HITACHI
Inspire the Next

Hitachi Analytics Infrastructure for Apache Ozone

Reference Architecture Guide

MK-SL-231
July 2021

Hitachi reserves the right to make changes to this Material at any time without notice and assumes no responsibility for its use. The Materials contain the most current information available at the time of publication.

Some of the features described in the Materials might not be currently available. Refer to the most recent product announcement for information about feature and product availability, or contact Hitachi Vantara LLC at https://support.hitachivantara.com/en_us/contact-us.html.

Notice: Hitachi products and services can be ordered only under the terms and conditions of the applicable Hitachi agreements. The use of Hitachi products is governed by the terms of your agreements with Hitachi Vantara LLC.

By using this software, you agree that you are responsible for:

1. Acquiring the relevant consents as may be required under local privacy laws or otherwise from authorized employees and other individuals; and
2. Verifying that your data continues to be held, retrieved, deleted, or otherwise processed in accordance with relevant laws.

Notice on Export Controls. The technical data and technology inherent in this Document may be subject to U.S. export control laws, including the U.S. Export Administration Act and its associated regulations, and may be subject to export or import regulations in other countries. Reader agrees to comply strictly with all such regulations and acknowledges that Reader has the responsibility to obtain licenses to export, re-export, or import the Document and any Compliant Products.

Feedback

Hitachi Vantara welcomes your feedback. Please share your thoughts by sending an email message to SolutionLab@HitachiVantara.com. To assist the routing of this message, use the paper number in the subject and the title of this white paper in the text.

Revision history

Changes	Date
Initial release	July 9, 2021

Reference Architecture Guide

Hitachi Analytics Infrastructure for Apache Ozone provides guidelines for deploying Apache Ozone. This reference architecture covers a SAN storge-based solution. For comparison, a local storage solution is also presented.

Apache Hadoop infrastructure is designed within the concepts of a distributed environment with data and compute located on the same nodes. The core components in this environment are MapReduce, a batch-based processing environment, and Hadoop Distributed File System (HDFS), a distributed storage environment. Since Apache Hadoop's first release in 2006, the compute landscape has shifted away from MapReduce towards Apache Spark and other Apache projects that do not take advantage of being collocated with HDFS. Some of the key factors of the migration are:

- Easier to manage and code
- Supports many different workloads, including interactive or real-time processing. Not just batch processing
- Increased performance
- More efficient use of hardware
- Uses diverse data from multiple source systems, not just HDFS

With this change in the compute element of the applications deployed, it is natural that the storage part of the solution be reviewed. Apache Ozone is designed to not only to solve problems with HDFS designs, but to also provide features to work with modern application environments. Some of the key points of Apache Ozone are:

- Works well with all file sizes - HDFS is designed to work with large files but is inefficient with small files, like those that are generated with streaming applications.
- Requires fewer computers - Because of the Apache HDFS design, Cloudera does not support more than 100 TB per node — they recommend 50 TB or less. With Apache Ozone hundreds of terabytes per node are supported.
- Improved Scalability - Apache Ozone scales to tens of billions of files.
- Supports additional interfaces besides the original HDFS interface - In addition to having an Apache HDFS-compliant interface, Apache Ozone has an Amazon S3 interface, Container Storage Interface (CSI), and a native Java API.
- Improved security - Apache Ozone integrates with Kerberos, Transparent Data Encryption, and on-wire encryption.

One important thing to know about Apache Ozone is not only is it Cloudera's direction, but also in some instances the Cloudera Data Platform already requires it.

The customer benefits of Hitachi Analytics Infrastructure for Apache Ozone are:

- Reduced hardware costs
- Reduced footprint and energy consumption in the data center
- Reduced software licensing costs
- Increased reliability with SAN storage

This reference architecture documents a deployment of Hitachi Analytics Infrastructure for Apache Ozone using SAN storage. For comparison's sake, a local storage solution is also shown. Apache Ozone is normally deployed as part of a much larger solution alongside Cloudera Data Platform. This architecture consists of the following components:

- Hitachi Virtual Storage Platform G350 (VSP G350) for storage
- Hitachi Advanced Server systems for compute and optionally storage
- Hitachi UCP Advisor for end-to-end management
- Cisco Nexus 9000 for Ethernet networking

The intended audience of this document is IT administrators, system architects, consultants, and sales engineers to assist in planning, designing, and implementing an Apache Ozone deployment.

Note: Testing of this configuration was in a lab environment. Many things affect production environments beyond prediction or duplication in a lab environment. Follow the recommended practice of conducting proof-of-concept testing for acceptable results in a non-production, isolated test environment that otherwise matches your production environment before your production implementation of this solution.

Solution overview

Hitachi Analytics infrastructure for Apache Ozone is a solution optimized for big data storage on Apache Ozone. Depending on the deployment purpose, a wide variety of other Apache Hadoop components are deployed with it.

Apache Ozone

Apache Ozone was introduced to address the issues that have arisen around HDFS (Hadoop Distributed File System), a component of Apache Hadoop. Introduced in 2006 as one of the components of the original Apache Hadoop release, HDFS has shown its age, with the following notable issues:

- Tops out at around 400 million files
- Inefficient for small files (<10 MB)
- Max storage density of 100 TB per node
- The industry is moving away from the MapReduce component in favor of Apache Spark
- Does not natively support new computing models such as containers

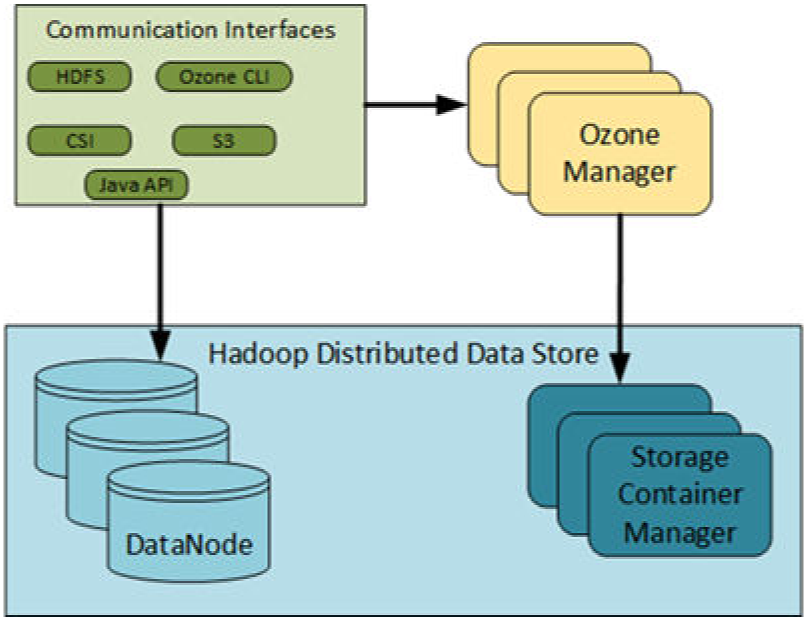
The Ozone project started in 2016 and was made generally available as version 1.0 in 2020. It includes features such as:

- Distributed Key-Value Object storage for Hadoop and other big data systems.
- High Availability and Scalability to hundreds of billions of objects.
- Native support for container environments like Kubernetes and YARN.
- Directly supports Hadoop, Spark, and Hive.
- POSIX compliant, interfaces with HDFS, S3, and CSI.
- On average, about 3.5% faster than HDFS across multiple test cases.
- Storage density of 300 TB per node with future releases aiming for greater density.

A technical comparison is provided in the following table:

Feature	Apache HDFS	Apache Ozone
Origin	First released in 2006 as part of the original Apache Hadoop distribution. Based on the Google File System.	First released in 2020 as a successor to Apache HDFS
Design	Chunk (Block) based distributed FS	Distributed Key-Value Object store
File Space Scalability	Tops out around 400 million files	Supports over 100 billion objects
File Size Handling	Inefficient for files < 10 MB	N/A
Consistency	Automatic chunk replication	Strictly Serializable
Interfaces	HDFS	HDFS, POSIX, S3, CSI
Storage/Node	100 TB max, preference is under 50 TB	300 Tb per node, large sizes planned

Ozone separates namespace management and block space management; this helps Ozone to scale. The following figure shows the main components of Apache Ozone. The namespace is managed by a daemon called Ozone Manager (OM). Block space is managed by Storage Container Manager (SCM); the block space is stored on the data nodes, and there are various communication interfaces. Hadoop Distributed Data Store (HDDS) consists of the data nodes and Storage Container Manager.



Apache Hadoop

The Apache Hadoop project develops open-source software for reliable, distributed computing.

The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, thus delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures.

The Hadoop Distributed File System (HDFS) is a distributed file system designed to run on commodity hardware. It has many similarities with existing distributed file systems. However, the differences from other distributed file systems are significant. HDFS is highly fault-tolerant and is designed to be deployed on low-cost hardware. HDFS provides high throughput access to application data and is suitable for applications that have large data sets. HDFS relaxes a few POSIX requirements to enable streaming access to file system data.

The Apache Hadoop project includes these modules:

- Hadoop Common — The common utilities that support the other Hadoop modules.
- Hadoop Distributed File System (HDFS) — A distributed file system that provides high-throughput access to application data.
- Hadoop YARN — This is a framework for job scheduling and cluster resource management.
- Hadoop MapReduce — This is a YARN-based system for parallel processing of large data sets.

When deploying this solution, it is very likely that other related Apache Projects will be used. Some of the common projects deployed are:

- Apache Hive is data warehouse software that facilitates reading, writing, and managing large datasets residing in distributed storage using SQL.
- Apache Kafka is a distributed streaming platform.
- Apache Spark is a fast, general-purpose engine for large-scale data processing.
- Apache HBase is a datastore built on top of HDFS.

Cloudera Data Platform

Cloudera Data Platform Private Cloud (CDP) is Cloudera's latest enterprise Apache Hadoop solution. It merges Cloudera Data Hub and Hortonworks Data Platform, and it provides public and private cloud support. The most common deployments of Apache Ozone are with CDP.

CDP Private Cloud has two options:

- CDP Private Cloud Base Edition – This provides traditional bare metal Apache Hadoop. It is also used to store the data that is used by CDP Private Cloud Plus Edition.
- CDP Private Cloud Plus Edition – This provides containerized compute processing that integrates with CDP Private Cloud Base Edition. For easier use, Cloudera bundles software into experiences that solutions can be built around.

Key solution components

The computing platform used in this reference architecture is provided by Hitachi Analytics Infrastructure, where a wide range of 1U and 2U servers are available for the Apache Ozone Manager and Apache Ozone Storage Container Manager.

Hardware components

The following tables describe the standard configuration for Apache Ozone with SAN storage. The optional choices are Apache Ozone Manager and Apache Ozone Storage Container Manger, and they are only provided with the first rack. The current release of Apache Storage Container Manager does not support high availability.

Note: Select one of the Hitachi Advanced Server DS120 configurations listed. When scaling out to more than one rack Ozone Manager and Storage Container Manager are optional.

Hardware	Description	Quantity	Details
Hitachi Advanced Server DS120	Apache Ozone Manager	0 or 3	▪ 2 × Intel Xeon Gold 6240R (24C 2.4G 165W) ▪ 1 × Broadcom SAS3516 2GB RAID Mezzanine for SFF Chassis ▪ 1 × Mellanox ConnectX-4 LX EN Dual Port 25GbE OCP Mezzanine ▪ 1 × QS3516 SUPER CAP W/ CABLE SP ▪ 16 × 32 GB DDR4 R-DIMM 2933MHz ▪ 2 × 1.92 TB SATA 6 Gbps 3DWDP SFF SSD S4610
Hitachi Advanced Server DS120	Apache Ozone Storage Container Manager	0, 1, or 3	▪ 2 × Intel Xeon Gold 6240R (24C 2.4G 165W) ▪ 1 × Broadcom SAS3516 2GB RAID Mezzanine for SFF Chassis ▪ 1 × Mellanox ConnectX-4 LX EN Dual Port 25 GbE OCP Mezzanine ▪ 1 × QS3516 SUPER CAP W/ CABLE SP ▪ 16 × 32 GB DDR4 R-DIMM 2933MHz ▪ 2 × 1.92 TB SATA 6 Gbps 3DWDP SFF SSD S4610
Hitachi Advanced Server DS120	Apache Ozone Data Nodes	8	▪ 2 × Intel Xeon Gold 6240R (24C 2.4G 165W) ▪ 1 × Broadcom SAS3516 2GB RAID Mezzanine for SFF Chassis ▪ 1 × Mellanox ConnectX-4 LX EN Dual Port 25 GbE OCP Mezzanine ▪ 1 × LPe32002-M2 - LIGHTPULSE 2P 32 GB Fibre Channel ADAPTER

Hardware	Description	Quantity	Details
			▪ 1 × QS3516 SUPER CAP W/ CABLE SP ▪ 16 × 32 GB DDR4 R-DIMM 2933MHz ▪ 2 × 1.92 TB SATA 6 Gbps 3DWDP SFF SSD S4610
Hitachi Virtual Storage Platform G350	Ozone Storage	1	▪ 1 × VSP G350 Covered Product Unified (FC/iSCSI) ▪ 1 × VSP G350 L Foundation Base Package ▪ 4 (trays) × 15 x VSP GXX0 HDD Pack 4 × 14TB NL-SAS HDD (240 total) ▪ 1 × VSP G350 Product Unified (FC/iSCSI) ▪ 4 × VSP GXX0 Drive Box (DBL) (dens storage, 60 LFF per box) ▪ 16 × VSP G SFP for 32Gbps Shortwave ▪ 4 × VSP G/FXX0 Host I/O Module Fibre Channel 16/32G 4 ports ▪ 1 × VSP G/FXX0 SVP3 - Service Processor
Cisco Nexus 93180YC-FX switch	ToR Data	2	▪ 48-port 10/25 GbE ▪ 6-port 40/100 GbE
Cisco Nexus 92348 switch	Management	1	▪ 48-port 1 GbE ▪ 4-port 1/10/25 GbE ▪ 2-port 40/100 GbE
Rack		1	▪ 1 × rack

For comparison, the following table shows a sample Apache Ozone configuration with local storage.

Hardware	Description	Quantity	Details
Hitachi Advanced Server DS120	Apache Ozone Manager	0 or 3	▪ 2 × Intel Xeon Gold 6240R (24C 2.4G 165W) ▪ 1 × Broadcom SAS3516 2 GB RAID Mezzanine for SFF Chassis ▪ 1 × Mellanox ConnectX-4 LX EN Dual Port 25 GbE OCP Mezzanine ▪ 1 × QS3516 SUPER CAP W/ CABLE SP ▪ 16 × 32 GB DDR4 R-DIMM 2933MHz ▪ 2 × 1.92 TB SATA 6 Gbps 3DWDP SFF SSD S4610
Hitachi Advanced Server DS120	Apache Ozone Storage Container Manager	0, 1, or 3	▪ 2 × Intel Xeon Gold 6240R (24C 2.4G 165W) ▪ 1 × Broadcom SAS3516 2 GB RAID Mezzanine for SFF Chassis ▪ 1 × Mellanox ConnectX-4 LX EN Dual Port 25 GbE OCP Mezzanine ▪ 1 × QS3516 SUPER CAP W/ CABLE SP ▪ 16 × 32 GB DDR4 R-DIMM 2933MHz ▪ 2 × 1.92 TB SATA 6 Gbps 3DWDP SFF SSD S4610
Hitachi Advanced Server DS120	Apache Ozone Data Nodes	24	▪ 2 × Intel Xeon Gold 6240R (24C 2.4G 165W) ▪ 1 × Broadcom SAS3516 2 GB RAID Mezzanine for SFF Chassis ▪ 1 × Mellanox ConnectX-4 LX EN Dual Port 25GbE OCP Mezzanine ▪ 1 × QS3516 SUPER CAP W/ CABLE SP

Hardware	Description	Quantity	Details
			▪ 16 × 32 GB DDR4 R-DIMM 2933MHz ▪ 2 × 1.92 TB SATA 6 Gbps 3DWDP SFF SSD S4610 ▪ 12 × 8 TB LFF HDD
Cisco Nexus 93180YC-FX switch	ToR Data	4	▪ 48-port 10/25 GbE ▪ 6-port 40/100 GbE
Cisco Nexus 92348 switch	Management	2	▪ 48-port 1 GbE ▪ 4-port 1/10/25 GbE ▪ 2-port 40/100 GbE
Rack		1	▪ 1 × rack

Software components

The following table lists the key software components used in this solution:

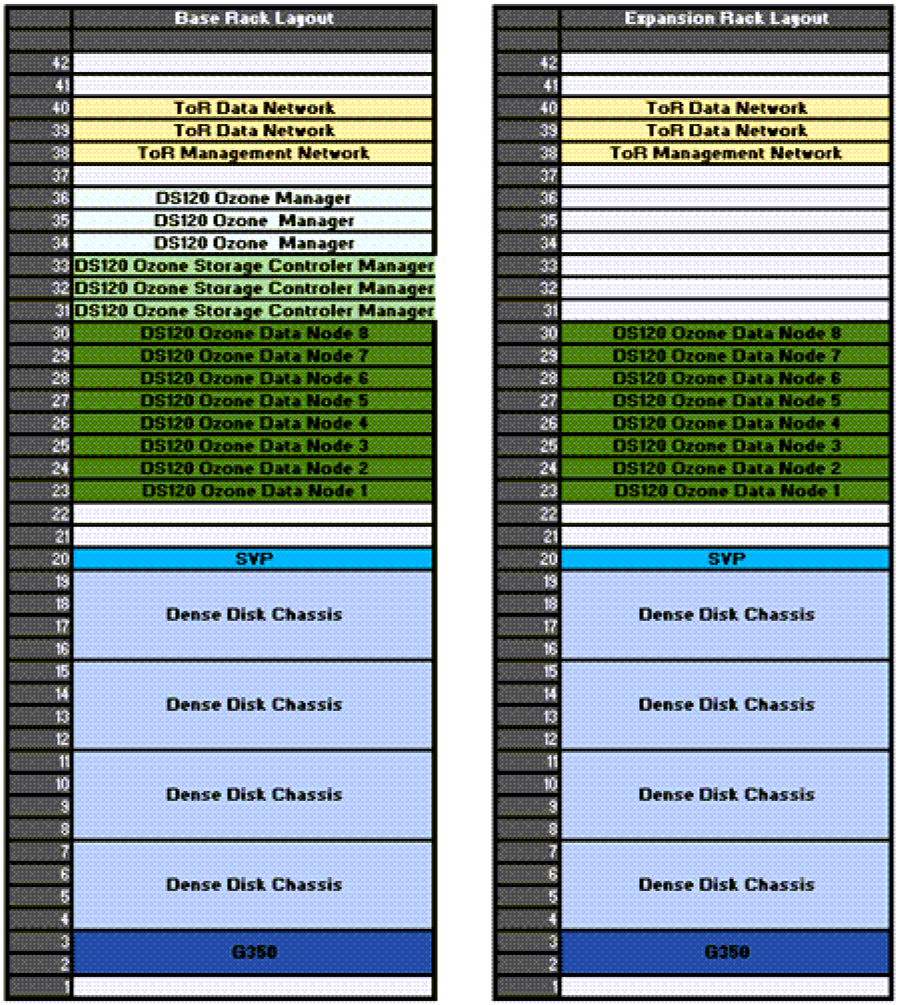
Software	Version
Hitachi Storage Virtualization Operating System RF	90-03-02-00/00 93-02-01-60/00
Red Hat Enterprise Linux	7.8
Apache Ozone	1.0.0.7.1.6.0-297
Cloudera Private Cloud	7.1.16
Cloudera Manager	2.3.1

Solution design for Apache Ozone with SAN

The detailed design for Hitachi Solution for Apache Ozone with SAN includes the following.

High-level infrastructure

The following figure shows a base rack and one expansion rack deployment using SAN storage. To provide the best availability, it is recommended that you use at least 3 racks.



	Base Rack Layout
42	
41	
40	ToR Data Network
39	ToR Data Network
38	ToR Management Network
37	
36	DS120 Ozone Manager
35	DS120 Ozone Manager
34	DS120 Ozone Manager
33	DS120 Ozone Storage Controler Manager
32	DS120 Ozone Storage Controler Manager
31	DS120 Ozone Storage Controler Manager
30	DS120 Ozone Data Node 8
29	DS120 Ozone Data Node 7
28	DS120 Ozone Data Node 6
27	DS120 Ozone Data Node 5
26	DS120 Ozone Data Node 4
25	DS120 Ozone Data Node 3
24	DS120 Ozone Data Node 2
23	DS120 Ozone Data Node 1
22	
21	
20	SVP
19	Dense Disk Chassis
18	
17	
16	
15	Dense Disk Chassis
14	
13	
12	
11	Dense Disk Chassis
10	
9	
8	
7	Dense Disk Chassis
6	
5	
4	
3	G350
2	
1	

	Expansion Rack Layout
42	
41	
40	ToR Data Network
39	ToR Data Network
38	ToR Management Network
37	
36	
35	
34	
33	
32	
31	
30	DS120 Ozone Data Node 8
29	DS120 Ozone Data Node 7
28	DS120 Ozone Data Node 6
27	DS120 Ozone Data Node 5
26	DS120 Ozone Data Node 4
25	DS120 Ozone Data Node 3
24	DS120 Ozone Data Node 2
23	DS120 Ozone Data Node 1
22	
21	
20	SVP
19	Dense Disk Chassis
18	
17	
16	
15	Dense Disk Chassis
14	
13	
12	
11	Dense Disk Chassis
10	
9	
8	
7	Dense Disk Chassis
6	
5	
4	
3	G350
2	
1	

Storage architecture

This solution uses SAN storage for data and local storage for the operating system and other Ozone storage.

For local storage, each node has 2 × 1.92 TB SATA 6 Gbps 3DWDP SFF SSD S4610; these devices are configured as RAID1.

When deploying this architecture with SAN storage the following configuration is used:

- 1 × G350 per rack
- 4 × LFF dense disk chassis fully populated
  - 240 × 14 TB drives
- Parity Groups
  - RAID 6 (14+2)
  - 15 parity groups
  - 196 TB per parity group

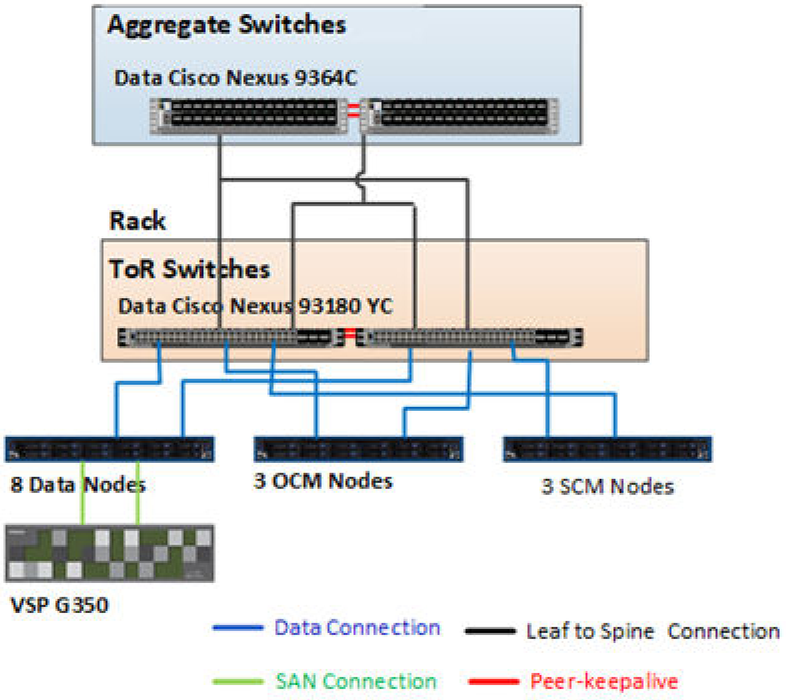
- LDEVS
  - 65 LDEVS per parity group
  - 2.99 TB per LDEV
- Pools
  - 1 Pool
  - All LDEVs are assigned to it
  - ~2,915 TB
- Virtual Volumes
  - 64 virtual volumes
  - ~45 TB per virtual volume
  - 8 virtual volumes assigned to each data node
- Each volume is assigned to both ports and attached to one node

The following table lists the connections between the Apache Ozone data nodes and the VSP G350 storage used in the solution.

Node	Storage Port Connected to Node Port 0	Storage Port Connected to Node Port 1
Data Node 1 - U23	CL1-A	CL2-A
Data Node 2 - U24	CL3-A	CL4-A
Data Node 3 - U25	CL5-A	CL6-A
Data Node 4 - U26	CL7-A	CL8-A
Data Node 5 - U27	CL1-B	CL2-B
Data Node 6 - U28	CL3-B	CL4-B
Data Node 7 - U29	CL5-B	CL6-B
Data Node 8 - U30	CL7-B	CL8-B

Network architecture

The following figure shows the network topology of Hitachi Analytics Infrastructure for Apache Ozone. There are dual 25 GB connections from all nodes to the switches in an active-active configuration.



Engineering validation

The following was done to validate and test this solution.

Test hardware

Test system consisted of the following with any changes from the standard solution noted.

- 1 × OCM node
- 1 × SC node
- 6 × Cloudera/ test driver nodes
- 8 × Ozone data nodes
- 1 × VSP G350
- 4 × dense disk cabinets
  - 240 × 6 TB drives
  - 6 × TB drives, this reduces the number of LDEVs per parity group to 25

Apache Ozone/Cloudera data platform configuration

Before installing the operating system, the SAN was configured and connected to the nodes. When installing the nodes, a minimal installation with OpenJDK 1.8 was selected.

The first post installation step was to configure multipathing access to the SAN on each data node.

Procedure

1. Enable multipath in the operating system.

```
mpathconf --enable --with_multipathd y
systemctl start multipathd.service
systemctl enable multipathd.service
```

2. Edit the multipath configuration file, */etc/multipath*, replacing it with the following:

```
devices
 {
    device {
        vendor                    "HITACHI"
        product                   ".*"
        user_friendly_names       "no"
        path_checker              "tur"
        path_grouping_policy      "multibus"
        path_selector             "queue-length 0"
        uid_attribute             "ID_SERIAL"
        failback                  "immediate"
        rr_weight                 "uniform"
        rr_min_io_rq              "128"
        features                  "0"
        no_path_retry             "5"
    }
}
blacklist {
    devnode                   "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-
9]*"
    devnode                   "^hd[a-z]"
    devnode                   "^dcssblk[0-9]*"
}
```

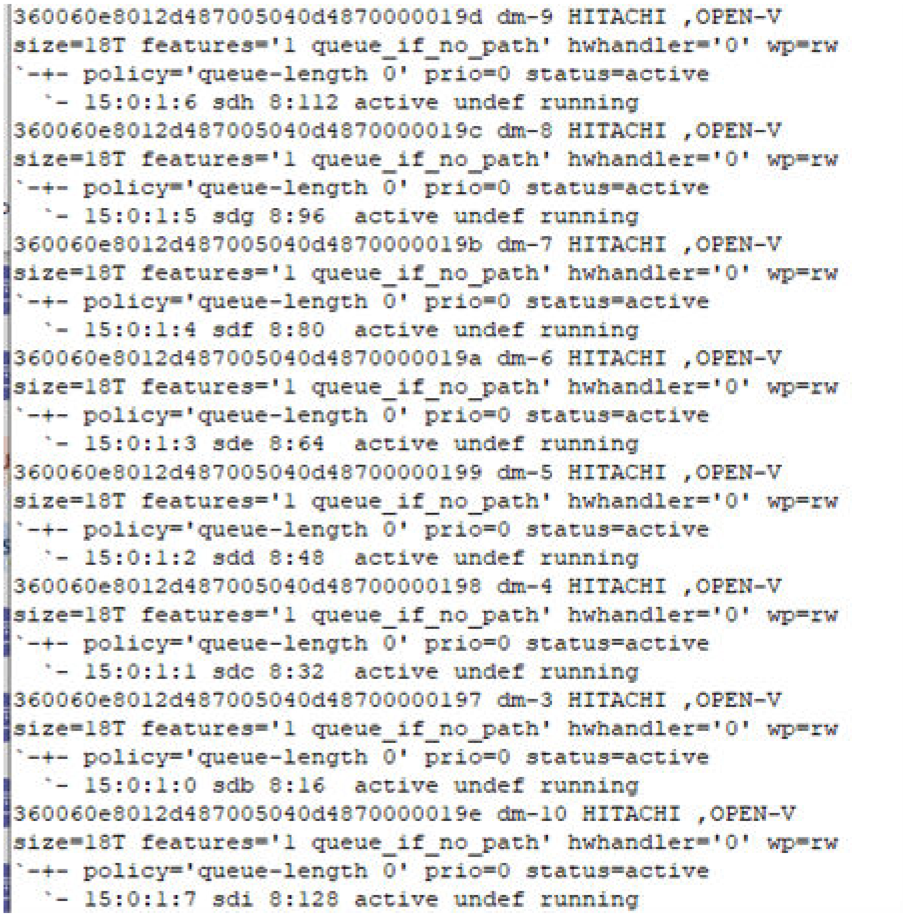
3. Restart multipathing.

```
systemctl start multipathd.service
```

4. Verify multipathing.

```
multipath -l
```

The output is shown in the following figure. It displays the SAN device and the local multipath device it maps to.



```
360060e8012d487005040d4870000019d dm-9 HITACHI ,OPEN-V
size=18T features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='queue-length 0' prio=0 status=active
  `- 15:0:1:6 sdh 8:112 active undef running
360060e8012d487005040d4870000019c dm-8 HITACHI ,OPEN-V
size=18T features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='queue-length 0' prio=0 status=active
  `- 15:0:1:5 sdg 8:96  active undef running
360060e8012d487005040d4870000019b dm-7 HITACHI ,OPEN-V
size=18T features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='queue-length 0' prio=0 status=active
  `- 15:0:1:4 sdf 8:80  active undef running
360060e8012d487005040d4870000019a dm-6 HITACHI ,OPEN-V
size=18T features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='queue-length 0' prio=0 status=active
  `- 15:0:1:3 sde 8:64  active undef running
360060e8012d487005040d48700000199 dm-5 HITACHI ,OPEN-V
size=18T features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='queue-length 0' prio=0 status=active
  `- 15:0:1:2 sdd 8:48  active undef running
360060e8012d487005040d48700000198 dm-4 HITACHI ,OPEN-V
size=18T features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='queue-length 0' prio=0 status=active
  `- 15:0:1:1 sdc 8:32  active undef running
360060e8012d487005040d48700000197 dm-3 HITACHI ,OPEN-V
size=18T features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='queue-length 0' prio=0 status=active
  `- 15:0:1:0 sdb 8:16  active undef running
360060e8012d487005040d4870000019e dm-10 HITACHI ,OPEN-V
size=18T features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='queue-length 0' prio=0 status=active
  `- 15:0:1:7 sdi 8:128 active undef running
```

5. For each Device Mapper Multipathing device (dm#) shown in the previous command, create the file system with a command similar to the following:

```
mkfs -t ext4 /dev/dm-3
```

6. Create the directories to use for mounting the filesystem.

```
mkdir -p /ozone_data/data{1..8}
```

7. Add the mount points to */etc/fstabs*
8. Mount all the devices.

```
mount -a
```

9. Reconfigure all nodes to follow Cloudera's best practices.
   a. Set *vm.swappiness* to 1
   b. Disable transparent huge page compaction
   c. Disable tuned
   d. Disable SELinux
   e. Turn of firewall or open ports

10. Choose one of the nodes that does not have storage attached to run Cloudera Manager and install Cloudera Manager on it following Cloudera's instructions.

> **Note:** A custom deployment was used and Hadoop Core and Apache Ozone were the only components installed. This is one of Cloudera's standard options.

11. When configuring the Cloudera cluster these items are set.

    a. *Core.DefaultFS* is set to a default bucket and volume on the Ozone Controller node *Core.DefaultFS= o3fs://bucket.volume.rhel7/*

    b. The Ozone replication factor is set to 1

    c. Ozone ID is set to oz1

    d. *ozone_datanode_heap_size* is set to 64G

12. The following roll assignment was used:

    - The eight nodes attached to the SAN were assigned the Ozone data node roles
    - Three of the Ozone data nodes were assigned the S3 gateway role
    - All nodes were assigned core gateway
    - All nodes were assigned Ozone gateway
    - One of the nodes without SAN was assigned the Ozone CM role
    - One of the nodes without SAN was assigned the SCM role
    - Monitoring roles were assigned to the node running Cloudera Manager

    > **Note:** No further tuning was done for these tests.

Tests

The following were tested:

- Basic Functionality with Ozone CLI
- Apache Ozone Freon
- Apache Ozone HDFS interface with MapReduce
- Apache Ozone S3 interface

Basic functionality

These tests validate the basic functionality of Apache Ozone using its command line interface to perform basic Apache Ozone functionality. To find out more about using Ozone's command line interface see Cloudera Apache Ozone CLI.

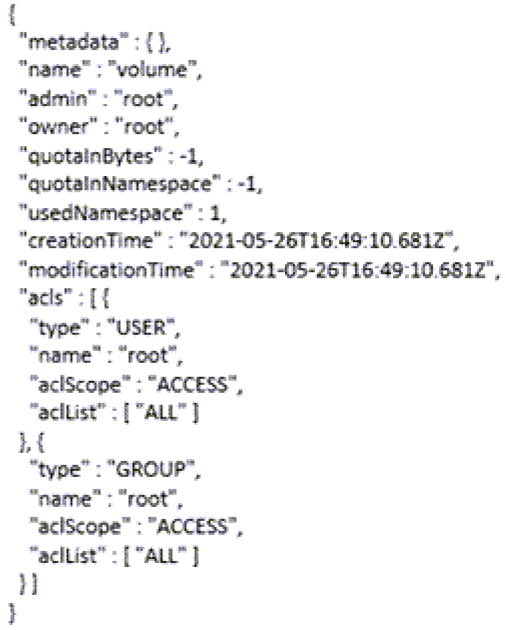
Procedure

1. Create the default bucket and volume used for *core.defaultFS*.

```
ozone sh volume create /volume
ozone sh bucket create /volume/bucket
```

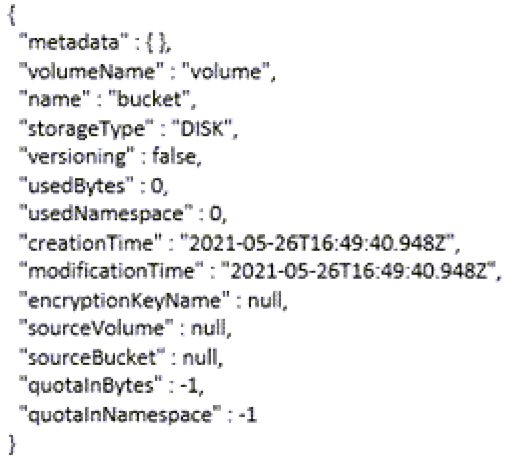
2. Verify the volume and bucket.

```
ozone sh volume list
```



```
{
  "metadata" : { },
  "name" : "volume",
  "admin" : "root",
  "owner" : "root",
  "quotaInBytes" : -1,
  "quotaInNamespace" : -1,
  "usedNamespace" : 1,
  "creationTime" : "2021-05-26T16:49:10.681Z",
  "modificationTime" : "2021-05-26T16:49:10.681Z",
  "acls" : [ {
    "type" : "USER",
    "name" : "root",
    "aclScope" : "ACCESS",
    "aclList" : [ "ALL" ]
  }, {
    "type" : "GROUP",
    "name" : "root",
    "aclScope" : "ACCESS",
    "aclList" : [ "ALL" ]
  } ]
}
```

```
ozone sh bucket list volume
```



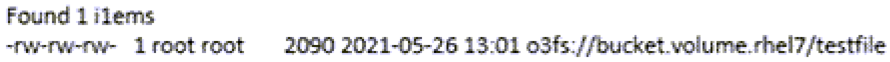
```
{
  "metadata" : { },
  "volumeName" : "volume",
  "name" : "bucket",
  "storageType" : "DISK",
  "versioning" : false,
  "usedBytes" : 0,
  "usedNamespace" : 0,
  "creationTime" : "2021-05-26T16:49:40.948Z",
  "modificationTime" : "2021-05-26T16:49:40.948Z",
  "encryptionKeyName" : null,
  "sourceVolume" : null,
  "sourceBucket" : null,
  "quotaInBytes" : -1,
  "quotaInNamespace" : -1
}
```

3. Put a file into a bucket.

```
ozone fs –put mytestfiles o3fs://bucket.volume.testfile
```

4. List the bucket's directory contents using Ozone.

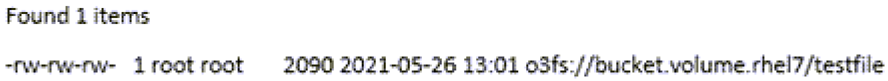
```
ozone fs -ls o3fs://bucket.volume.rhel7/
```



```
Found 1 i1ems
-rw-rw-rw-   1 root root      2090 2021-05-26 13:01 o3fs://bucket.volume.rhel7/testfile
```

5. List the directory using HDFS.

```
hdfs dfs -ls /
```



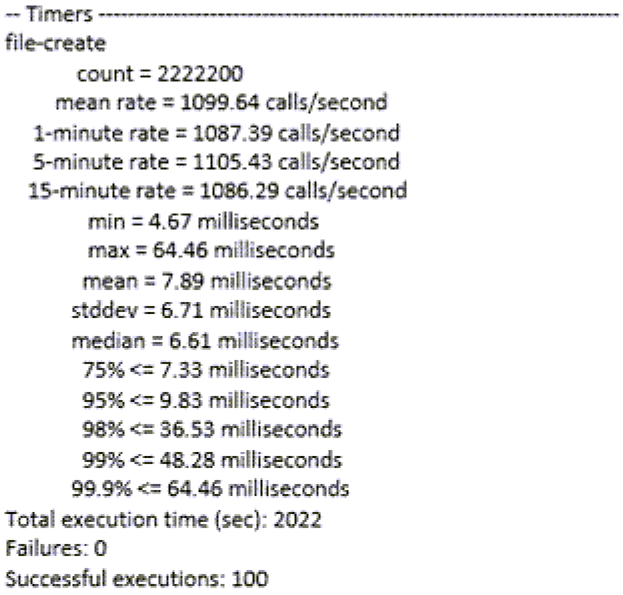
```
Found 1 items

-rw-rw-rw-   1 root root      2090 2021-05-26 13:01 o3fs://bucket.volume.rhel7/testfile
```

Apache Ozone Freon testing

Apache Ozone Freon is a load generator used in stress tests that is included in the Ozone distribution.

The following test creates nested directories and 1024 byte files.

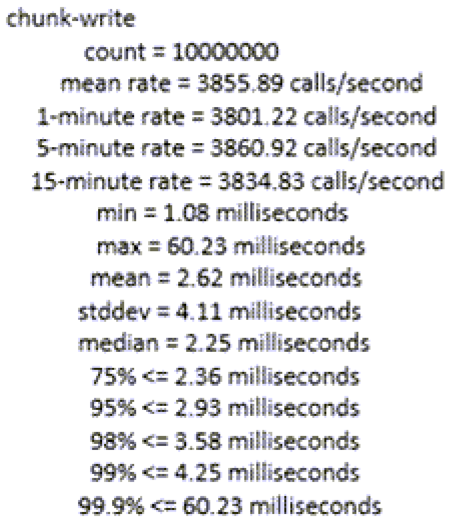
```
ozone freon dtsg --number-of-tests 100 --fileCount 2 ---depth 11111-
rpath="o3fs://bucket.volume.rhel7/dir1"
```



```
-- Timers ----------------------------------------------------------------------
file-create
             count = 2222200
         mean rate = 1099.64 calls/second
     1-minute rate = 1087.39 calls/second
     5-minute rate = 1105.43 calls/second
    15-minute rate = 1086.29 calls/second
               min = 4.67 milliseconds
               max = 64.46 milliseconds
              mean = 7.89 milliseconds
            stddev = 6.71 milliseconds
            median = 6.61 milliseconds
              75% <= 7.33 milliseconds
              95% <= 9.83 milliseconds
              98% <= 36.53 milliseconds
              99% <= 48.28 milliseconds
            99.9% <= 64.46 milliseconds
Total execution time (sec): 2022
Failures: 0
Successful executions: 100
```

The following test creates 10000000 chunks with a size of 8000.

```
ozone freon dcg -a -n 10000000 -s 8000
```

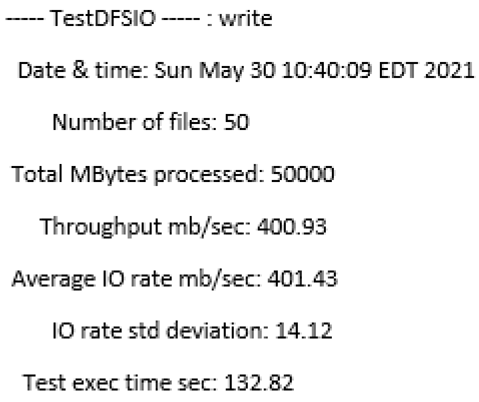


```
chunk-write
             count = 10000000
         mean rate = 3855.89 calls/second
     1-minute rate = 3801.22 calls/second
     5-minute rate = 3860.92 calls/second
    15-minute rate = 3834.83 calls/second
               min = 1.08 milliseconds
               max = 60.23 milliseconds
              mean = 2.62 milliseconds
            stddev = 4.11 milliseconds
            median = 2.25 milliseconds
              75% <= 2.36 milliseconds
              95% <= 2.93 milliseconds
              98% <= 3.58 milliseconds
              99% <= 4.25 milliseconds
            99.9% <= 60.23 milliseconds
```

Apache Ozone HDFS interface with MapReduce

The following tests with TestDFSIO are simple MapReduce jobs running against Apache Ozone. A full Hadoop cluster was not configured, therefore all map reduce jobs ran on single nodes.

```
hadoop jar ./TestDFSIO hadoop-mapreduce-client-jobclient-tests.jar -write
-nrfiles 50  -filesize 1000 -resFile  tesstdfsio-50-1000
```



```
----- TestDFSIO ----- : write
  Date & time: Sun May 30 10:40:09 EDT 2021
        Number of files: 50
 Total MBytes processed: 50000
      Throughput mb/sec: 400.93
 Average IO rate mb/sec: 401.43
    IO rate std deviation: 14.12
    Test exec time sec: 132.82
```

Apache Ozone S3 interface

S3tester and AWS S3 CLI were used for Apache Ozone testing. S3tester is an opensource tool that is used to test S3 compliant file systems. Both AWS command line interface and S3tester must be installed.

Procedure

1. Install AWS CLI. See https://docs.aws.amazon.com/cli/latest/userguide/install-cliv2-linux.html.

2. Install S3tester. See https://github.com/s3tester/s3tester.
3. Configure S3tester and AWS CLI to work without security by setting dummy values that are assigned to *aws_access_key_id* and *aws_secret_access_key*. This needs to be done using environment variables in the *~/.aws/credentials* file.

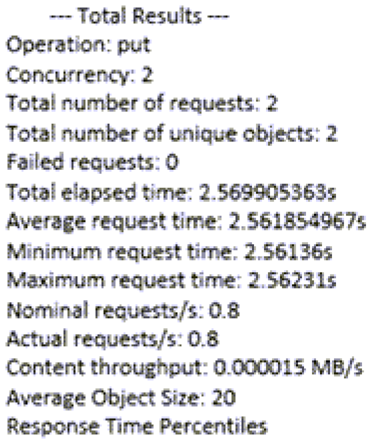
```
export AWS_ACCESS_KEY_ID=dummy
export AWS_SECRET_ACCESS_KEY=dummy
vi  ~/.aws/credentials
[default]
aws_access_key_id=dummy
aws_secret_access_key=dummy
```

4. With Apache Ozone CLI, create the test volume and bucket.

```
ozone sh volume create testv1
ozone sh bucket create /testv1/test
-- s3v is pseudo volume
ozone sh bucket link /testv1/test1 /s3v/test
```

5. When using S3tester and AWS CLI, access Ozone through an S3 gateway, the following simple test confirms that S3tester is configured to access Ozone:

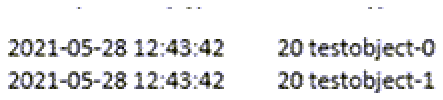
```
./s3tester -concurrency=2 -size=20 -operation=put -requests=2 -
endpoint=http://rhel15:9878 -bucket test
```



```
      --- Total Results ---
Operation: put
Concurrency: 2
Total number of requests: 2
Total number of unique objects: 2
Failed requests: 0
Total elapsed time: 2.569905363s
Average request time: 2.561854967s
Minimum request time: 2.56136s
Maximum request time: 2.56231s
Nominal requests/s: 0.8
Actual requests/s: 0.8
Content throughput: 0.000015 MB/s
Average Object Size: 20
Response Time Percentiles
```

6. Use AWS CLI to list buckets.

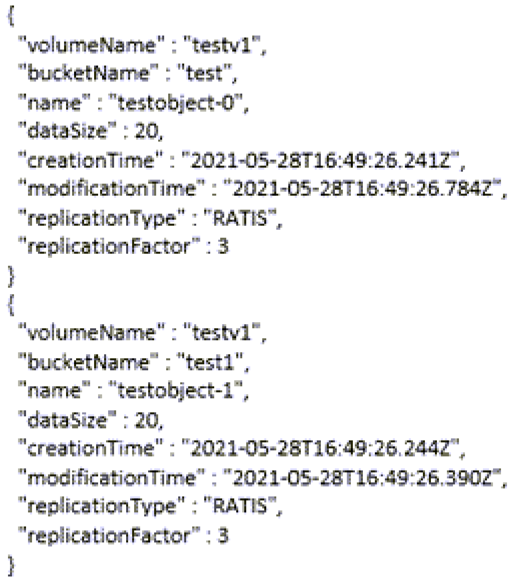
```
aws s3 ls --endpoint http://rhel15:9878 s3://test
```



```
 2021-05-28 12:43:42        20 testobject-0
 2021-05-28 12:43:42        20 testobject-1
```

7. Use Ozone CLI to view the keys.

```
ozone sh key ls /s3v/test
```

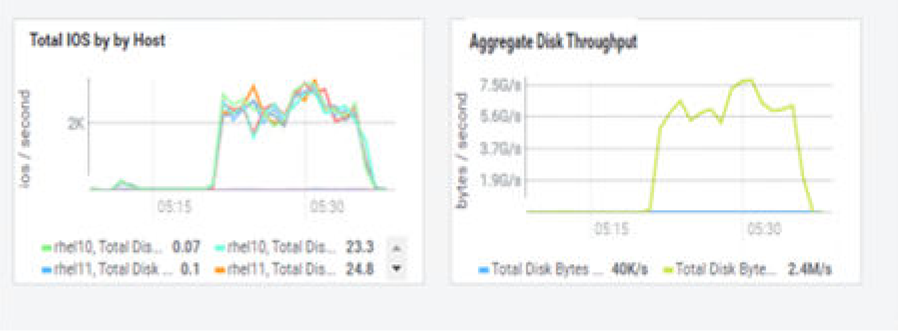


```
{
  "volumeName" : "testv1",
  "bucketName" : "test",
  "name" : "testobject-0",
  "dataSize" : 20,
  "creationTime" : "2021-05-28T16:49:26.241Z",
  "modificationTime" : "2021-05-28T16:49:26.784Z",
  "replicationType" : "RATIS",
  "replicationFactor" : 3
}
{
  "volumeName" : "testv1",
  "bucketName" : "test1",
  "name" : "testobject-1",
  "dataSize" : 20,
  "creationTime" : "2021-05-28T16:49:26.244Z",
  "modificationTime" : "2021-05-28T16:49:26.390Z",
  "replicationType" : "RATIS",
  "replicationFactor" : 3
}
```

8. The following test is done with two S3 Gateways, and 6 processes running S3tester with a command similar to the following for each process:

```
./s3tester -concurrency=10 -size=200000000 -operation=put -repeat=3 -
endpoint=http://rhel5:9878 -bucket test -prefix rhel4_inst2b
```

The aggregate IOs across the data nodes is slightly higher than 7.5 GBs.



Product descriptions

This is information about the hardware and software components used in this solution for Hitachi Analytics Infrastructure for Apache Ozone.

Hardware components

These are the hardware components used in this reference architecture.

Hitachi Advanced Server DS120

Optimized for performance, high density, and power efficiency in a dual-processor server, Hitachi Advanced Server DS120 delivers a balance of compute and storage capacity. This 1U rack-mounted server has the flexibility to power a wide range of solutions and applications.

The highly-scalable memory supports up to 3 TB using 24 slots of high-speed DDR4 memory. Advanced Server DS120 is powered by the Intel Xeon Scalable processor family for complex and demanding workloads. There are flexible OCP and PCIe I/O expansion card options available. This server supports up to 12 small form factor storage devices with up to 4 NVMe drives.

This solution allows you to have a high CPU-to-storage ratio. This is ideal for balanced and compute-heavy workloads.

Multiple CPU and storage devices are available. Contact your Hitachi Vantara sales representative to get the latest list of options.

Hitachi Virtual Storage Platform G series family

The Hitachi Virtual Storage Platform G series family enables the seamless automation of the data center. It has a broad range of efficiency technologies that deliver maximum value while making ongoing costs more predictable. You can focus on strategic projects and consolidating more workloads while using a wide range of media choices.

The benefits start with Hitachi Storage Virtualization Operating System RF. This includes an all new enhanced software stack that offers up to three times greater performance than our previous midrange models, even as data scales to petabytes

Hitachi Virtual Storage Platform G series offers support for containers to accelerate cloud-native application development. Provision storage in seconds, and provide persistent data availability, all the while being orchestrated by industry leading container platforms. Move these workloads into an enterprise production environment seamlessly, saving money while reducing support and management costs.

Cisco Nexus switches

The Cisco Nexus switch product line provides a series of solutions that make it easier to connect and manage disparate data center resources with software-defined networking (SDN). Leveraging the Cisco Unified Fabric, which unifies storage, data and networking (Ethernet/IP) services, the Nexus switches create an open, programmable network foundation built to support a virtualized data center environment.

Software components

These are the software components used in this reference architecture.

Hitachi Storage Virtualization Operating System RF

Hitachi Storage Virtualization Operating System RF powers the Hitachi Virtual Storage Platform (VSP) family. It integrates storage system software to provide system element management and advanced storage system functions. Used across multiple platforms, Storage Virtualization Operating System includes storage virtualization, thin provisioning, storage service level controls, dynamic provisioning, and performance instrumentation.

Flash performance is optimized with a patented flash-aware I/O stack, which accelerates data access. Adaptive inline data reduction increases storage efficiency while enabling a balance of data efficiency and application performance. Industry-leading storage virtualization allows SVOS RF to use third-party all-flash and hybrid arrays as storage capacity, consolidating resources for a higher ROI and providing a high-speed front end to slower, less-predictable arrays.

Hitachi Unified Compute Platform Advisor

Hitachi Unified Compute Platform Advisor (UCP Advisor) is a comprehensive cloud infrastructure management and automation software that enables IT agility and simplifies day 0-N operations for edge, core, and cloud environments. The fourth-generation UCP Advisor accelerates application deployment and drastically simplifies converged and hyperconverged infrastructure deployment, configuration, life cycle management, and ongoing operations with advanced policy-based automation and orchestration for private and hybrid cloud environments.

The centralized management plane enables remote, federated management for the entire portfolio of converged, hyperconverged, and storage data center infrastructure solutions to improve operational efficiency and reduce management complexity. Its intelligent automation services accelerate infrastructure deployment and configuration, significantly minimizing deployment risk and reducing provisioning time and complexity, automating hundreds of mandatory tasks.

Red Hat Enterprise Linux

Using the stability and flexibility of Red Hat Enterprise Linux, reallocate your resources towards meeting the next challenges instead of maintaining the status quo. Deliver meaningful business results by providing exceptional reliability on military-grade security. Use Enterprise Linux to tailor your infrastructure as markets shift and technologies evolve.

Hitachi Vantara

Corporate Headquarters
2535 Augustine Drive
Santa Clara, CA 95054 USA
HitachiVantara.com | community.HitachiVantara.com

Contact Information
USA: 1-800-446-0744
Global: 1-858-547-4526
HitachiVantara.com/contact