capacity do not affect the scan rate as Komprise does not open files to read the data within them. The analysis scan uses adaptive scan rate technology to not overwhelm the FlashBlade//S.

## Solution Benefits

### Different tiering policy for different types of data

Komprise along with Pure Storage FlashBlade offers different tiering policies for different types of data. With this, users have the flexibility to pick and choose the type of data based on the application hosted on NFS or SMB shares on FlashBlade//S to be tiered based on access timings, file size to NFS or S3 bucket on FlashBlade//E. For example, if an EDA type application uses NFS shares on FlashBlade//S requires tiering to a NFS target on FlashBlade//E, an administrator can create a Komprise tiering policy with access timings and file size filters along with the destination of a NFS share on the target. With another application, such as log analytics that has data on NFS shares on FlashBlade//S, the administrator can create a policy to tier data to object store on FlashBlade//E with relevant filters along with the S3 bucket destination. Users have the flexibility to choose the shares and the target locations for every tiering policy within Komprise.

### Eliminates rehydration

Traditional block-level tiering, used by many data management solutions, requires rehydrating the tiered data before it can be used, migrated, or backed up. Rehydration incurs licensing costs on the primary storage and neutralizes the benefit of data tiering altogether. Komprise eliminates this with the use of symbolic links and Komprise Access Address that points to the tiered data on the target FlashBlade//E. Backup software will save the symbolic links without rehydrating the files they point to, and restores will restore the links, which still point to the same files, so third-party backup applications will function without rehydration.

### Access data in native format

With Komprise, when inactive data is tiered on FlashBlade//S, it is written in a format that is native to FlashBlade, which ensures that the tiered data is not locked away and can be reused to extract value. Komprise preserves all the standard and extended metadata, so the files retain their full context wherever they are.

### Access data from any tier without going to the source

When Komprise tiers data from FlashBlade//S to FlashBlade//E, users can directly access it on FlashBlade//E using its native protocol. For example, if an NFS share on FlashBlade//S is tiered to a S3 bucket on FlashBlade//E, users can transparently access it on FlashBlade//S using NFS and directly on FlashBlade//E using standard S3 tools like s5cmd. This can be an added benefit if users wanted to access the aged data that are tiered on FlashBlade//E to be used with their data lake for AI/ML applications without going through FlashBlade//S to transparently access them.

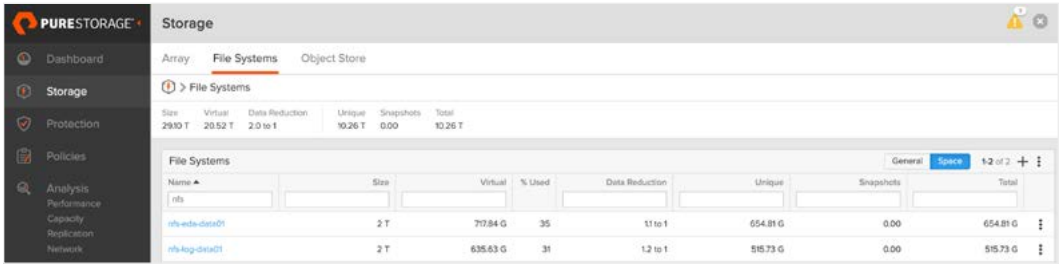
## Solution Validation

The following tests were performed as part of the Solution validation to ensure the source data on FlashBlade//S was tiered automatically to FlashBlade//E based on a given aging policy and have seamless access to the data that were tiered.

- Tiering NFS data on FB//S to NFS on FB//E
- Tiering NFS/SMB data on FB//S to Object on FB//E
- Accessing tiered data on NFS and Object on FB//E
- Recalling tiered data from FB//E to FB//S

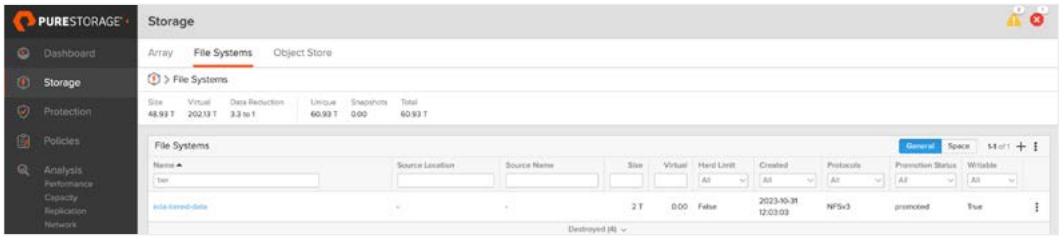
## Tests Overview

The synthetic data used for the NFS-to-NFS test were created using SPECfs toolkit with type EDA and hosted them on a network filesystem (`nfs-eda-data01`) on the source FlashBlade//S. Another dataset involving log-analytics data like apache logs used for NFS-to-Object test was hosted on a second network filesystem (`nfs-log-data01`) on the same FlashBlade//S.

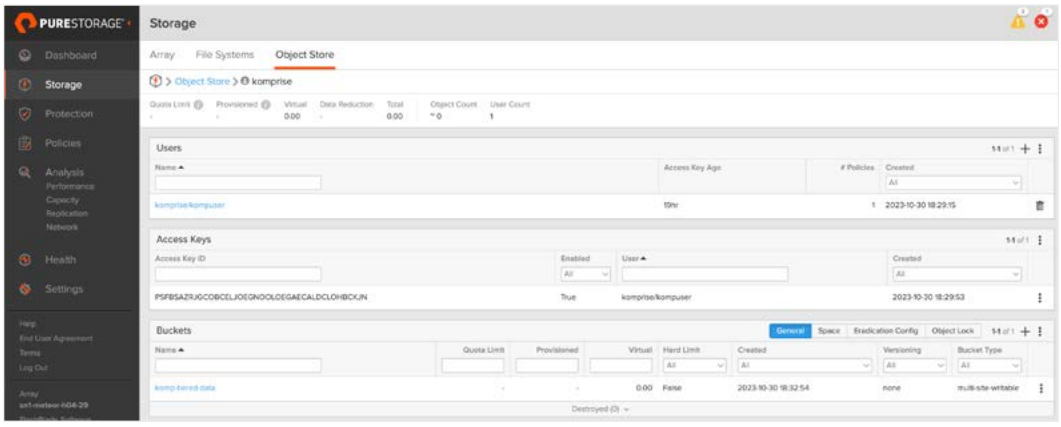



Once Komprise is deployed, the source and target FlashBlades were configured in Komprise Director by creating the relevant data stores. The first data store pointed to FlashBlade//S (using the data VIP) as the source. At this point the two NFS shares were enabled to be periodically analyzed by Komprise to provide meaningful insights from the data.

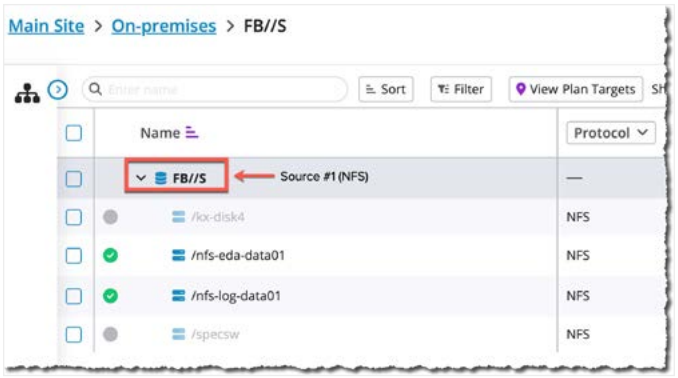
The second data store pointed to FlashBlade//E with a NFS share as the target. The following image shows the NFS share on the FlashBlade//E.

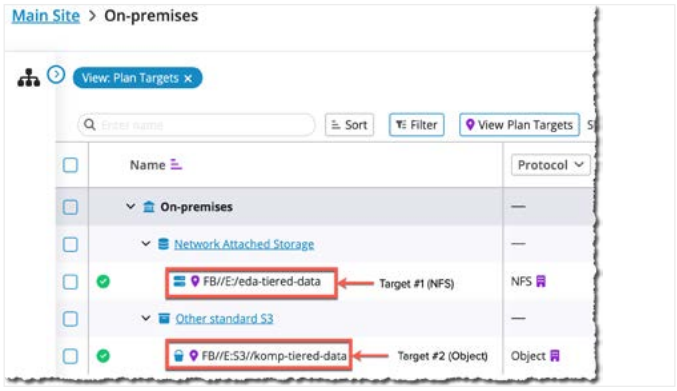



A third data store was created which again pointed to FlashBlade//E but with a S3 bucket as the target. The following image shows the S3 bucket on the target, FlashBlade//E.



The image below on the left shows the source, FlashBlade//S, configured as the data store with two NFS shares enabled under Komprise. The image below on the right shows the two target data stores, one with the NFS share and the other one with the S3 bucket from the FlashBlade//E.

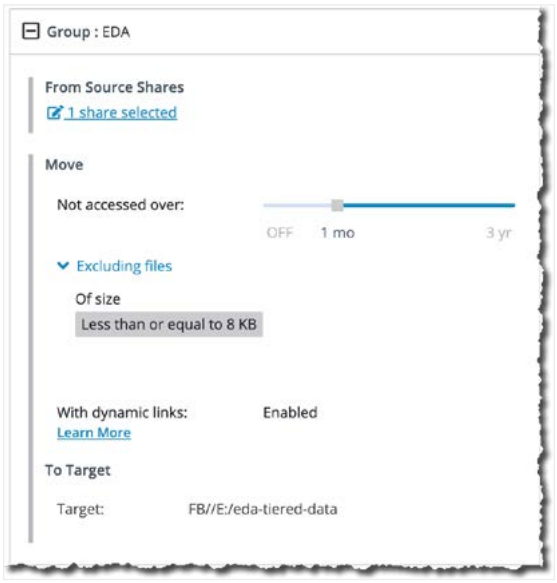


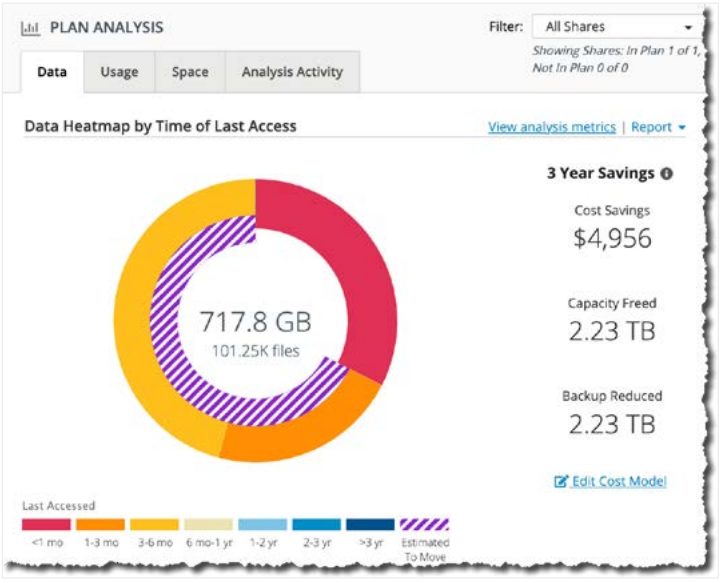




## Tiering NFS data on FB//S to NFS on FB//E

Tiering from a source to target on Komprise is done through what is identified as "Plan". A Komprise Plan allows automated, analytics-driven policies for tiering, copying or confining data from selected NAS sources to desired targets.

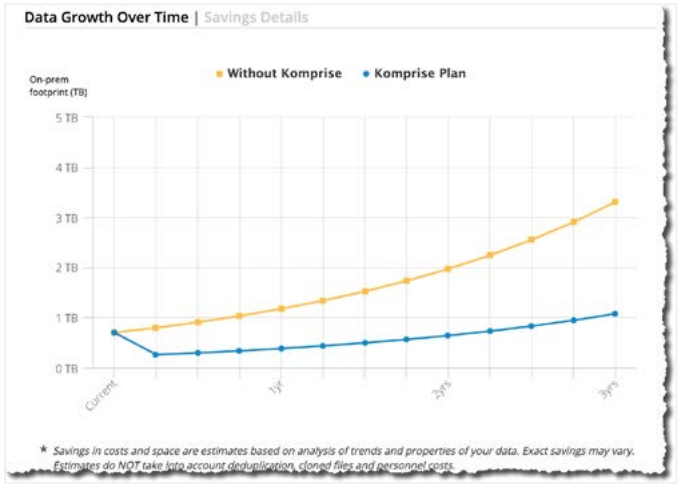
For this test to tier data from a NFS source to a NFS target, a new group named **EDA** was created under the Plan and the share "`nfs-eda-data01`" was selected. The tiering policy was defined under the group where the access time limit and the target were selected. In this case, the access time limit was set to 1 month which identified all the files that were not accessed over a month and marked them to be tiered or moved to the NFS target on FlashBlade//E. There is also a filter option to limit or exclude files by size.



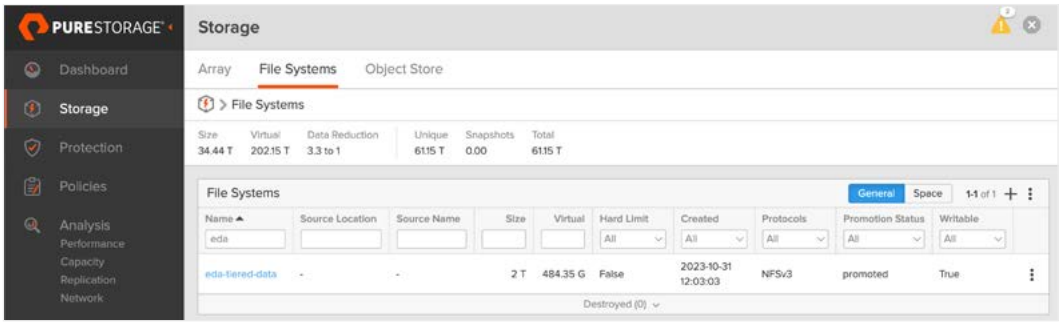




Once the Plan was activated, Komprise initiates the transparent data tiering where it moves the data from FlashBlade//S to the NFS filesystem on FlashBlade//E. The time to move the data from FB//S to FB//E depends on the network bandwidth on the Komprise Observers as well as the Observers' VM resource settings like the CPU and memory.

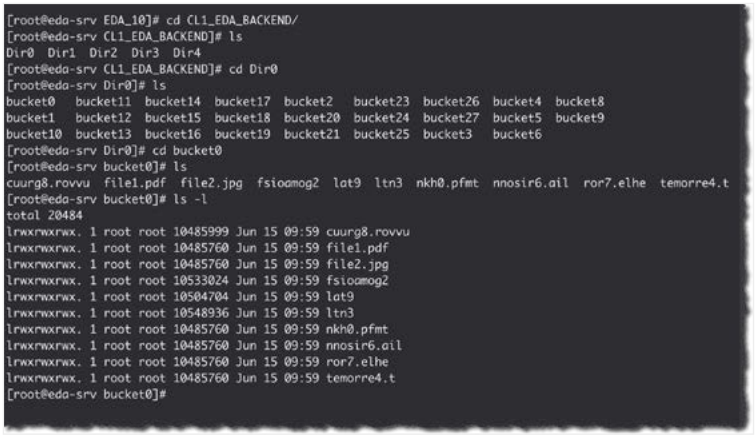
Komprise Plan Analysis dashboard shows more details about the data tiered.

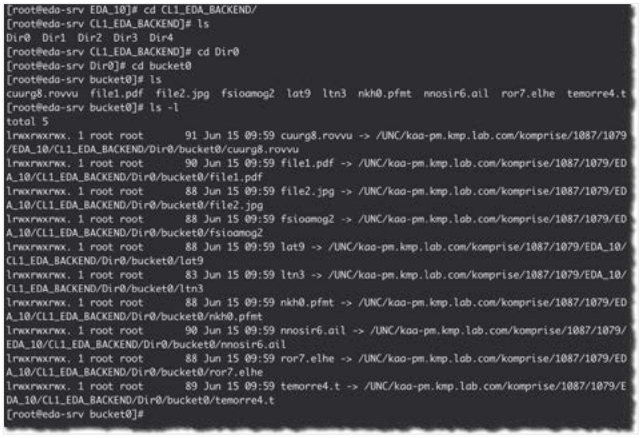

The following image shows the tiered data on the target, FlashBlade//E.




The following screenshots shows a before and after view of a directory that was part of the tiering. The screenshot on the left shows the actual content of a directory prior to tiering which had 10 files, each around 10MB in size. The screenshot on the right shows the content of the directory after the tiering where all files are now a symbolic link pointing to the original files that are on the FlashBlade//E.



```
[root@eda-srv EDA_10]# cd CL1_EDA_BACKEND/
[root@eda-srv CL1_EDA_BACKEND]# ls
Dir0  Dir1  Dir2  Dir3  Dir4
[root@eda-srv CL1_EDA_BACKEND]# cd Dir0
[root@eda-srv Dir0]# ls
bucket0   bucket11  bucket14  bucket17  bucket2   bucket23  bucket26  bucket4  bucket8
bucket1   bucket12  bucket15  bucket18  bucket20  bucket24  bucket27  bucket5  bucket9
bucket10  bucket13  bucket16  bucket19  bucket21  bucket25  bucket3   bucket6
[root@eda-srv Dir0]# cd bucket0
[root@eda-srv bucket0]# ls
cuurg8.rovvu  file1.pdf  file2.jpg  fsioamog2  lat9  ltn3  nkh0.pfmt  nnosir6.ail  ror7.elhe  temorre4.t
[root@eda-srv bucket0]# ls -l
total 20484
lrwxrwxrwx. 1 root root 10485999 Jun 15 09:59 cuurg8.rovvu
lrwxrwxrwx. 1 root root 10485760 Jun 15 09:59 file1.pdf
lrwxrwxrwx. 1 root root 10485760 Jun 15 09:59 file2.jpg
lrwxrwxrwx. 1 root root 10533024 Jun 15 09:59 fsioamog2
lrwxrwxrwx. 1 root root 10504704 Jun 15 09:59 lat9
lrwxrwxrwx. 1 root root 10548936 Jun 15 09:59 ltn3
lrwxrwxrwx. 1 root root 10485760 Jun 15 09:59 nkh0.pfmt
lrwxrwxrwx. 1 root root 10485760 Jun 15 09:59 nnosir6.ail
lrwxrwxrwx. 1 root root 10485760 Jun 15 09:59 ror7.elhe
lrwxrwxrwx. 1 root root 10485760 Jun 15 09:59 temorre4.t
[root@eda-srv bucket0]#
```

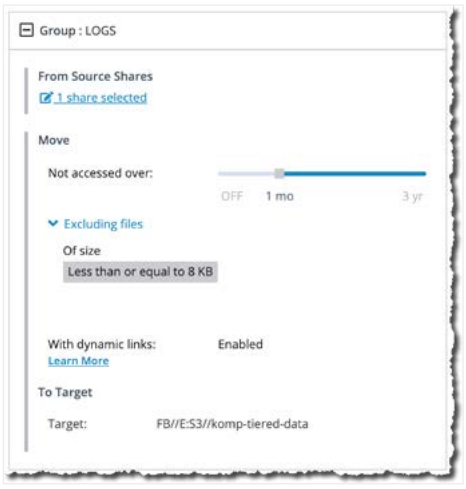


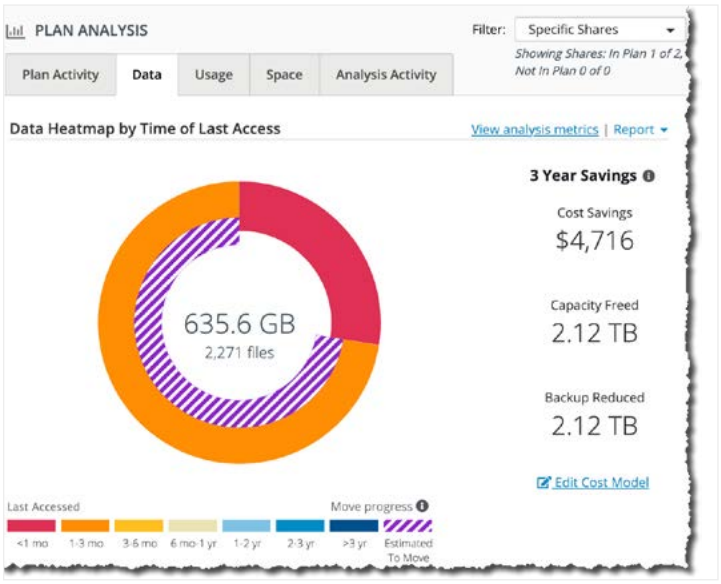
```
[root@eda-srv EDA_10]# cd CL1_EDA_BACKEND/
[root@eda-srv CL1_EDA_BACKEND]# ls
Dir0  Dir1  Dir2  Dir3  Dir4
[root@eda-srv CL1_EDA_BACKEND]# cd Dir0
[root@eda-srv Dir0]# cd bucket0
[root@eda-srv bucket0]# ls
cuurg8.rovvu  file1.pdf  file2.jpg  fsioamog2  lat9  ltn3  nkh0.pfmt  nnosir6.ail  ror7.elhe  temorre4.t
[root@eda-srv bucket0]# ls -l
total 5
lrwxrwxrwx. 1 root root       91 Jun 15 09:59 cuurg8.rovvu -> /UNC/kaa-pm.kmp.lab.com/komprise/1087/1079
/EDA_10/CL1_EDA_BACKEND/Dir0/bucket0/cuurg8.rovvu
lrwxrwxrwx. 1 root root       90 Jun 15 09:59 file1.pdf -> /UNC/kaa-pm.kmp.lab.com/komprise/1087/1079/ED
A_10/CL1_EDA_BACKEND/Dir0/bucket0/file1.pdf
lrwxrwxrwx. 1 root root       88 Jun 15 09:59 file2.jpg -> /UNC/kaa-pm.kmp.lab.com/komprise/1087/1079/ED
A_10/CL1_EDA_BACKEND/Dir0/bucket0/file2.jpg
lrwxrwxrwx. 1 root root       88 Jun 15 09:59 fsioamog2 -> /UNC/kaa-pm.kmp.lab.com/komprise/1087/1079/ED
A_10/CL1_EDA_BACKEND/Dir0/bucket0/fsioamog2
lrwxrwxrwx. 1 root root       88 Jun 15 09:59 lat9 -> /UNC/kaa-pm.kmp.lab.com/komprise/1087/1079/EDA_10/
CL1_EDA_BACKEND/Dir0/bucket0/lat9
lrwxrwxrwx. 1 root root       83 Jun 15 09:59 ltn3 -> /UNC/kaa-pm.kmp.lab.com/komprise/1087/1079/EDA_10/
CL1_EDA_BACKEND/Dir0/bucket0/ltn3
lrwxrwxrwx. 1 root root       88 Jun 15 09:59 nkh0.pfmt -> /UNC/kaa-pm.kmp.lab.com/komprise/1087/1079/ED
A_10/CL1_EDA_BACKEND/Dir0/bucket0/nkh0.pfmt
lrwxrwxrwx. 1 root root       90 Jun 15 09:59 nnosir6.ail -> /UNC/kaa-pm.kmp.lab.com/komprise/1087/1079/
EDA_10/CL1_EDA_BACKEND/Dir0/bucket0/nnosir6.ail
lrwxrwxrwx. 1 root root       88 Jun 15 09:59 ror7.elhe -> /UNC/kaa-pm.kmp.lab.com/komprise/1087/1079/ED
A_10/CL1_EDA_BACKEND/Dir0/bucket0/ror7.elhe
lrwxrwxrwx. 1 root root       89 Jun 15 09:59 temorre4.t -> /UNC/kaa-pm.kmp.lab.com/komprise/1087/1079/E
DA_10/CL1_EDA_BACKEND/Dir0/bucket0/temorre4.t
[root@eda-srv bucket0]#
```

At this point, the users and applications can still access the contents at the source which was one of the key requirements with the transparent data tiering.

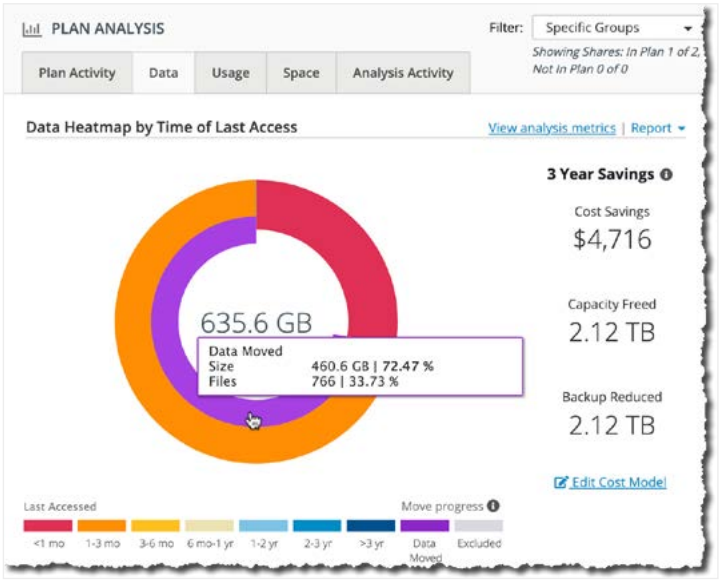
## Tiering NFS/SMB data on FB//S to Object on FB//E

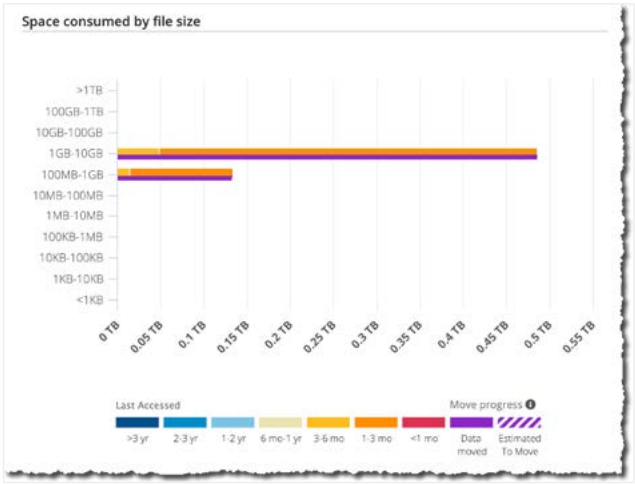
As like the prior test, to test data from a NFS source to an Object target, a new group named **LOGS** was created under the same Plan and the share "`nfs-log-data01`" was selected. The tiering policy included the access time limit at 1 month and excluded files that were sized below 8KB. This identified all the files that were over 8KB and were not accessed over a month and marked them to be tiered or moved to the object target on FlashBlade//E.



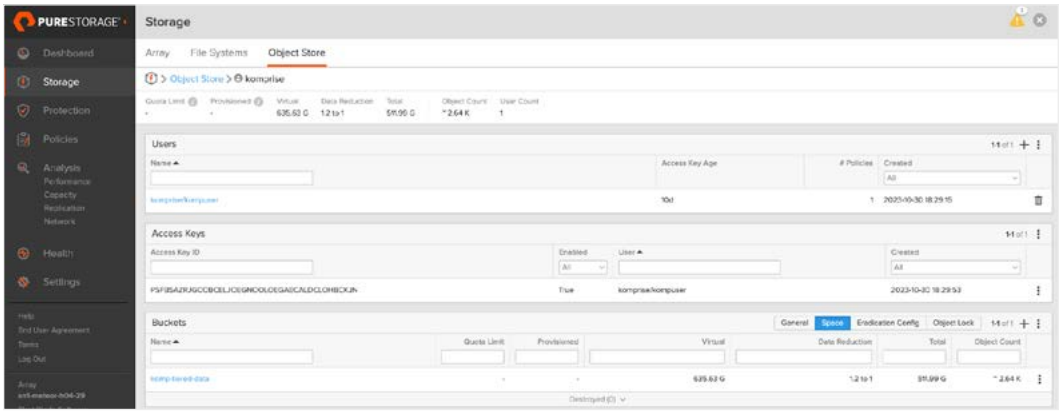




Once the plan was activated, the data was tiered from the "`nfs-logs-data01`" share on FlashBlade//S to the S3 bucket "`komp-tiered-data`" on the target FlashBlade//E. The Komprise Plan Analysis dashboard for this group shows the details.



The following image shows the tiered data on the S3 bucket on the FlashBlade//E.




Like the prior test, the following screenshots show a before and after view of a directory that was part of the tiering. The screenshot on the left shows the actual content of a directory prior to tiering which had 4 files that were over 8KB in size and over 3 months old. The screenshot on the right shows the content of the same directory after the tiering where the 4 files are now a symbolic link pointing to the original files that are on the FlashBlade//E.

```
[root@splk-ix06 db_1696154399_1696149000_1_728B577E-A0C8-4FB3-B9A2-D0CECC0EFCBB]# df -h .
Filesystem                Size  Used Avail Use% Mounted on
10.21.236.98:/nfs-log-data01  2.0T  636G  2.0T  31% /log-data
[root@splk-ix06 db_1696154399_1696149000_1_728B577E-A0C8-4FB3-B9A2-D0CECC0EFCBB]# ls -l
total 849035
-rw-------. 1 root root  869406186 Aug  1 13:36 1696151364-1696149000-1524861890742634264 3.tsidx
lrwxrwxrwx. 1 root root 1115709540 Aug  1 13:36 1696154399-1696151364-1570524333341319278 1.tsidx
lrwxrwxrwx. 1 root root     123164 Aug  1 13:36 bloomfilter
-rw-------. 1 root root         75 Aug  1 13:36 bucket_info.csv
-rw-------. 1 root root        110 Aug  1 13:36 Hosts.data
lrwxrwxrwx. 1 root root     657450 Aug  1 13:36 merged_lexicon.lex
-rw-------. 1 root root         49 Aug  1 13:36 optimize.result
drwx------. 2 root root          0 Oct 31 15:08 rawdata
-rw-------. 1 root root        136 Aug  1 13:36 Sources.data
-rw-------. 1 root root        122 Aug  1 13:36 SourceTypes.data
-rw-------. 1 root root         98 Aug  1 13:36 splunk-autogen-params.dat
-rw-------. 1 root root        283 Aug  1 13:36 Strings.data
[root@splk-ix06 db_1696154399_1696149000_1_728B577E-A0C8-4FB3-B9A2-D0CECC0EFCBB]#
```

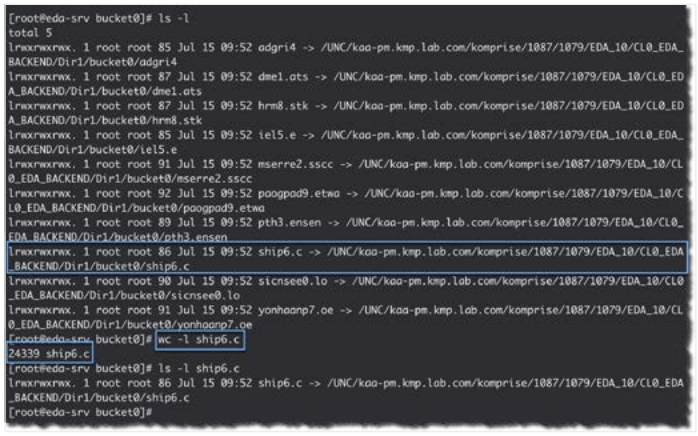
```
[root@splk-ix06 db_1696154399_1696149000_1_728B577E-A0C8-4FB3-B9A2-D0CECC0EFCBB]# df -h .
Filesystem                Size  Used Avail Use% Mounted on
10.21.236.98:/nfs-log-data01  2.0T  175G  2.0T   8% /log-data
[root@splk-ix06 db_1696154399_1696149000_1_728B577E-A0C8-4FB3-B9A2-D0CECC0EFCBB]# ls -l
total 6
lrwxrwxrwx. 1 root root       172 Aug  1 13:36 1696151364-1696149000-1524861890742634264 3.tsidx -> /UNC
/kaa-pm.kmp.lab.com/komprise/1091/1111/splunk/apache-log/db_1696154399_1696149000_1_728B577E-A0C8-4FB3-
B9A2-D0CECC0EFCBB/1696151364-1696149000-1524861890742634264 3.tsidx
lrwxrwxrwx. 1 root root       173 Aug  1 13:36 1696154399-1696151364-1570524333341319278 1.tsidx -> /UNC
/kaa-pm.kmp.lab.com/komprise/1091/1111/splunk/apache-log/db_1696154399_1696149000_1_728B577E-A0C8-4FB3-
B9A2-D0CECC0EFCBB/1696154399-1696151364-1570524333341319278 1.tsidx
lrwxrwxrwx. 1 root root       136 Aug  1 13:36 bloomfilter -> /UNC/kaa-pm.kmp.lab.com/komprise/1091/111
1/splunk/apache-log/db_1696154399_1696149000_1_728B577E-A0C8-4FB3-B9A2-D0CECC0EFCBB/bloomfilter
-rw-------. 1 root root        75 Aug  1 13:36 bucket_info.csv
-rw-------. 1 root root       110 Aug  1 13:36 Hosts.data
lrwxrwxrwx. 1 root root       143 Aug  1 13:36 merged_lexicon.lex -> /UNC/kaa-pm.kmp.lab.com/komprise/1
091/1111/splunk/apache-log/db_1696154399_1696149000_1_728B577E-A0C8-4FB3-B9A2-D0CECC0EFCBB/merged_lexic
on.lex
-rw-------. 1 root root        49 Aug  1 13:36 optimize.result
drwx------. 2 root root         0 Oct 31 15:08 rawdata
-rw-------. 1 root root       136 Aug  1 13:36 Sources.data
-rw-------. 1 root root       122 Aug  1 13:36 SourceTypes.data
-rw-------. 1 root root        98 Aug  1 13:36 splunk-autogen-params.dat
-rw-------. 1 root root       283 Aug  1 13:36 Strings.data
[root@splk-ix06 db_1696154399_1696149000_1_728B577E-A0C8-4FB3-B9A2-D0CECC0EFCBB]#
```

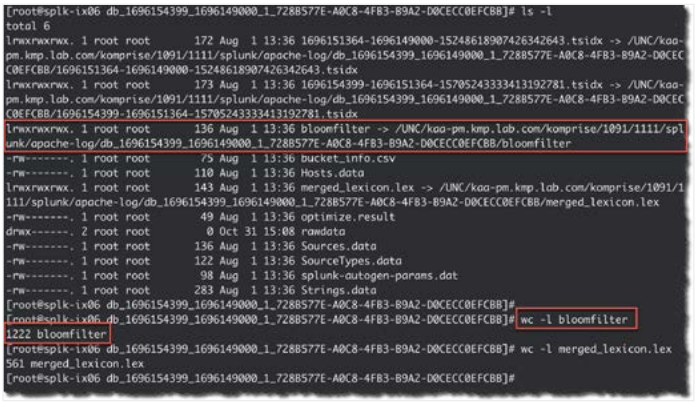
**NOTE:** *The above procedure can be followed to tier SMB data hosted on FlashBlade//S to the object store on FlashBlade//E. The SMB shares should be added as part of the data store setup (see Appendix-A) and included in the Plan for the SMB data to be tiered.*

## Accessing tiered data on NFS and Object on FB//E

To validate that the data is indeed accessible by the users/applications on the source, wordcount (wc) command was run on the files that were tiered.

Irrespective of the target being on the NFS share or on the Object (S3 bucket), the command ran successfully accessing the data from the target FlashBlade//E. The following image on the left shows the contents from the "`nfs-eda-data01`" share that was tiered into a NFS share on FlashBlade//E while the image on the right shows the contents from the "`nfs-log-data01`" share that was tiered into the S3 bucket on FlashBlade//E.



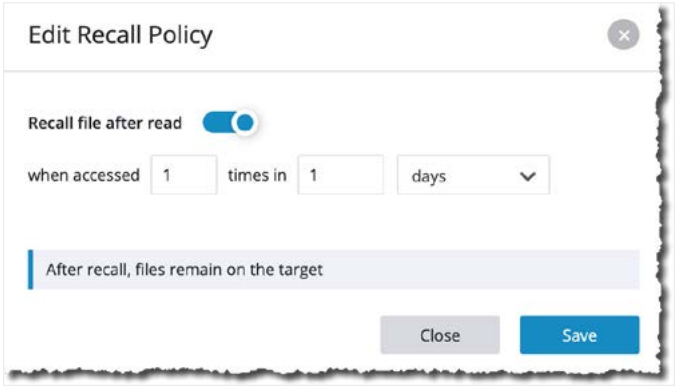

The access to the tiered data was possible with **Komprise Access Address** which is a DNS hostname that transparently accesses the data transferred by Komprise.

## Recalling tiered data from FB//E to FB//S

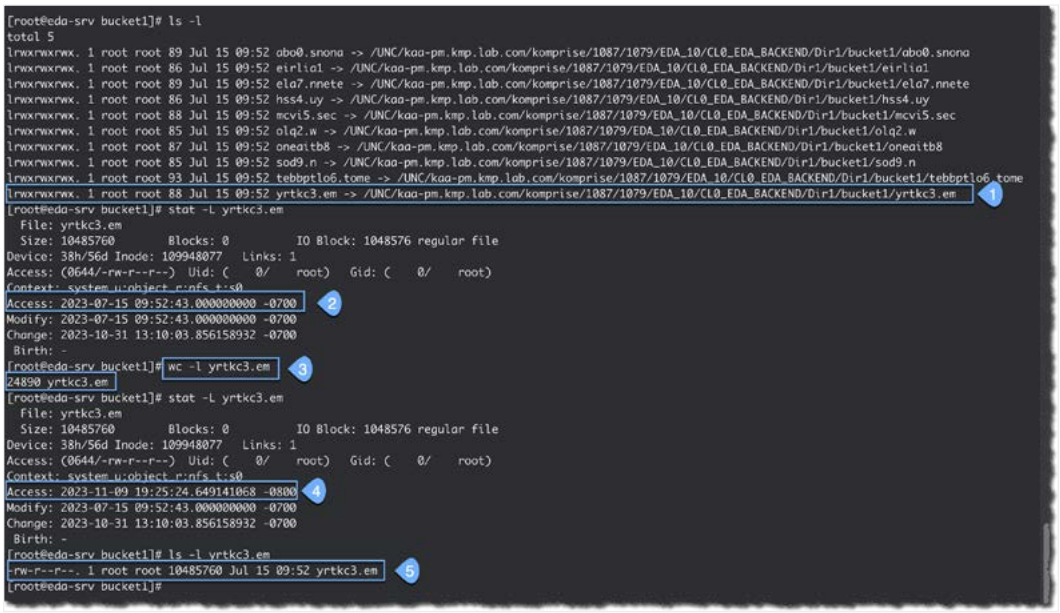
The tiered data can be recalled to the source share either through a bulk recall or through a recall policy. The bulk recalls can recall either an entire share or just a specific directory in a share. In contrast to the bulk recall, the recall policy recalls a single file based on the access.

In this test, a recall policy was configured to recall a file after 1 time it was accessed in 1 day.

**NOTE:** *This is not the typical setting where a file is recalled immediately after its first access in a day. Users should exercise caution in setting this policy that adheres to their application access usage against the tiered data.*




Once the policy was configured, the tiered file was accessed.




1. Listing the directory content showed that the tiered file yrtkc3.em was indeed a symbolic link.
2. Running a stat command on the linked file showed the access date as July 15th, 2023.
3. The wordcount command was run on the tiered file which accessed the original file from FlashBlade//E and displayed the results.
4. Running a stat command again on the linked file showed the access date indeed changed to the latest date, Nov 9th, 2023.
5. This prompted Komprise to recall as per the recall policy and the list command indeed shows the recalled file without any symbolic link.

## Conclusion

Based on the tests, Komprise Intelligent Data Management indeed transparently tiered data from FlashBlade//S to FlashBlade//E without any proprietary interfaces.

FlashBlade//S solves the complexity of data storage while allowing the organizations to use and grow cutting edge data capabilities. FlashBlade//E is quickly becoming the scale-out unstructured data repository to handle exponential file and object data growth while keeping the price comparable to disk-based solutions but with improved energy efficiency.

Komprise along with both FlashBlade//S and FlashBlade//E offers a unique proposition for enterprises to tier their inactive out of high-performance, top tier FlashBlade//S to the capacity and cost optimized FlashBlade//E, with the following benefits.

- Zero disruption to users and applications for tiered data. Simply find the data at the link they always have used.
- Eliminates any rehydration and associated cost.
- Access data from any tier without going to the source.
- Access data in native format.
- Customize tiering policy for different data types/segments.

# Appendix

## Appendix A: FlashBlade configuration within Komprise

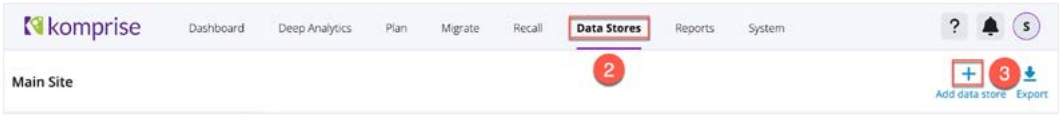
Please follow the Komprise deployment guide to deploy Komprise prior to configuring FlashBlade.

### FlashBlade//S (source) configuration

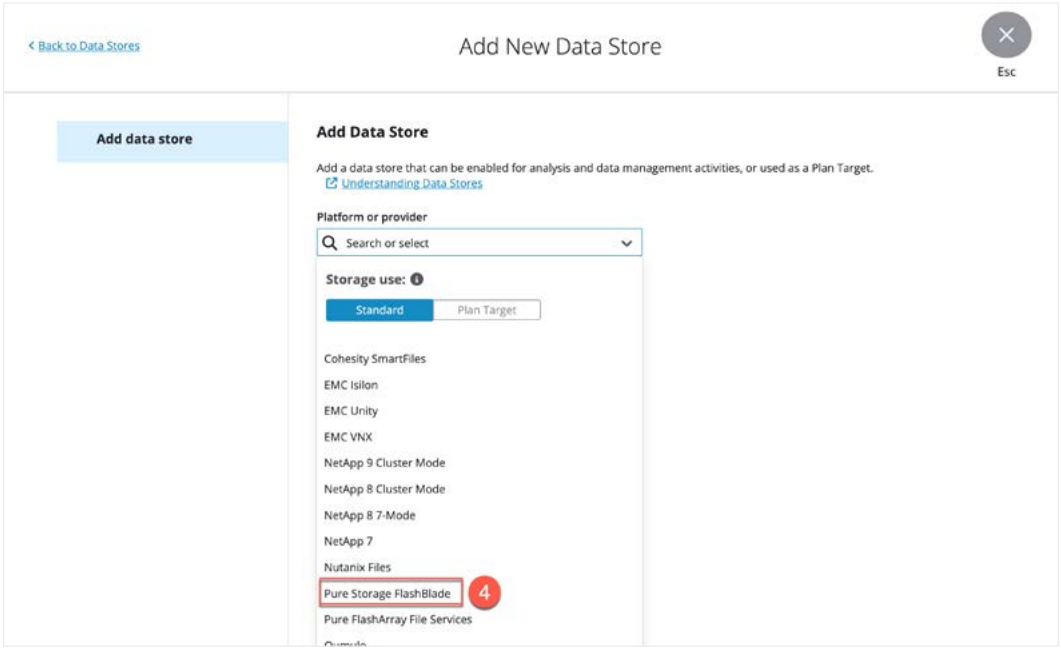
Once Komprise is deployed, the below sequences should be followed to configure FlashBlade//S as the source.

Prior to this, the FlashBlade//S should have the NFS filesystems holding the source data.

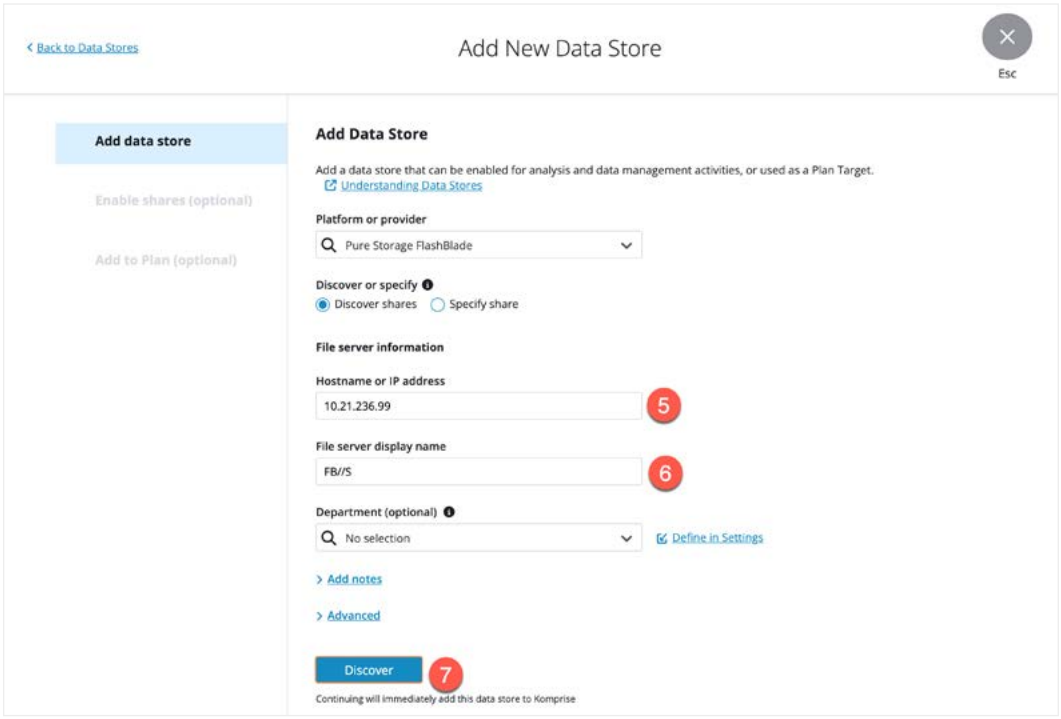
1. Login to the Komprise Director.



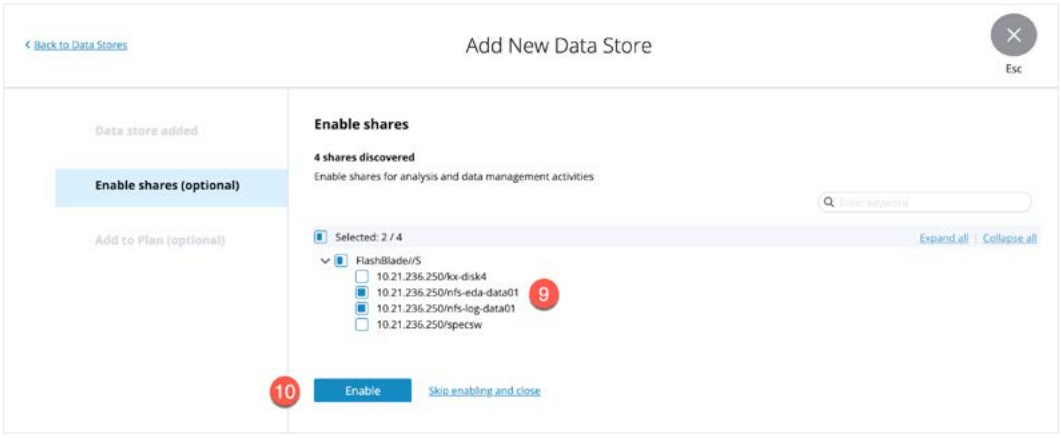

2. Click Data Stores from the top-level menu.
3. Click the “Add data store” option.



4. In the Add New Data Store menu, select “Pure Storage FlashBlade” under the “Standard” storage use.



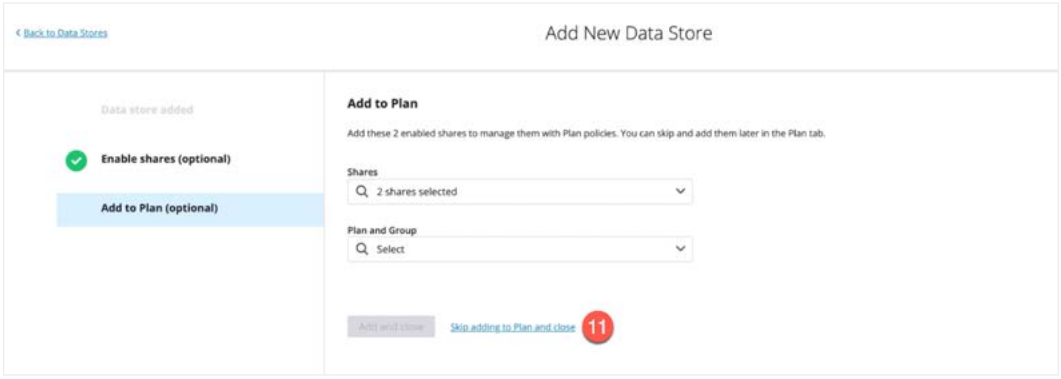

5. Once “Pure Storage FlashBlade” is selected, enter the data VIP of the FlashBlade//S in Hostname or IP address field. Leave the “Discover shares” radio button selected.
6. Provide a meaningful name to identify the File server.
7. Click the Discover button to discover the NFS and SMB shares from the FlashBlade//S.
8. This will discover any SMB shares and if found will display. If the FB//Purity is below 4.1, discovery of SMB shares is not supported. If the FB//Purity is between 4.1 and 4.5.1 and if there were no SMB shares listed, then either there were no SMB shares or over 50 SMB shares in which case the discovery of SMB shares is unsupported. Press the Next button.



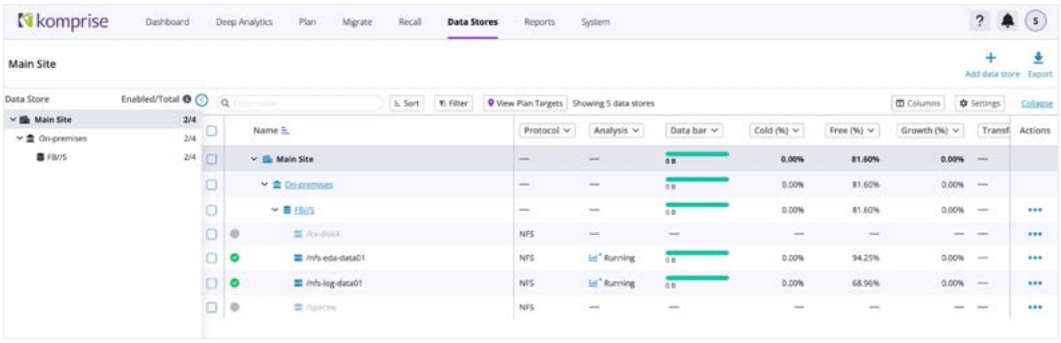
9. The Komprise Director will list all the NFS shares that are discovered. Select the relevant NFS shares that should be included for the analysis and data management activities by Komprise.

10. Click Enable to enable the analysis.

***Note:*** *Steps 9 and 10 are optional. Users can click the "Skip enabling and close" to close this and enable the NFS share later.*




11. The next screen allows the shares to be added to a Komprise Plan. This is optional and can be skipped.

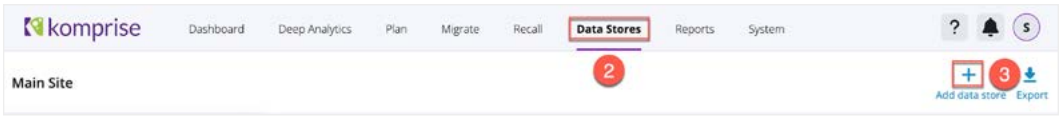



12. The Data Store tab will show the source data store that was configured along with the shares that are enabled for analysis and data management activities.

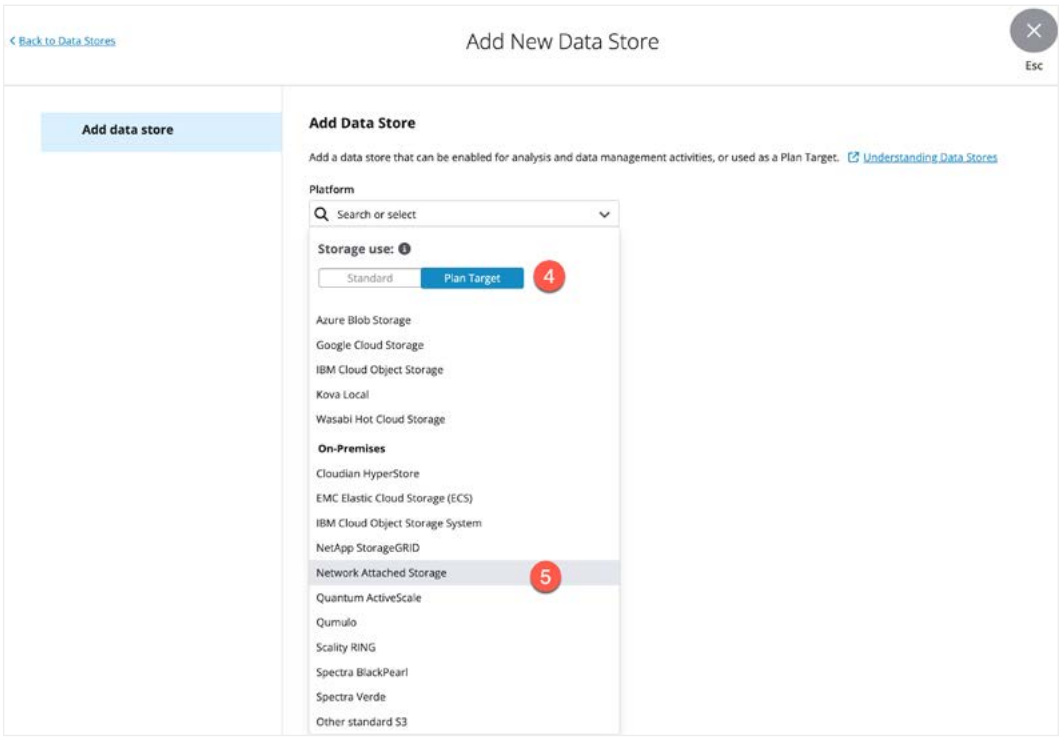
### FlashBlade//E (NFS Target) configuration

Prior to the target configuration, ensure that the NFS share that will be hosting the tiered data on FlashBlade//E is created.

1. Login to the Komprise Director.

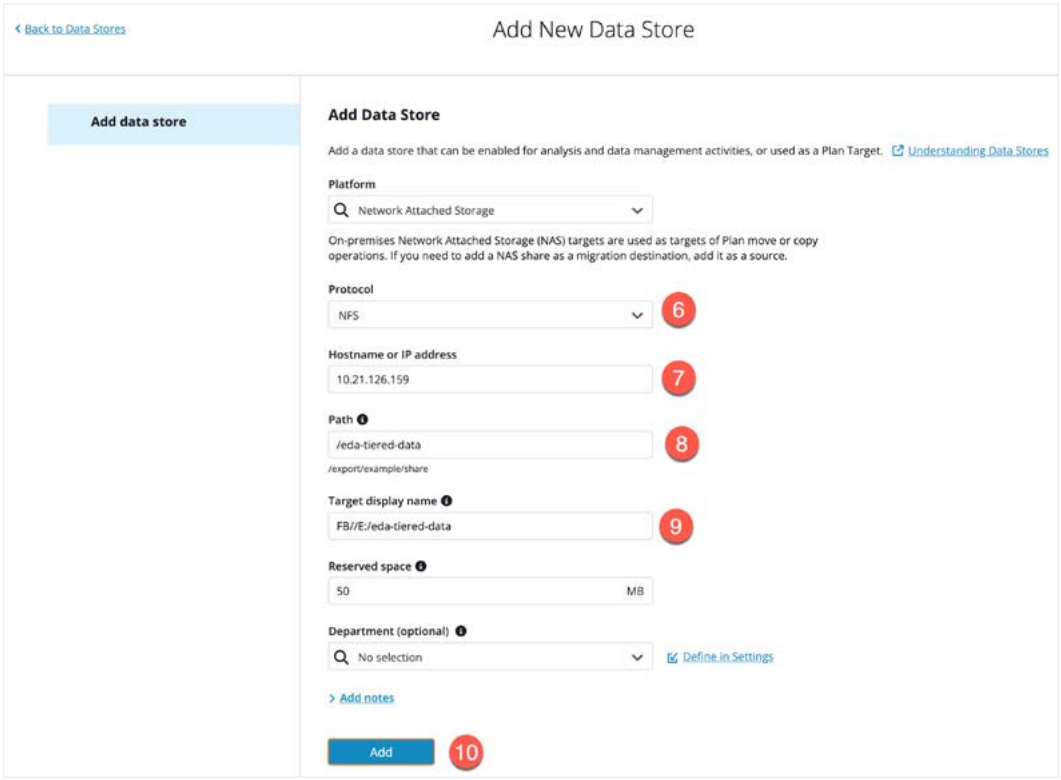



2. Click Data Stores from the top-level menu.
3. Click the “Add data store” option.

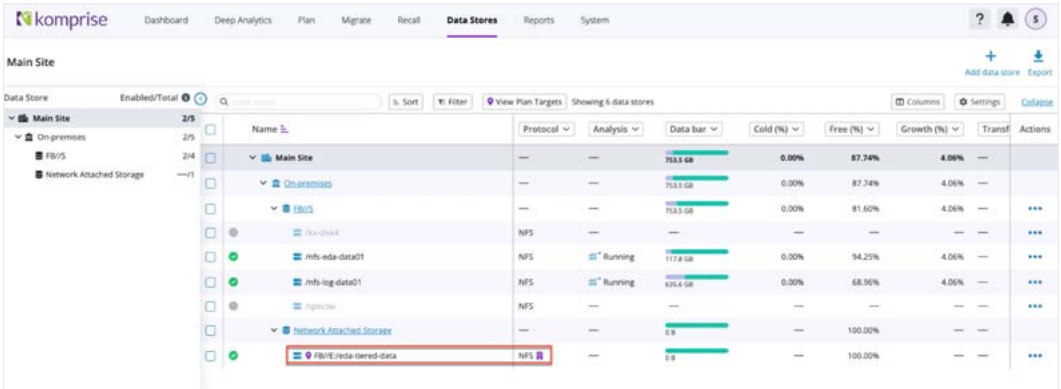



4. In the Add Data Store screen, select the “Plan Target” under Storage use.

5. From the list select “Network Attached Storage”.




6. From the Protocol list, select NFS.
7. Specify the FlashBlade//E data VIP.
8. Specify the NFS share from FlashBlade//E that will hold the tiered data.
9. Enter a meaningful name for the Target display name.
10. Click Add button to the add the target data store.

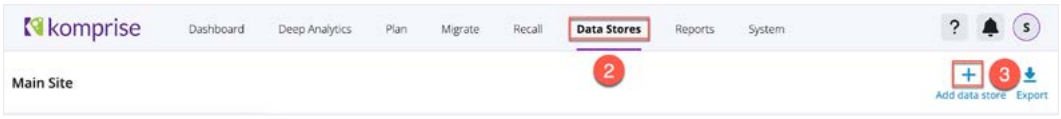



11. The Data store tab will show the target data store that was added, and the protocol should show NFS.

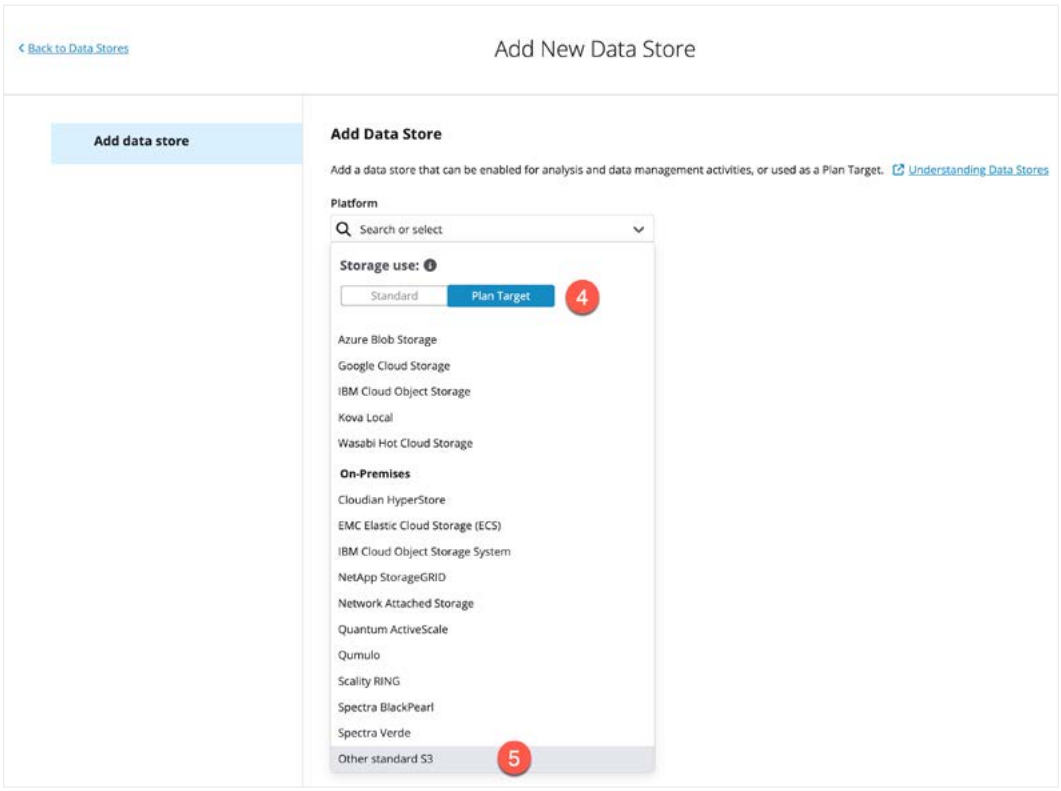
### FlashBlade//E (S3 Target) configuration

Prior to the target configuration, ensure that the S3 bucket that will be hosting the tiered data on FlashBlade//E is created.

1. Login to the Komprise Director.

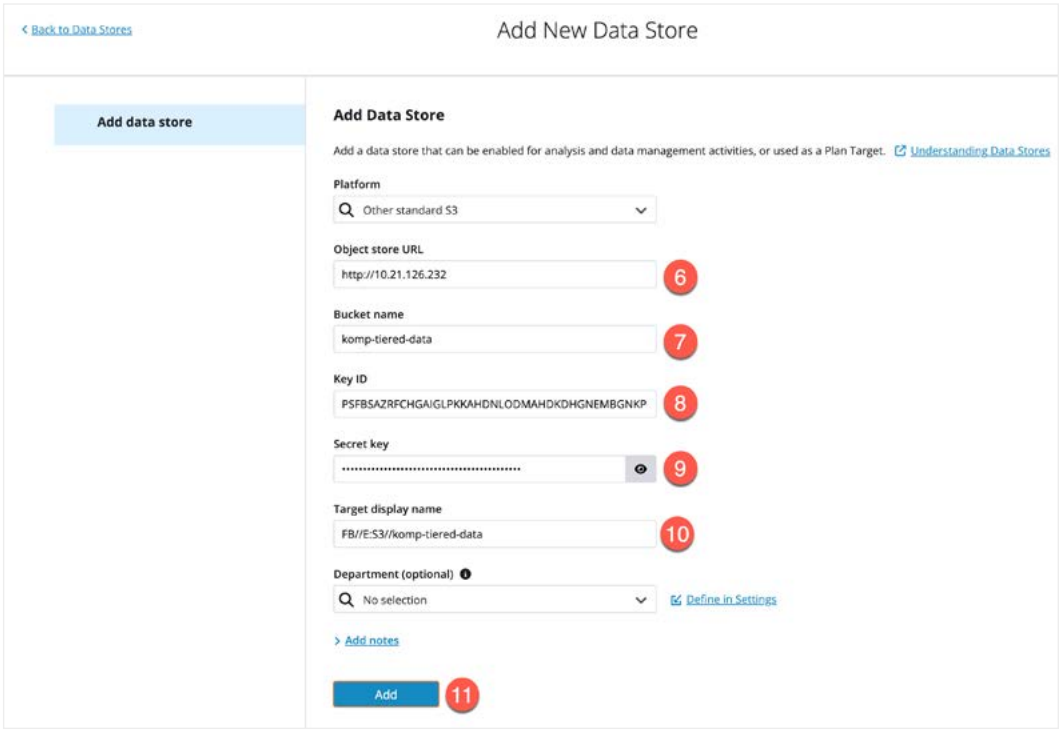



2. Click Data Stores from the top-level menu.
3. Click the “Add data store” option.

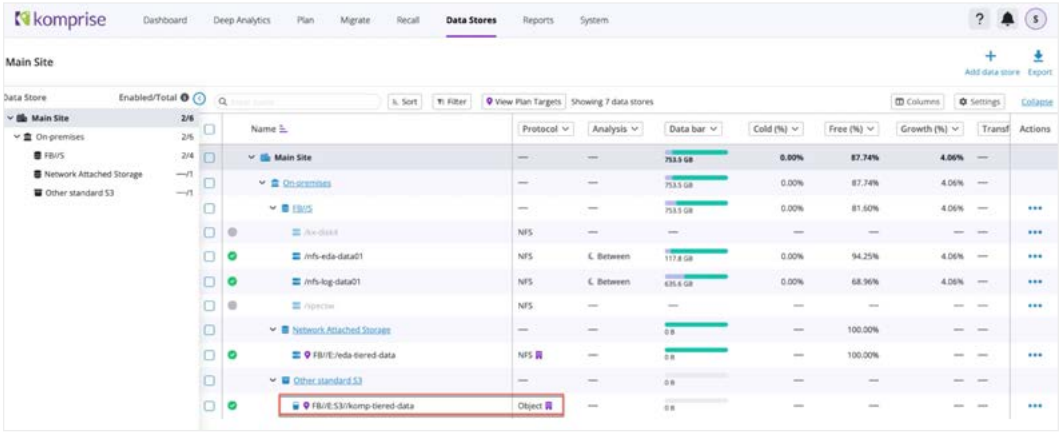



4. In the Add Data Store screen, select the Storage use as “Plan Target”.

5. Select the “Other standard S3” from the list.




6. Specify the FlashBlade//E’s data VIP with http or https under Object store URL.
7. Specify the S3 bucket name that will be hosting the tiered data.
8. Specify the access key for the bucket.
9. Specify the secret key for the bucket.
10. Enter a meaningful name for the target.
11. Click the Add button to add the target.




12. The Data store tab will show the target data store that was added, and the protocol should show as Object.

## Additional Resources

### Supporting Information

- FlashBlade//S and FlashBlade//E
  https://www.purestorage.com/products/unstructured-data-storage.html
- FlashBlade//E: The Dawn of a New Era for Unstructured Data Storage
  https://blog.purestorage.com/products/flashblade-e-new-era-for-unstructured-data-storage/
- Komprise Deployment guide
  https://komprise.freshdesk.com/support/solutions/articles/17000121959-komprise-deployment-guide
- Komprise Admin guide
  https://komprise.freshdesk.com/en/support/solutions/articles/17000048966-introduction-and-table-of-contents
- Komprise Elastic Grid
  https://www.komprise.com/blog/closer-look-komprise-elastic-grid/

## About the Author

Somu Rajarathinam is a Technical Director in the Portfolio Solutions team at Pure Storage. His responsibilities include defining application solutions for Pure Storage products, performing benchmarks, and developing reference architectures for applications running on Pure Storage products. Somu has over 25 years of experience in the database and analytics area. He was a member of the Systems Performance and Oracle Applications Performance Groups at Oracle Corporation. He also worked with Logitech, Inspirage, and Autodesk, where his assignments include delivering database and performance solutions, managing infrastructure, and providing support to database and analytical applications both on-prem and on the Cloud.

### Intellectual Property Notice



1 https://www.komprise.com/resource/the-2023-komprise-unstructured-data-management-report/
2 https://www.statista.com/statistics/871513/worldwide-data-created/
3 https://www.purestorage.com/content/dam/pdf/en/white-papers/protected/wp-all-flash-unstructured-data-repository.pdf





PS2536-01-en 11/23