NetApp and Dremio's Next Generation Hybrid Iceberg Lakehouse Solution

NetApp Solutions

NetApp
September 19, 2024

Table of Contents

NetApp and Dremio’s Next Generation Hybrid Iceberg Lakehouse Solution . . . 1
- Introduction . . . 1
- Solution Overview . . . 1
- Technology Requirements . . . 2
- Deployment Procedure . . . 3
- Solution verification overview . . . 14
- Customer Use Cases . . . 21
- Conclusion . . . 23

NetApp and Dremio’s Next Generation Hybrid Iceberg Lakehouse Solution

Introduction

Karthikeyan Nagalingam, NetApp.
Roger Frey, Dremio.
Mark Shainman, Dremio.

In this document, we cover the deployment details of Dremio with different source of data from NetApp storage controllers such as ONTAP S3 and NAS as well as storageGRID. In the deployment, we used TPC-DS benchmarking tool to run 99 SQL queries on top of various sources. The document further delves into customer use cases within NetApp as well as auto parts sales customer use-case.

Solution Overview

The Hybrid Iceberg Lakehouse solution provides unique benefits to address customer challenges faced by data lake customers. By Leveraging Dremio Unified Lakehouse platform and NetApp ONTAP, StorageGRID, NetApp Cloud solutions, companies can add significant value to their business operations. The solution not only provides access to multiple data sources, including NetApp sources to but also enhance overall analytical performance and helps companies drive business insight that leads to business growth.

NetApp Overview

- NetApp’s offerings, such as ONTAP and StorageGRID, enable the separation of storage and compute, enabling optimal resource utilization based on specific requirements. This flexibility empowers customers to independently scale their storage using NetApp storage solutions.
- By leveraging NetApp’s storage controllers, customers can efficiently serve data to their vector database using NFS and S3 protocols. These protocols facilitate customer data storage and manage the vector database index, eliminating the need for multiple copies of data accessed through file and object methods.
- NetApp ONTAP provides native support for NAS and Object storage across leading cloud service providers like AWS, Azure, and Google Cloud. This wide compatibility ensures seamless integration, enabling customer data mobility, global accessibility, disaster recovery, dynamic scalability, and high performance.

Dremio overview

Dremio is the Unified Lakehouse Platform for self-service analytics and AI. Dremio’s Unified Analytics Platform brings users closer to the data with lakehouse flexibility, scalability, and performance at a fraction of the cost. Dremio enables shift-left analytics to eliminate complex and costly data integration and ETL, delivering seamless enterprise-scale analytics with no data movement.

Easy-to-use self-service analytics enabled through a universal semantic layer and a tightly integrated, highly performant SQL query engine makes it easier to connect, govern, and analyze all data, both in the cloud and on-premises.

Dremio’s Apache Iceberg-native lakehouse management capabilities simplify data discovery, and automate

data optimization, delivering high-performance analytics with Git-inspired data versioning.

Foundationally built on open source and open standards, Dremio lets companies avoid lock-in and remain positioned for innovation. Enterprise companies trust Dremio as the easiest-to-use lakehouse platform with the best price-performance on all workloads

What value does the Dremio and NetApp Hybrid Iceberg Lakehouse solution deliver to customers?

- **Improved Data Management and Accessibility**: Dremio is well-known for its data lakehouse platform that allows organizations to query data directly from their data lakes at high speed. NetApp, on the other hand, is a leading provider of cloud data services and data storage solutions. The joint offer provides companies a comprehensive solution for storing, managing, accessing and analyzing their enterprise's data efficiently.
- **Performance Optimization**: With NetApp's expertise in data storage and Dremio's capabilities in data processing and data optimization, the partnership offers a solution that improves the performance of data operations, reduces latency and increasing speed to business insight. Dremio has even delivered performance benefits to NetApp's own internal IT analytical infrastructure .
- **Scalability**: Both Dremio and NetApp offer a solution that is designed to scale. The joint solution provides customers with highly scalable data storage, data management and analytical environment. In a Hybrid Iceberg Lakehouse environment, the Dremio SQL query engine paired with NetApp StorageGRID delivers unparalleled scalability, concurrency and query performance, capable of handling the analytical needs of any business.
- **Data Security and Governance**: Both companies have a strong focus on data security and governance. Together, they offer robust security and data governance features, ensuring that data is protected and that data governance requirements are met. Features such as role-based and fine-grain access controls, comprehensive auditing, end-to-end data lineage, unified identity management and SSO with an extensive compliance and security framework ensures companies analytical data environments are secure and governed.
- **Cost Efficiency**: By integrating Dremio's data lake engine with NetApp's storage solutions, customers are able to reduce costs associated with data management and data movement. Organizations are also able to move from legacy data lake environments to a more modern lakehouse solution composed of NetApp and Dremio. This Hybrid Iceberg Lakehouse solution delivers high-speed query performance and market-leading query concurrency that lowers TCO and decreases time to business insight.

Technology Requirements

The hardware and software configurations outlined below were utilized for validations performed in this document. These configurations serve as a guideline to help you set up your environment, However, please note the specific components may vary depending on individual customer requirements.

Hardware requirements

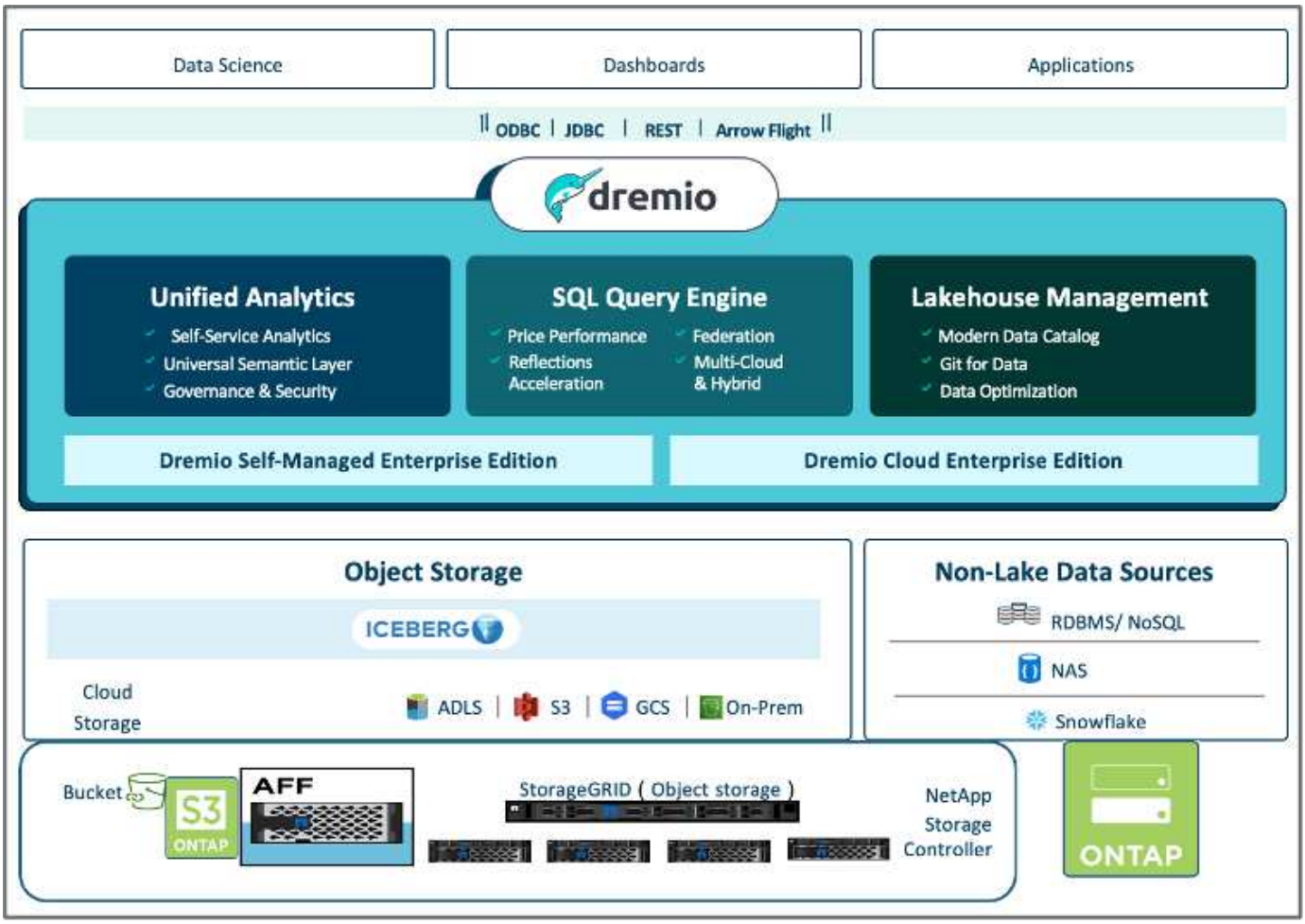
Hardware	Details
NetApp AFF Storage array HA Pair	• A800 • ONTAP 9.14.1 • 48 x 3.49TB SSD-NVM • Two S3 Buckets: Dremio metadata and customer data.
4 x FUJITSU PRIMERGY RX2540 M4	• 64 CPUs • Intel® Xeon® Gold 6142 CPU @ 2.60GHz • 256 GM Physical Memory • 1 x 100GbE network port
Networking	• 100 GbE
StorageGRID	* 1 x SG100, 3xSGF6024 * 3 x 24 x 7.68TB * Two S3 Buckets: Dremio metadata and customer data.

Software requirements

Software	Details
Dremio	• version - 25.0.3-202405170357270647-d2042e1b • Enterprise Edition
On-Prem	• 5 node Dremio cluster • 1 Master coordinator and 4 executors

Deployment Procedure

In this reference architecture validation, we used a Dremio configuration composed of one coordinator and four executors



NetApp setup

- Storage system initialization
- Storage virtual machine (SVM) creation
- Assignment of logical network interfaces
- NFS, S3 configuration and licensing

Please follow the steps below for NFS (Network File System):

1. Create a Flex Group volume for NFSv4 or NFSv3. In our set up for this validation, we have used 48 SSDs, 1 SSD dedicated for the controller's root volume and 47 SSDs spread across for NFSv4]].Verify that the NFS export policy for the Flex Group volume has read/write permissions for the Dremio servers network.
2. On all Dremio servers, create a folder and mount the Flex Group volume onto this folder through a Logical Interface (LIF) on each Dremio servers.

Please follow the steps below for S3 (Simple Storage Service):

1. Set up an object-store-server with HTTP enabled and the admin status set to 'up' using the "vserver object-store-server create" command. You have the option to enable HTTPS and set a custom listener port.
2. Create an object-store-server user using the "vserver object-store-server user create -user <username>" command.
3. To obtain the access key and secret key, you can run the following command: "set diag; vserver object-store-server user show -user <username>". However, moving forward, these keys will be supplied during

the user creation process or can be retrieved using REST API calls.

4. Establish an object-store-server group using the user created in step 2 and grant access. In this example, we have provided "FullAccess".
5. Create a two S3 buckets by setting its type to "S3". One for Dremio configuration and one for customer data.

Zookeeper setup

You can use Dremio provided zookeeper configuration. In this validation, we used separate zookeeper. we followed the steps mentioned in this weblink https://medium.com/@ahmetfurkandemir/distributed-hadoop-cluster-1-spark-with-all-dependincies-03c8ec616166

Dremio setup

We followed this weblink to install Dremio via tar ball.

1. Create a Dremio group.

```
sudo groupadd -r dremio
```

2. Create a dremio user.

```
sudo useradd -r -g dremio -d /var/lib/dremio -s /sbin/nologin dremio
```

3. Create Dremio directories.

```
sudo mkdir /opt/dremio
sudo mkdir /var/run/dremio && sudo chown dremio:dremio /var/run/dremio
sudo mkdir /var/log/dremio && sudo chown dremio:dremio /var/log/dremio
sudo mkdir /var/lib/dremio && sudo chown dremio:dremio /var/lib/dremio
```

4. Download the tar file from https://download.dremio.com/community-server/
5. Unpack Dremio into the /opt/dremio directory.

```
sudo tar xvf dremio-enterprise-25.0.3-202405170357270647-d2042e1b.tar.gz
-C /opt/dremio --strip-components=1
```

6. Create a symbolic link for the configuration folder.

```
sudo ln -s /opt/dremio/conf /etc/dremio
```

7. Set up your service configuration (SystemD setup).
    1. Copy the unit file for the dremio daemon from /opt/dremio/share/dremio.service to

/etc/systemd/system/dremio.service.

2. Restart system

```
sudo systemctl daemon-reload
```

3. Enable dremio to start at boot.

```
sudo systemctl enable dremio
```

8. Configure Dremio on coordinator. See Dremio Configuration for more information

1. Dremio.conf

```
root@hadoopmaster:/usr/src/tpcds# cat /opt/dremio/conf/dremio.conf

paths: {
  # the local path for dremio to store data.
  local: ${DREMIO_HOME}"/dremiocache"

  # the distributed path Dremio data including job results,
downloads, uploads, etc
  #dist: "hdfs://hadoopmaster:9000/dremiocache"
  dist: "dremioS3:///dremioconf"
}

services: {
  coordinator.enabled: true,
  coordinator.master.enabled: true,
  executor.enabled: false,
  flight.use_session_service: false
}

zookeeper: "10.63.150.130:2181,10.63.150.153:2181,10.63.150.151:2181"
services.coordinator.master.embedded-zookeeper.enabled: false
root@hadoopmaster:/usr/src/tpcds#
```

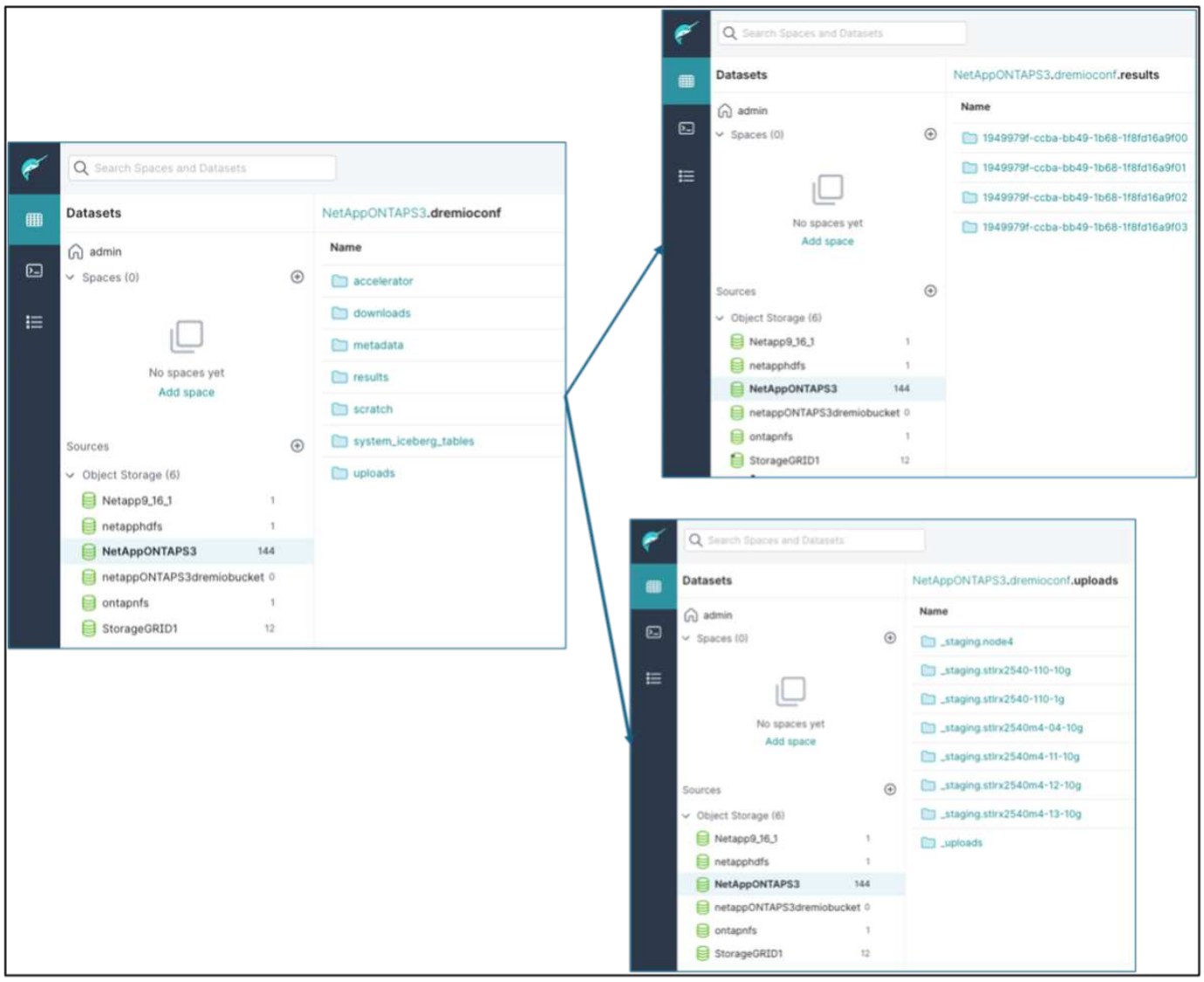
2. Core-site.xml

```
root@hadoopmaster:/usr/src/tpcds# cat /opt/dremio/conf/core-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
```

```
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
    <property>
        <name>fs.dremioS3.impl</name>
        <value>com.dremio.plugins.s3.store.S3FileSystem</value>
    </property>
    <property>
                <name>fs.s3a.access.key</name>
                <value>24G4C1316APP2BIPDE5S</value>
    </property>
    <property>
                <name>fs.s3a.endpoint</name>
                <value>10.63.150.69:80</value>
        </property>
    <property>
            <name>fs.s3a.secret.key</name>
            <value>Zd28p43rgZaU44PX_ftT279z9nt4jBSro97j87Bx</value>
    </property>
    <property>
            <name>fs.s3a.aws.credentials.provider</name>
            <description>The credential provider type.</description>

<value>org.apache.hadoop.fs.s3a.SimpleAWSCredentialsProvider</value>
    </property>
    <property>
                <name>fs.s3a.path.style.access</name>
                <value>false</value>
        </property>
    <property>
            <name>hadoop.proxyuser.dremio.hosts</name>
            <value>*</value>
    </property>
    <property>
```

```
            <name>hadoop.proxyuser.dremio.groups</name>
            <value>*</value>
    </property>
    <property>
            <name>hadoop.proxyuser.dremio.users</name>
            <value>*</value>
    </property>
    <property>
        <name>dremio.s3.compat</name>
        <description>Value has to be set to true.</description>
        <value>true</value>
    </property>
    <property>
        <name>fs.s3a.connection.ssl.enabled</name>
        <description>Value can either be true or false, set to true
to use SSL with a secure Minio server.</description>
        <value>false</value>
    </property>
</configuration>
root@hadoopmaster:/usr/src/tpcds#
```

9. The Dremio configuration are stored in netapp object storage. In our validation, the “dremioconf” bucket resides in ontap s3 bucket. The below picture shows some details from “scratch” and “uploads” folder of the “dremioconf” s3 bucket.



1. Configure Dremio on executors. In our setup, we have 3 executors.
    1. dremio.conf

```
paths: {
  # the local path for dremio to store data.
  local: ${DREMIO_HOME}"/dremiocache"

  # the distributed path Dremio data including job results,
downloads, uploads, etc
  #dist: "hdfs://hadoopmaster:9000/dremiocache"
  dist: "dremioS3:///dremioconf"
}

services: {
  coordinator.enabled: false,
  coordinator.master.enabled: false,
  executor.enabled: true,
  flight.use_session_service: true
}

zookeeper: "10.63.150.130:2181,10.63.150.153:2181,10.63.150.151:2181"
services.coordinator.master.embedded-zookeeper.enabled: false
```

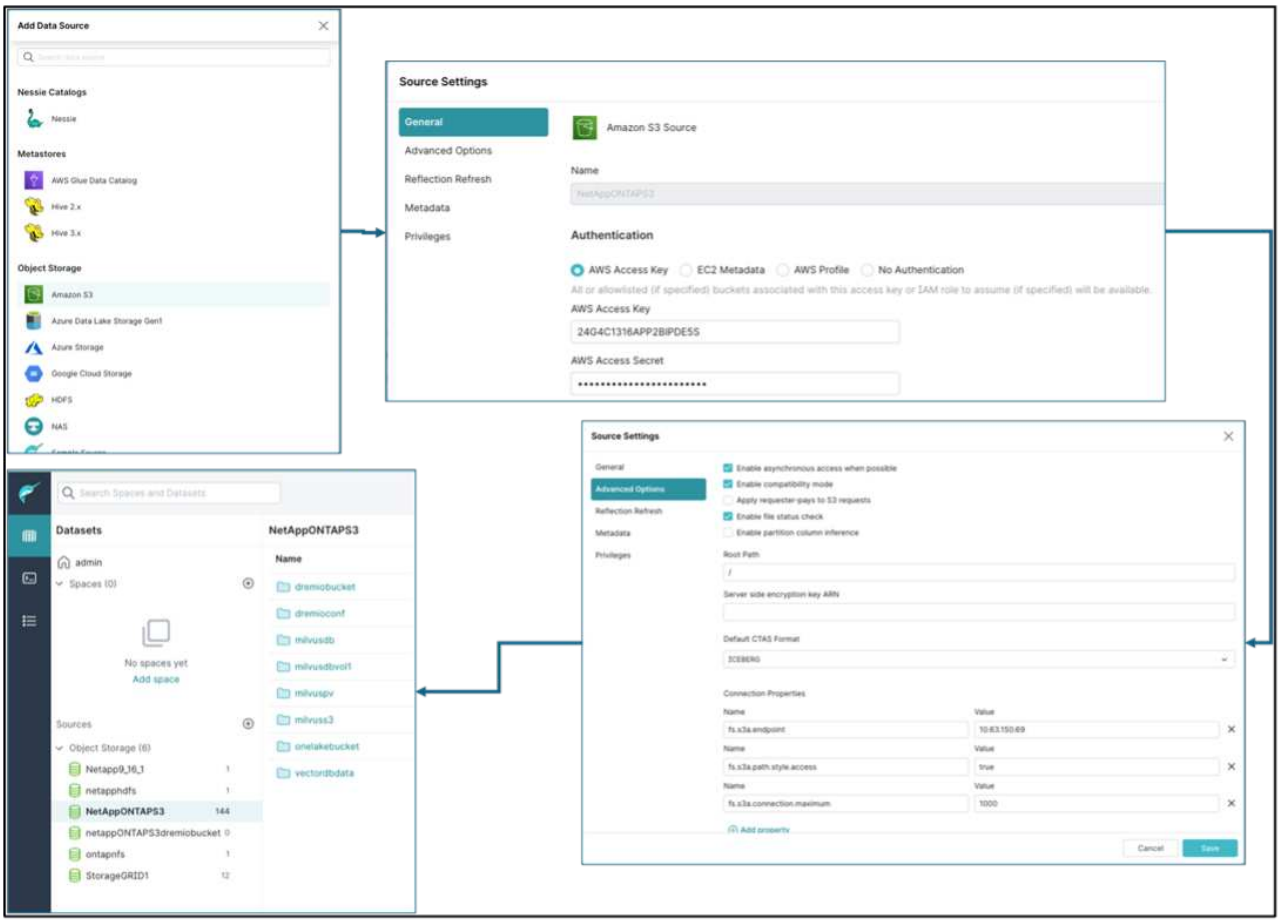
2. Core-site.xml – same as coordinator configuration.

Multiple sources setup

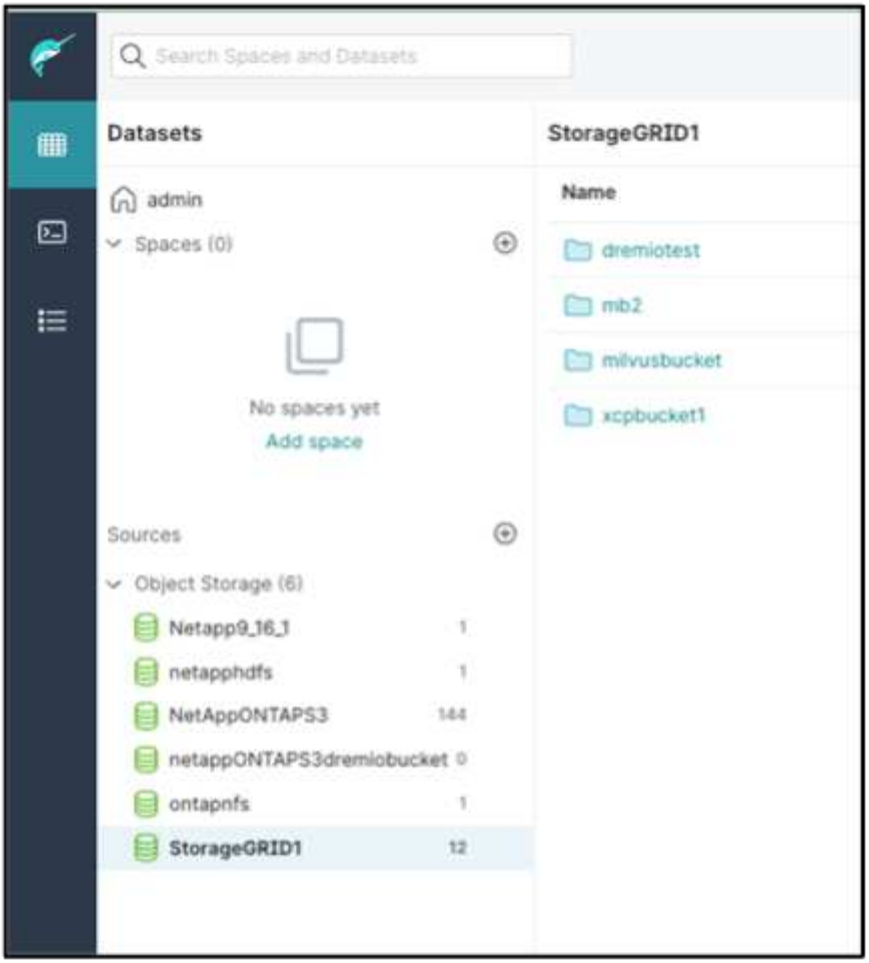
1. Configure ONTAP S3 and storageGRID as a s3 source in Dremio.
    1. Dremio dashboard → datasets → sources → add source.
    2. In general section, please update AWS access and secret key
    3. In advanced option, enable compatibility mode, update connection properties with the below details. The endpoint IP/Name from NetApp storage controller either from ontap s3 or storageGRID.

```
fs.s3a.endoint = 10.63.150.69
fs.s3a.path.style.access = true
fs.s3a.connection.maximum=1000
```

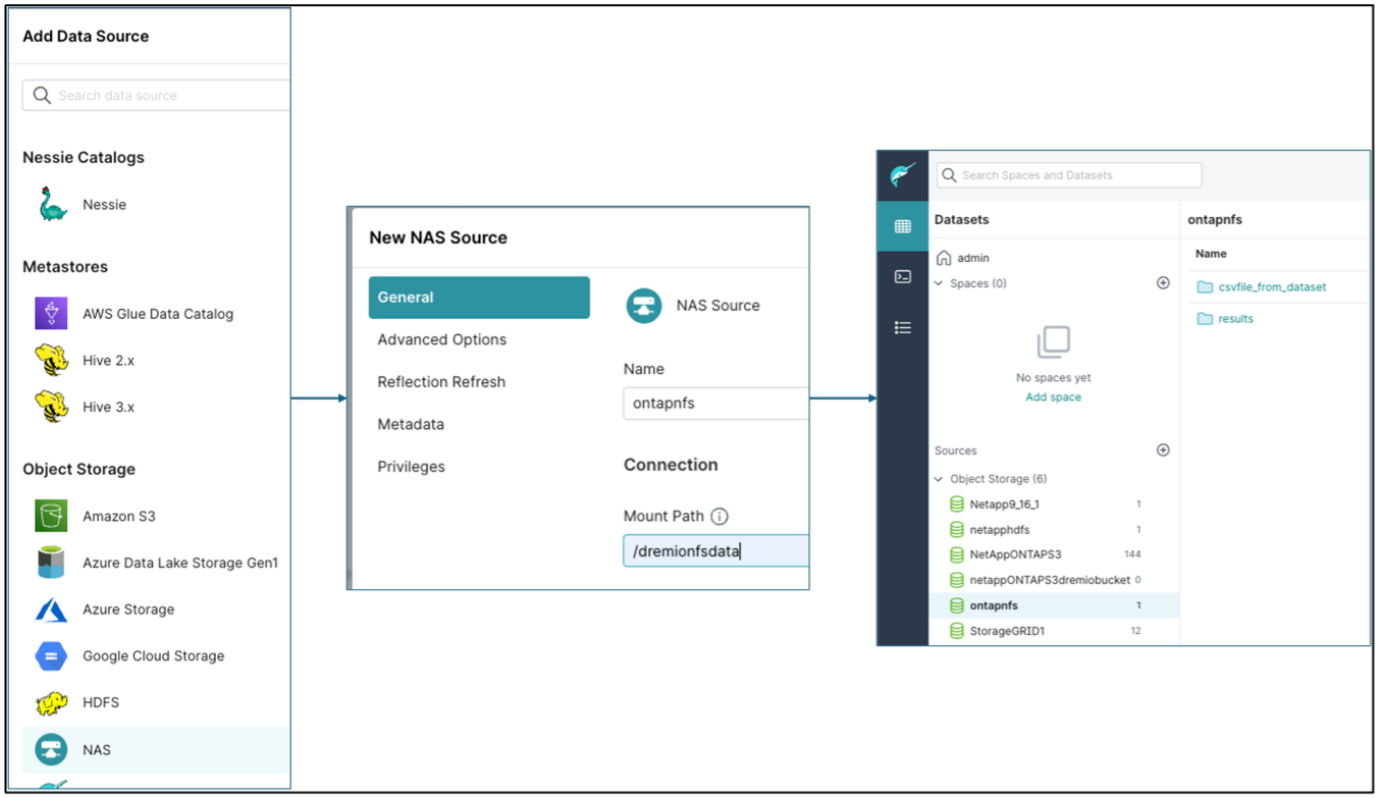
    4. Enable local caching when possible, Max Percent of total available cache to use when possible = 100
    5. Then view the list of buckets from NetApp object storage.



6. Sample view of storageGRID bucket details



2. Configure NAS (specifically NFS) as a source in Dremio.
    1. Dremio dashboard → datasets → sources → add source.
    2. In general section, enter the name and NFS mount path. Please make sure the NFS mount path is mounted on the same folder on all the nodes in the Dremio cluster.


Add Data Source
Search data source
Nessie Catalogs
Nessie
Metastores
AWS Glue Data Catalog
Hive 2.x
Hive 3.x
Object Storage
Amazon S3
Azure Data Lake Storage Gen1
Azure Storage
Google Cloud Storage
HDFS
NAS
New NAS Source
General
Advanced Options
Reflection Refresh
Metadata
Privileges
NAS Source
Name
ontapnfs
Connection
Mount Path
/dremionfsdata
Search Spaces and Datasets
Datasets
admin
Spaces (0)
No spaces yet
Add space
Sources
Object Storage (6)
Netapp9_16_1
netapphdfs
NetAppONTAPS3
netappONTAPS3dremiobucket
ontapnfs
StorageGRID1
ontapnfs
Name
csvfile_from_dataset
results

+

```
root@hadoopmaster:~# for i in hadoopmaster hadoopnode1 hadoopnode2
hadoopnode3 hadoopnode4; do ssh $i "date;hostname;du -hs
/opt/dremio/data/spill/ ; df -h //dremionfsdata "; done
Fri Sep 13 04:13:19 PM UTC 2024
hadoopmaster
du: cannot access '/opt/dremio/data/spill/': No such file or directory
Filesystem                   Size  Used Avail Use% Mounted on
10.63.150.69:/dremionfsdata  2.1T  921M  2.0T   1% /dremionfsdata
Fri Sep 13 04:13:19 PM UTC 2024
hadoopnode1
12K /opt/dremio/data/spill/
Filesystem                   Size  Used Avail Use% Mounted on
10.63.150.69:/dremionfsdata  2.1T  921M  2.0T   1% /dremionfsdata
Fri Sep 13 04:13:19 PM UTC 2024
hadoopnode2
12K /opt/dremio/data/spill/
Filesystem                   Size  Used Avail Use% Mounted on
10.63.150.69:/dremionfsdata  2.1T  921M  2.0T   1% /dremionfsdata
Fri Sep 13 16:13:20 UTC 2024
hadoopnode3
16K /opt/dremio/data/spill/
Filesystem                   Size  Used Avail Use% Mounted on
10.63.150.69:/dremionfsdata  2.1T  921M  2.0T   1% /dremionfsdata
Fri Sep 13 04:13:21 PM UTC 2024
node4
12K /opt/dremio/data/spill/
Filesystem                   Size  Used Avail Use% Mounted on
10.63.150.69:/dremionfsdata  2.1T  921M  2.0T   1% /dremionfsdata
root@hadoopmaster:~#
```

Solution verification overview

In this section, we have executed SQL test queries from multiple sources to verify the functionality, test and verify the spillover to NetApp storage.

SQL query on Object storage

1. Set the memory to 250GB per server in dremio.env

```
root@hadoopmaster:~# for i in hadoopmaster hadoopnode1 hadoopnode2
hadoopnode3 hadoopnode4; do ssh $i "hostname; grep -i
DREMIO_MAX_MEMORY_SIZE_MB /opt/dremio/conf/dremio-env; cat /proc/meminfo
| grep -i memtotal"; done
hadoopmaster
#DREMIO_MAX_MEMORY_SIZE_MB=120000
DREMIO_MAX_MEMORY_SIZE_MB=250000
MemTotal:       263515760 kB
hadoopnode1
#DREMIO_MAX_MEMORY_SIZE_MB=120000
DREMIO_MAX_MEMORY_SIZE_MB=250000
MemTotal:       263515860 kB
hadoopnode2
#DREMIO_MAX_MEMORY_SIZE_MB=120000
DREMIO_MAX_MEMORY_SIZE_MB=250000
MemTotal:       263515864 kB
hadoopnode3
#DREMIO_MAX_MEMORY_SIZE_MB=120000
DREMIO_MAX_MEMORY_SIZE_MB=250000
MemTotal:       264004556 kB
node4
#DREMIO_MAX_MEMORY_SIZE_MB=120000
DREMIO_MAX_MEMORY_SIZE_MB=250000
MemTotal:       263515484 kB
root@hadoopmaster:~#
```

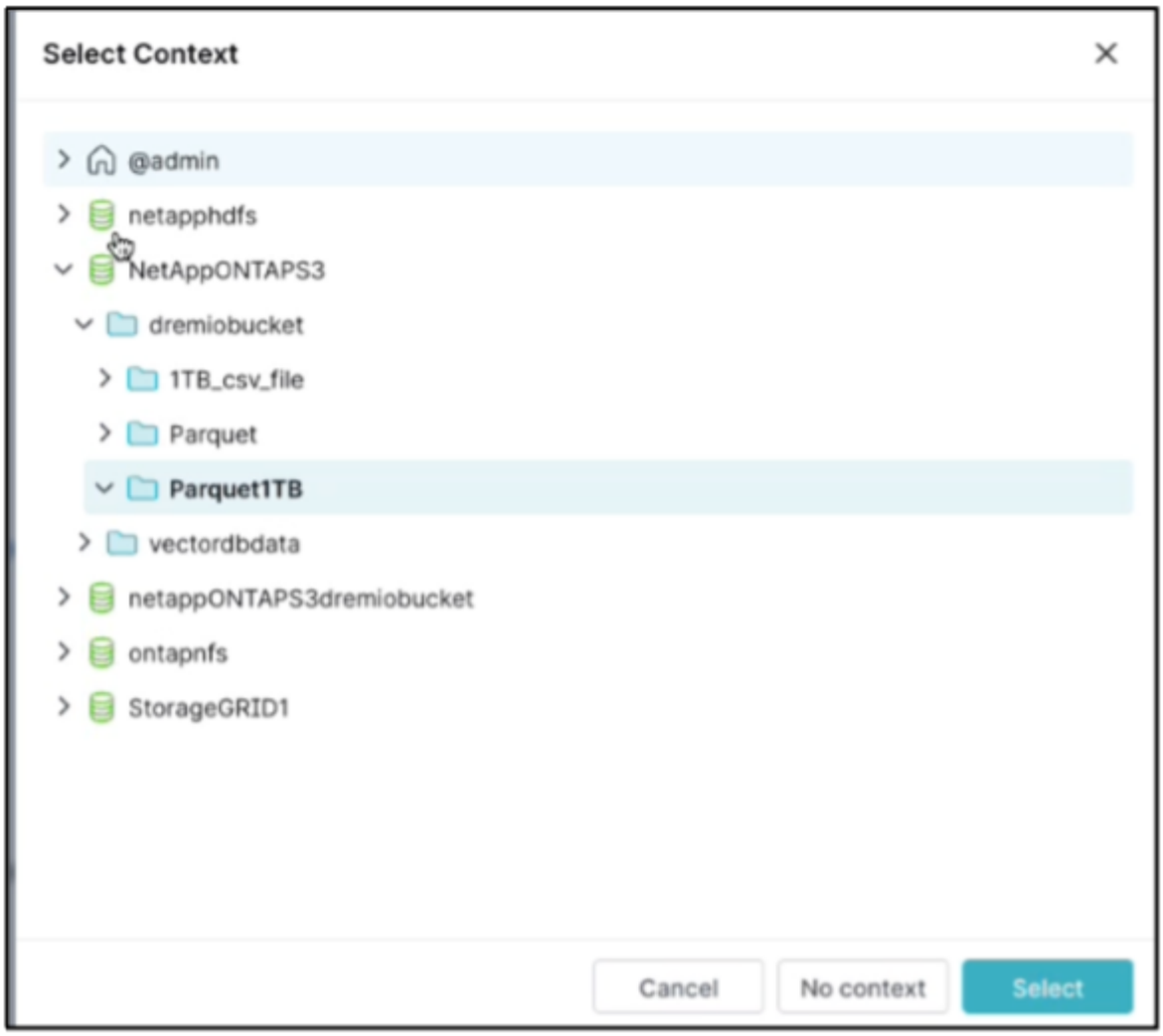
2. Check the spill over location (${DREMIO_HOME}"/dremiocache) in dremio.conf file and storage details.

```
paths: {
  # the local path for dremio to store data.
  local: ${DREMIO_HOME}"/dremiocache"

  # the distributed path Dremio data including job results, downloads,
uploads, etc
  #dist: "hdfs://hadoopmaster:9000/dremiocache"
  dist: "dremioS3:///dremioconf"
}

services: {
  coordinator.enabled: true,
  coordinator.master.enabled: true,
  executor.enabled: false,
  flight.use_session_service: false
}

zookeeper: "10.63.150.130:2181,10.63.150.153:2181,10.63.150.151:2181"
services.coordinator.master.embedded-zookeeper.enabled: false
```

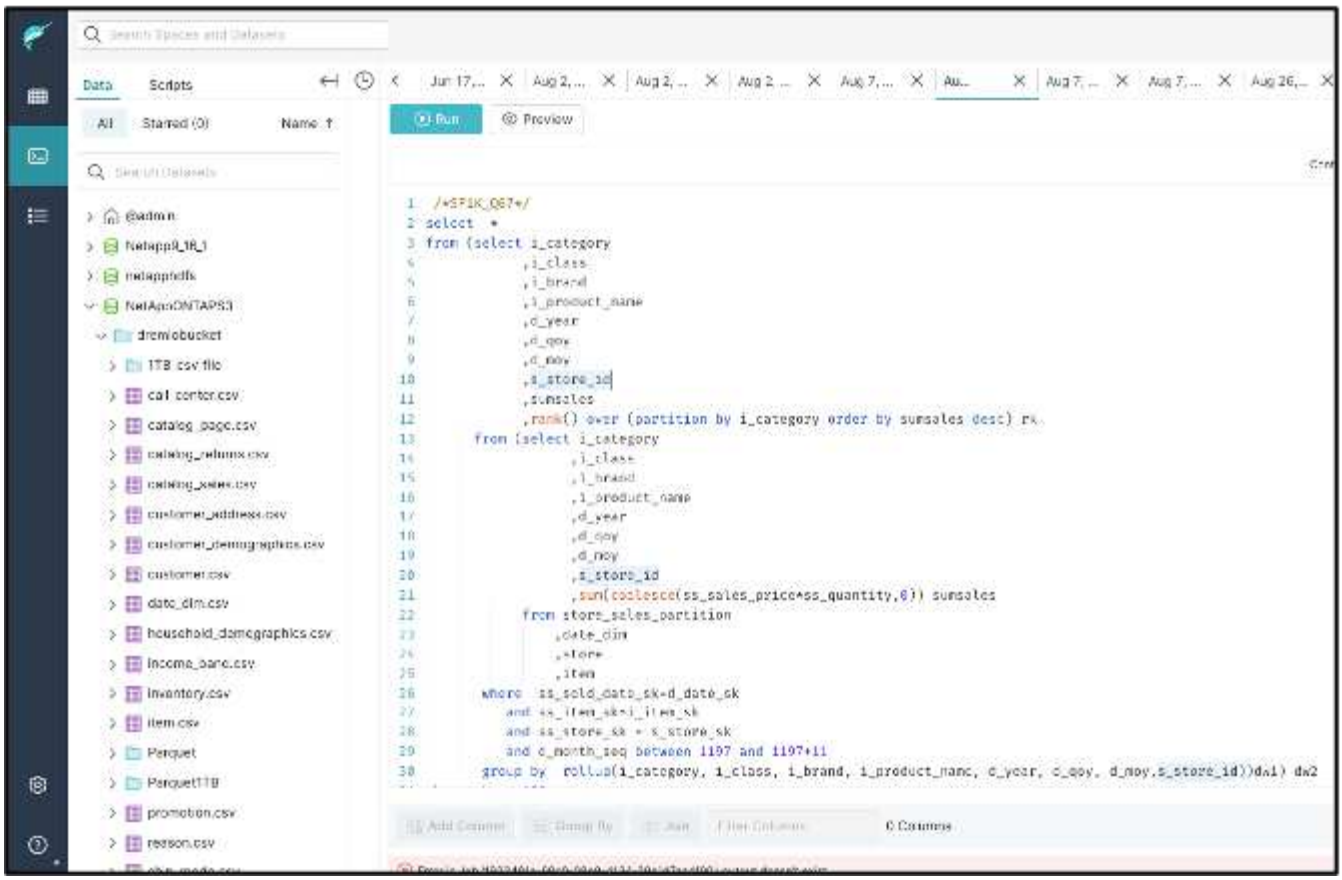
3. Dremio spill over location to NetApp NFS storage

```
root@hadoopnode1:~# ls -ltrh /dremiocache
total 4.0K
drwx------ 3 nobody nogroup 4.0K Sep 13 16:00 spilling_stlrx2540m4-12-
10g_45678
root@hadoopnode1:~# ls -ltrh /opt/dremio/dremiocache/
total 8.0K
drwxr-xr-x 3 dremio dremio 4.0K Aug 22 18:19 spill_old
drwxr-xr-x 4 dremio dremio 4.0K Aug 22 18:19 cm
lrwxrwxrwx 1 root   root     12 Aug 22 19:03 spill -> /dremiocache
root@hadoopnode1:~# ls -ltrh /dremiocache
total 4.0K
drwx------ 3 nobody nogroup 4.0K Sep 13 16:00 spilling_stlrx2540m4-12-
10g_45678
root@hadoopnode1:~# df -h /dremiocache
Filesystem                              Size  Used Avail Use% Mounted on
10.63.150.159:/dremiocache_hadoopnode1  2.1T  209M  2.0T   1%
/dremiocache
root@hadoopnode1:~#
```

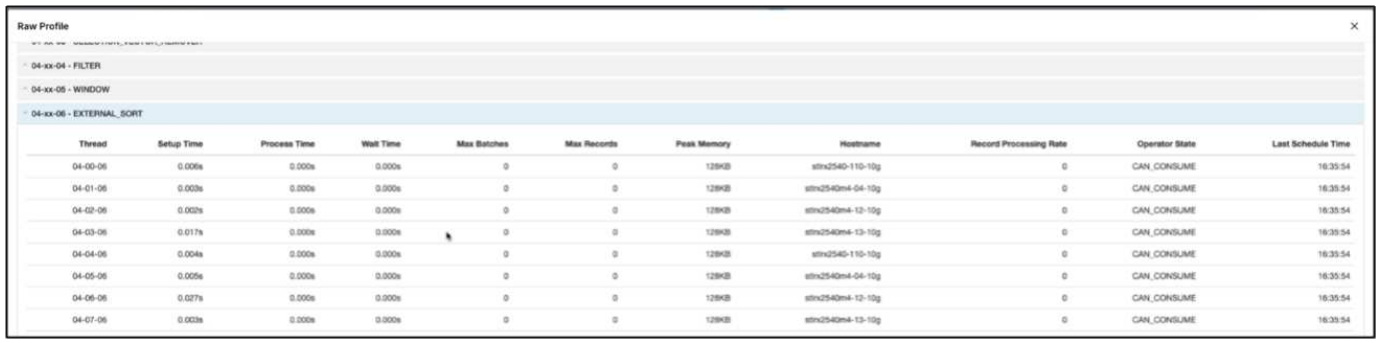
4. Select the context. In our test, we ran the test against TPCDS generated parquet files resides in ONTAP S3. Dremio Dashboard → SQL runner → context → NetAppONTAPS3→Parquet1TB



1. Run the TPC-DS query67 from Dremio dashboard



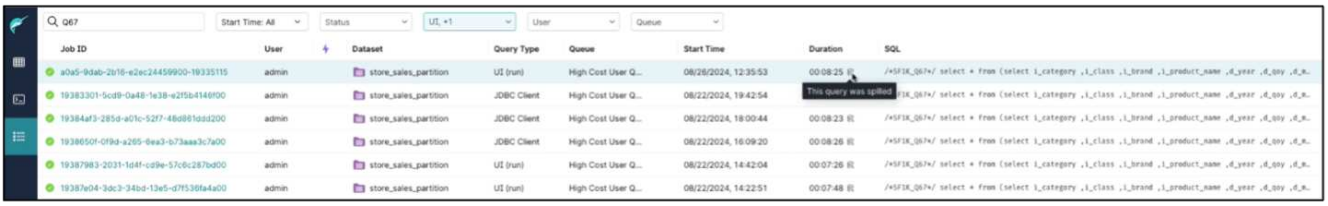
1. Check the job is running on all executor. Dremio dashboard → jobs → <jobid> → raw profile → select EXTERNAL_SORT → Hostname



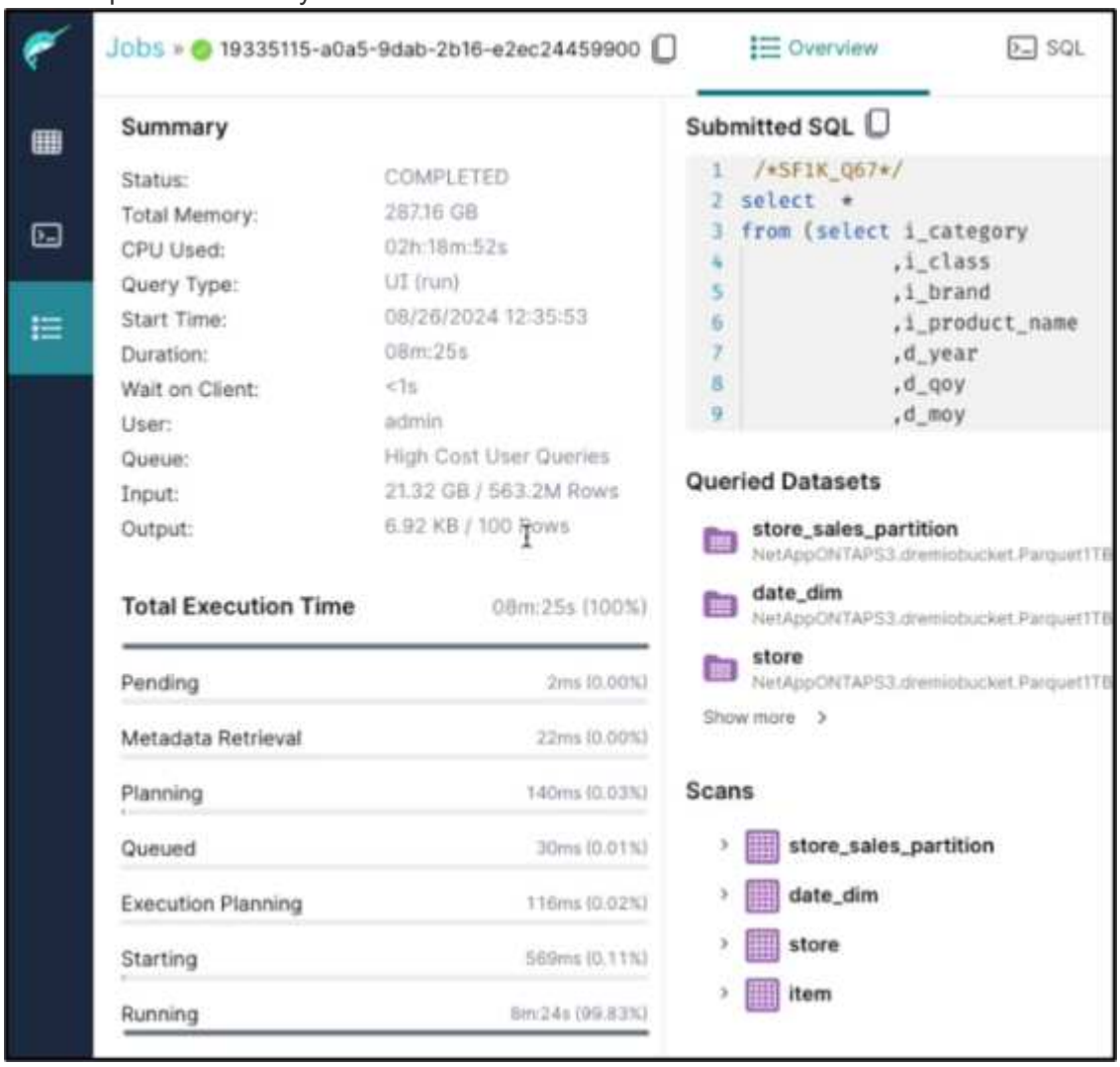
1. when the SQL query running, you can check the split folder for data caching in NetApp storage controller.

```
root@hadoopnode1:~# ls -ltrh /dremiocache
total 4.0K
drwx------ 3 nobody nogroup 4.0K Sep 13 16:00 spilling_stlrx2540m4-12-
10g_45678
root@hadoopnode1:~# ls -ltrh /dremiocache/spilling_stlrx2540m4-12-
10g_45678/
total 4.0K
drwxr-xr-x 2 root daemon 4.0K Sep 13 16:23 1726243167416
```

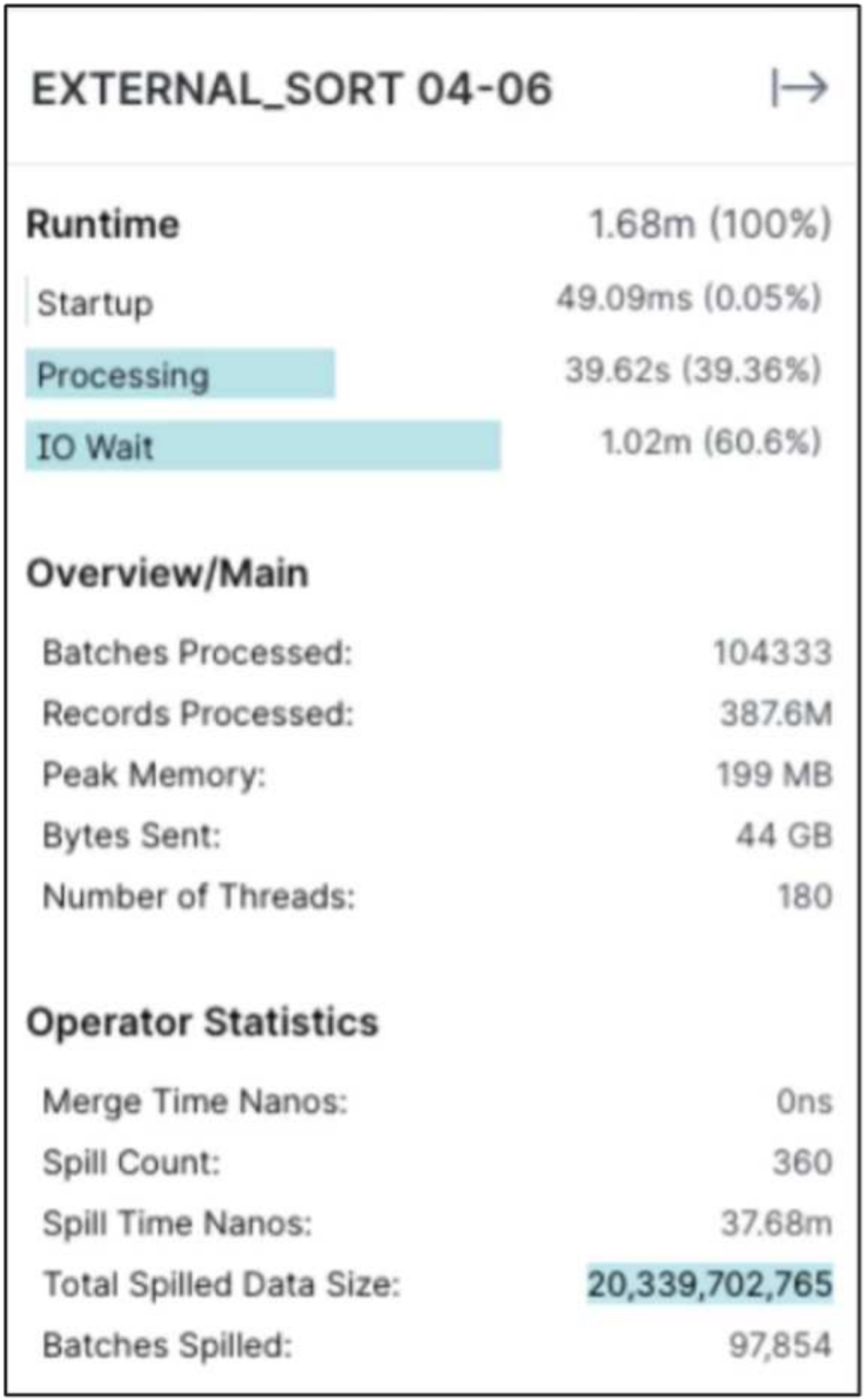
2. The SQL query completed with spill over



3. Job completion summary.



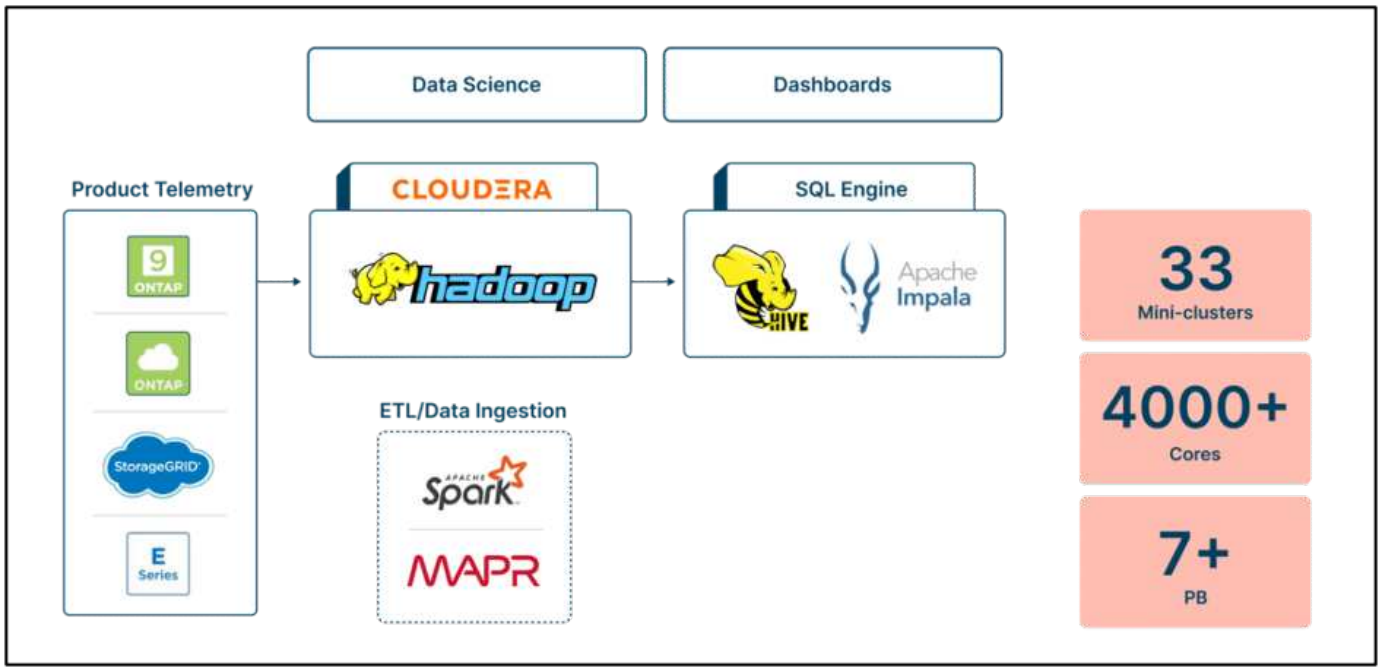
4. Check the spilled data size


EXTERNAL_SORT 04-06
Runtime
1.68m (100%)
Startup
49.09ms (0.05%)
Processing
39.62s (39.36%)
IO Wait
1.02m (60.6%)
Overview/Main
Batches Processed:
104333
Records Processed:
387.6M
Peak Memory:
199 MB
Bytes Sent:
44 GB
Number of Threads:
180
Operator Statistics
Merge Time Nanos:
0ns
Spill Count:
360
Spill Time Nanos:
37.68m
Total Spilled Data Size:
20,339,702,765
Batches Spilled:
97,854

The same procedure applicable for NAS and StorageGRID Object storage.

Customer Use Cases

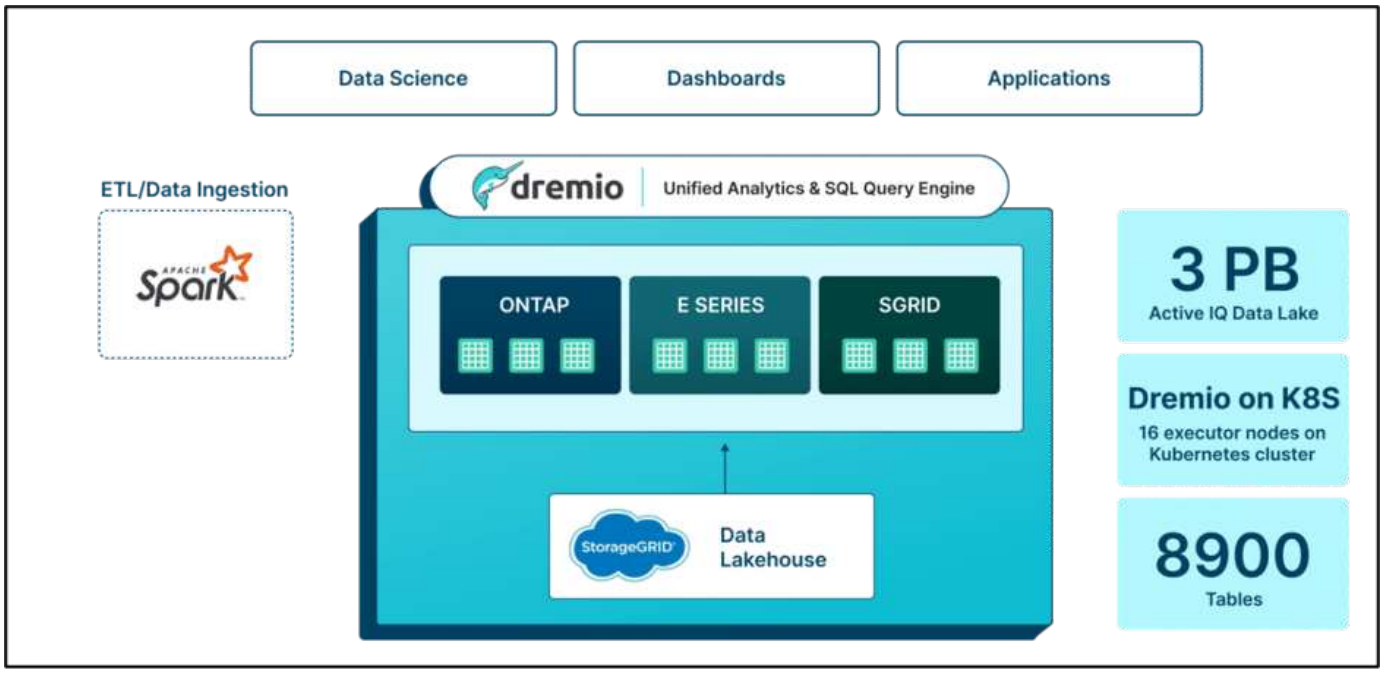
NetApp ActiveIQ use case



Challenge: NetApp's own internal Active IQ solution, initially designed for supporting numerous use cases, had evolved into a comprehensive offering for both internal users and customers. However, the underlying Hadoop/MapR-based backend infrastructure posed challenges around cost and performance, due to the rapid growth of data and the need for efficient data access. Scaling storage meant adding unnecessary compute resources, resulting in increased costs.

Additionally, managing the Hadoop cluster was time-consuming and required specialized expertise. Data performance and management issues further complicated the situation, with queries taking an average of 45 minutes and resources starvation due to misconfigurations. To address these challenges, NetApp sought an alternative to the existing legacy Hadoop environment and determined a new modern solution built on Dremio would reduce costs, decouple storage and compute, improve performance, simplify data management, offer fine-grained controls, and provide disaster recovery capabilities.

Solution:



Dremio enabled NetApp to modernize its Hadoop-based data infrastructure in a phased approach, providing a roadmap for unified analytics. Unlike other vendors that required significant changes to data processing, Dremio seamlessly integrated with existing pipelines, saving time and expenses during migration. By transitioning to a fully-containerized environment, NetApp reduced management overhead, improved security, and enhanced resilience. Dremio's adoption of open ecosystems like Apache Iceberg and Arrow ensured future-proofing, transparency, and extensibility.

As a replacement for the Hadoop/Hive infrastructure, Dremio offered functionality for secondary use cases through the semantic layer. While the existing Spark-based ETL and data ingestion mechanisms remained, Dremio provided a unified access layer for easier data discovery and exploration without duplication. This approach significantly reduced data replication factors and decoupled storage and compute.

Benefits:
With Dremio, NetApp achieved significant cost reductions by minimizing compute consumption and disk space requirements in their data environments. The new Active IQ Data Lake is comprised of 8,900 tables holding 3 petabytes of data, compared to the previous infrastructure with over 7 petabytes. The migration to Dremio also involved transitioning from 33 mini-clusters and 4,000 cores to 16 executor nodes on Kubernetes clusters. Even with the significant decrease in compute resources, NetApp experienced remarkable performance improvements. By directly accessing data through Dremio, query runtime decreased from 45 minutes to 2 minutes, resulting in 95% faster time to insights for predictive maintenance and optimization. The migration also yielded a more than 60% reduction in compute costs, more than 20 times faster queries, and more than 30% savings in total cost of ownership (TCO).

==Auto Parts Sales customer use-case.

Challenges: Within this global auto parts sales company, executive and corporate financial planning and analysis groups were unable to get a consolidated view of sales reporting and were forced into reading individual line of business sales metrics reports and attempting to consolidate them. This resulted in customers making decisions with data that was one day old. The lead times to get new analytical insights would typically take over four weeks. Troubleshooting data pipelines would require even more time, adding an additional three days or more to the already long timeline. The slow report development process as well as report performance forced the analyst community to continually wait for data to process or load, rather than enabling them to find new business insights and drive new business behavior. The troubled environments were composed of numerous different databases for different lines of businesses, resulting in numerous data silos. The slow and fragmented environment complicated data governance as there were too many ways for analysts to come up

with their own version of the truth versus a single source of truth. The approach cost over $1.9 million in data platform & people costs. Maintaining the legacy platform and filling data requests required seven Field Technical Engineers (FTE)s per year. With data requests growing, data intelligence team could not scale the legacy environment to meet future needs
Solution: Cost-effectively store and manage large Iceberg tables in Object Store (NetApp). Build data domains within Dremio's semantic layer, allowing business users to easily create, search, and share data products.

Benefits to customer:
• Improved and optimized existing data architecture and reduced time to insights from four weeks to just hours
• Reduced troubleshooting time from three days to only hours
• Decreased data platform and management costs by more than $380,000
• (2) FTEs of Data Intelligence effort saved per year

Conclusion

In conclusion, this technical report has provided comprehensive deployment details of Dremio in conjunction with various data sources from NetApp storage controllers, including ONTAP S3, NAS, and StorageGRID. The deployment process was successfully executed, and the TPC-DS benchmarking tool was utilized to perform 99 SQL queries across the different data sources. The report has also explored customer use cases within NetApp, demonstrating the versatility and effectiveness of Dremio in meeting diverse business requirements. Additionally, a specific use case involving an auto parts sales customer was examined, highlighting the practical application and benefits of leveraging Dremio for data analytics and insights.

Overall, this document serves as a valuable resource for understanding the deployment and usage of Dremio with NetApp storage controllers, showcasing its capabilities and potential for driving data-driven decision-making and optimization in various industries.

Where to find additional information

To learn more about the information that is described in this document, review the following documents and/or websites:

- Zookeeper installation

  https://medium.com/@ahmetfurkandemir/distributed-hadoop-cluster-1-spark-with-all-dependincies-03c8ec616166

- Dremio

  https://docs.dremio.com/current/get-started/cluster-deployments/deployment-models/standalone/standalone-tarball/

- Configuring Dremio with storageGRID

  https://docs.netapp.com/us-en/storagegrid-enable/tools-apps-guides/configure-dremio-storagegrid.html#configure-dremio-data-source

- NetApp use case

  https://www.dremio.com/customers/netapp/

Copyright information

Trademark information