cloudera®

Cloudera Enterprise 5.x with EMC Isilon Scale-out Storage as DFS

Important Notice

The information in this document is subject to change without notice. Cloudera shall not be liable for any damages resulting from technical errors or omissions which may be present in this document, or from use of this document.

Cloudera, Inc.
1001 Page Mill Road, Building 2
Palo Alto, CA 94304-1008
info@cloudera.com
US: 1-888-789-1488
Intl: 1-650-843-0595
www.cloudera.com

Release Information

Date: November 19, 2015

Table of Contents

Executive Summary

This document is a high-level design and best-practices guide for deploying Cloudera Enterprise Distribution on bare-metal infrastructure with the EMC Isilon scale-out NAS solution as a shared storage back end.

Audience and Scope

This guide is for IT architects responsible for the design and deployment of infrastructure and a shared storage platform in the data center, as well as for Hadoop administrators and architects who are data center architects or engineers or collaborate with specialists in that space.

This document describes Cloudera recommendations on the following topics:

- Storage array considerations
- Data network considerations
- Hardware/platform considerations

Glossary of Terms

Term	Description
DataNode	Worker nodes of the cluster to which the HDFS data is written.
HBA	Host bus adapter. An I/O controller that is used to interface a host with storage devices.
HDD	Hard disk drive.
HDFS	Hadoop Distributed File System.
High Availability	Configuration that addresses availability issues in a cluster. In a standard configuration, the NameNode is a single point of failure (SPOF). Each cluster has a single NameNode, and if that machine or process became unavailable, the cluster as a whole is unavailable until the NameNode is either restarted or brought up on a new host. The secondary NameNode does not provide failover capability. High availability enables two NameNodes running in the same cluster: the active NameNode and the standby NameNode. The standby NameNode allows a fast failover to a new NameNode in case of machine crash or planned maintenance.
ISL	Inter-Switch Link
JBOD	Just a bunch of disks (in contrast to disks configured via software or hardware with redundancy mechanisms for data protection).
Job History Server	A process that archives job metrics and metadata. One per cluster.
NameNode	The metadata master of HDFS essential for the integrity and proper functioning of the distributed filesystem.
NIC	Network interface card.

NodeManager	The process that starts application processes and manages resources on the DataNodes.
PDU	Power distribution unit.
QJM	Quorum Journal Manager. Provides a fencing mechanism for high availability in a Hadoop cluster. This service is used to distribute HDFS edit logs to multiple hosts (at least three are required) from the active NameNode. The standby NameNode reads the edits from the JournalNodes and constantly applies them to its own namespace. In case of failover, the standby NameNode applies all of the edits from the JournalNodes before promoting itself to the active state.
QJN	Quorum JournalNodes. Nodes on which the journal services are installed.
RM	ResourceManager. The resource management component of YARN that initiates application startup and controls scheduling on the DataNodes of the cluster (one instance per cluster).
SP-SW	Spine switch.
ToR	Top of rack.
ZooKeeper	A centralized service for maintaining configuration information, naming, and providing distributed synchronization and group services.

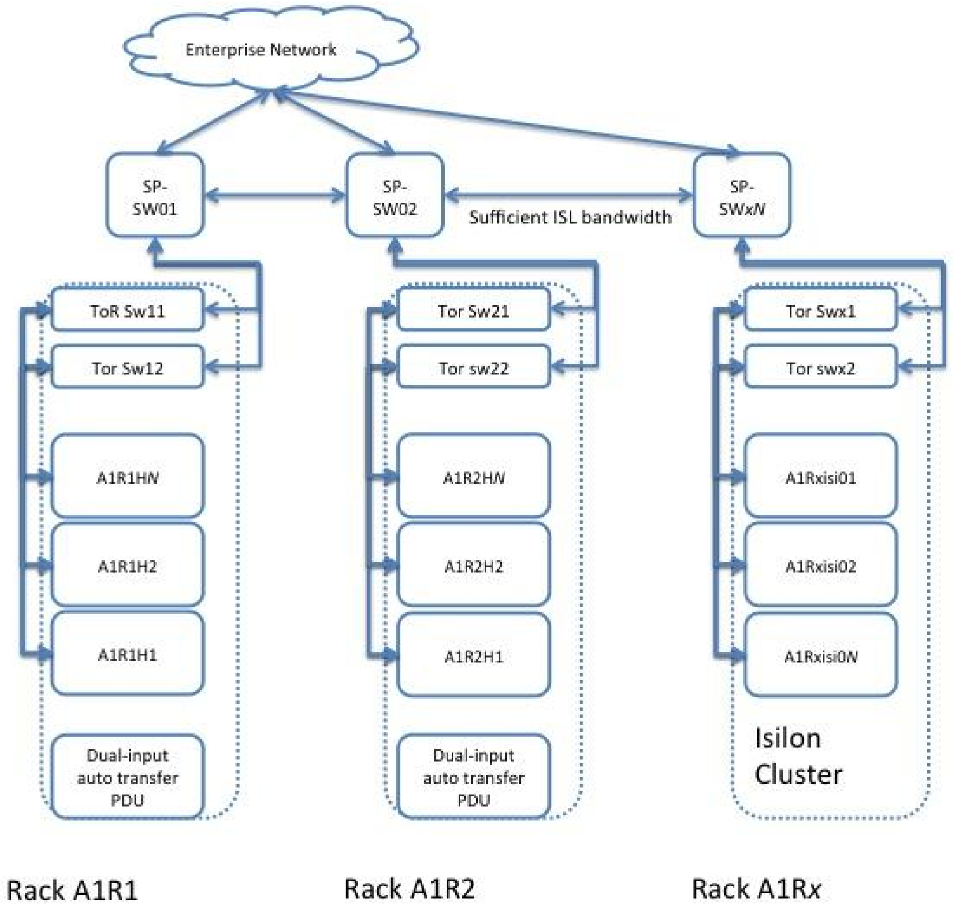
Isilon Distributed Storage Array for HDFS and Bare-Metal Nodes as Compute Nodes

In this model, Isilon replaces HDFS shipped in Cloudera Enterprise.

In this architecture, Isilon acts as the HDFS/storage layer, and the bare-metal nodes only provide the compute resources needed.

Considerations for a storage component are not required, but you must ensure a reasonable oversubscription ratio between Isilon switches and the compute node switches.

Physical Cluster Topology



Note: In each rack of compute nodes, EMC recommends interspersing Isilon storage nodes connected to the respective ToR switches (if possible). For example, for two racks of compute nodes, distribute the Isilon storage nodes between the first and second rack, sharing the ToR switches. The Isilon storage nodes share an InfiniBand backend to provide better front-end performance (10 GB Ethernet).

Physical Cluster Component List

Component	Configuration	Description	Quantity
Physical servers	Two-socket, 6-10 cores per socket > 2 GHz; minimally 256 GB RAM.	Hosts that house the various NodeManager and compute instances.	Minimum 3 master + 5 compute (8 nodes)
NICs	Dual-port 10 Gbps Ethernet NICs. The connector type depends on the network design; could be SFP+ or Twinax.	Provide the data network services	At least 2 per server.
Internal HDDs	Standard OS sizes - 300 1 TB drives. Can be larger but not necessary.	Ensure continuity of service on server resets.	2 per physical server configured as a RAID-1 volume (mirrored).
Ethernet ToR/leaf switches	Minimally 10 Gbps switches with sufficient port density to accommodate the compute cluster. These require enough ports to create a realistic spine-leaf topology providing ISL bandwidth above a 1:4 oversubscription ratio (preferably 1:1).	Although most enterprises have mature data network practices, consider building a dedicated data network for the Hadoop cluster.	At least two per rack.
Ethernet spine switches	Minimally 10 Gbps switches with sufficient port density to accommodate incoming ISL links and ensure required throughput over the spine (for inter-rack traffic).	Same considerations as for ToR switches.	Depends on the number of racks.

Logical Cluster Topology

For the YARN NodeManager instances, data protection at the HDFS level is not required, because the physical nodes are running only the compute part of the cluster.

The minimum requirements to build out the cluster are:

- Three master nodes
- The number of compute nodes depends on the cluster size (see sizing considerations section below)

The following table identifies service roles for different node types.

	Master Node	Master Node	Master Node	YARN NodeManager nodes 1..n
ZooKeeper	ZooKeeper	ZooKeeper	ZooKeeper	
YARN	ResourceManager	ResourceManage r	History Server	NodeManager
Hive			MetaStore, WebHCat, HiveServer2	

Management (misc)	Cloudera Agent	Cloudera Agent	Cloudera Agent, Oozie, Cloudera Manager, Management Services	Cloudera Agent
Navigator			Navigator, Key Management Services	
HUE			HUE	
HBASE	HMaster	HMaster	HMaster	RegionServer
Impala			StateStore, Catalog	Impala Daemon

Note: Low-latency workloads are subject to network latency, because all data traffic between compute nodes and HDFS (Isilon-based) is north-south.

The following table provides size recommendations for the physical nodes.

Component	Configuration	Description	Quantity
Master Nodes: **2-socket with 6-10 cores/socket > 2 GHz; minimally 128 GB RAM; 8-10 disks**	2RU 2-socket nodes with at least 256 GB RAM	These nodes house the Cloudera Master services and serve as the gateway/edge device that connects the rest of the customer's network to the Cloudera cluster.	3 (for scaling up to 100 cluster nodes).
Compute instances: **2-socket with 6-10 cores/socket > 2 GHz; minimally 256 GB RAM** **2 x OS disks, 8 SATA or SAS drives or 2x SSDs**	At least 8 SATA or SAS Drives, or 2 SSD drives for intermediate storage.	These nodes house the YARN node managers and any additional required services.	EMC recommends a 2:1 ratio of compute nodes to Isilon nodes for most use cases. For heavy Impala workloads, use a 1.5:1 ratio. For example, if Isilon has 5 nodes, use 8 compute nodes.

The following table provides recommendations for storage allocation.

Node/Role	Disk Layout	Description
Management/Master	• 2 x 500 GB OS (RAID 1) • Swap partition <= 2 GB • 4 x 500 GB RAID 10 (database) • 1 x 500 GB RAID 0 - ZooKeeper	Avoid fracturing the filesystem layout into multiple smaller filesystems. Instead, keep a separate "/" and "/var".
Compute nodes	• 2 x 500 GB OS (RAID 1) • Approximately 20% of total HDFS storage needs to be provisioned as intermediate storage on these nodes. The storage can either be NFS mounts from the Isilon storage array direct-attached SAS/SATA drives, or a pair of SSD drives of sufficient capacity. Distribute the 20% of capacity evenly across all the NodeManager nodes, each with its own mount-point and filesystem.	Avoid fracturing the filesystem layout into multiple smaller filesystems. Instead, keep a separate "/" and "/var". For example, for 10 TB of HDFS storage in Isilon, 2 TB is needed for intermediate storage. More or faster local spindles will speed up the intermediate shuffle stage of MapReduce.

Supportability/Compatibility Matrix

CDH	Cloudera Manager	OneFS	Supported
5.4.4 and higher (HDFS 2.6)	5.4	7.2.0.3	All services except Navigator*

*Navigator support is contingent on iNotify and fsmanage functionality being added into OneFS.

Environment Sizing and Platform Tuning Considerations

Start with the following guidelines for compute node sizing and selection. The number of isilon nodes depends on required storage capacity and backend performance considerations. Work with the Cloudera and EMC sales teams to determine backend requirements.

- Default option -- Cloudera and EMC recommend a starting configuration with a ratio of 2:1 for compute nodes to EMC Isilon storage nodes. So, if the Isilon backend has four storage nodes, use eight compute nodes.
- Heavy IO option -- When higher IO performance is required, Cloudera and EMC recommend a 1.5:1 ratio for compute nodes to Isilon storage nodes. So, for four storage nodes in the backend, use six compute nodes.

Note: These estimates are provided as a guideline. Cloudera recommends running a pilot with a preliminarily sized cluster, and then fine-tuning the requirements based on empirical data (corresponding to specific workloads).

Platform Tuning Recommendations

NOTE: This section includes general recommendations. They should be applied only after sufficient testing.

CPU

CPU BIOS Settings

In your compute nodes' BIOS, set CPU to Performance mode for best performance.

CPUfreq Governor

The following CPUfreq governor types are available in RHEL 6. (Check other OS-specific governors if you are not using CentOS or RHEL 6).

Governor Type	Description
cpufreq_performance	Forces the CPU to use the highest possible clock frequency. Intended for heavy workloads, this is best fit for interactive workloads.
Cpufreq_powersave	Forces the CPU to stay at the lowest clock frequency possible.
Cpufreq_ondemand	Allows CPU frequency to scale to maximum under heavy load, but drop down to the lowest frequency under light or no load. This is the ideal governor and, after appropriate testing, can be used to reduce power consumption under low load/idle conditions.
Cpufreq_userspace	Allows userspace programs to set the frequency. This is used in conjunction with the cpuspeed daemon.
Cpufreq_conservative	Similar to the cpufreq_ondemand, but switches frequencies more gradually.

Find the appropriate kernel modules available on the system, and then use modprobe to add the driver needed:

```
# modprobe cpufreq_performance
```

After a particular governor is loaded into the kernel, enable it:

```
# cpupower frequency-set –governor cpufreq_performance
```

Available drivers are in the /lib/modules/<kernelversion>/kernel/arch/<architecture>/kernel/cpu/cpufreq/ directory:

```
/lib/modules/2.6.32-
358.14.1.el6.centos.plus.x86_64/kernel/arch/x86/kernel/cp
u/cpufreq
# ls
acpi-cpufreq.ko  mperf.ko  p4-clockmod.ko  pcc-cpufreq.ko
powernow-k8.ko  speedstep-lib.ko
```

If the required cpufreq drivers are not available, get them from `/lib/modules/<kernel version>/kernel/drivers/cpufreq`:

```
# cd /lib/modules/2.6.32-
358.14.1.el6.centos.plus.x86_64/kernel/drivers/cpufreq
# ls
cpufreq_conservative.ko  cpufreq_ondemand.ko
cpufreq_powersave.ko  cpufreq_stats.ko  freq_table.ko
```

NOTE: Use the uname –r command to see the kernel version.

The cpupower utility is provided by the cpupowerutils package. If you have not installed it, you can set the tunables in /sys/devices/system/cpu/<cpu id>/cpufreq/.

Memory

Minimize Anonymous Page Faults

Minimize anonymous page faults, thereby freeing from page cache before “swapping” application pages. (This reduces the OOM-killer invocation.)

To minimize anonymous page faults:

1. Edit `/etc/sysctl.conf` to add following line:

```
vm.swappiness=1
```

2. Run the following command:

```
# sysctl –p
# sysctl –a|grep "vm.swappiness"
```

Disable Transparent Hugepage Compaction and Defragmentation

Add the following commands to `/etc/rc.local` to ensure that transparent hugepage compaction and defragmentation remain disabled across reboots:

```
echo "never" >
/sys/kernel/mm/redhat_transparent_hugepage/enabled
```

```
echo "never" >
/sys/kernel/mm/redhat_transparent_hugepage/defrag
```

Network

Add the following parameters to `/etc/sysctl.conf`.

Disable TCP timestamps to improve CPU utilization (optional and depends on your NIC vendor):

```
net.ipv4.tcp_timestamps=0
```

Enable TCP sacks to improve throughput:

```
net.ipv4.tcp_sack=1
```

Increase the maximum length of processor input queues:

```
net.core.netdev_max_backlog=250000
```

Increase the TCP max and default buffer sizes using setsockopt():

```
net.core.rmem_max=4194304
net.core.wmem_max=4194304
net.core.rmem_default=4194304
net.core_wmem_default=4194304
net.core.optmem_max=4194304
```

Increase memory thresholds to prevent packet dropping:

```
net.ipv4.tcp_rmem="4096 87380 4194304"
net.ipv4.tcp_wmem="4096 65536 4194304"
```

Set the socket buffer to be divided evenly between TCP window size and application buffer:

```
net.ipv4.tcp_adv_win_scale=1
```

Verify NIC Advanced Features

Determine which features are available with your NIC by using ethtool:

```
    $ sudo ethtool -k
    Features for eth0:
    rx-checksumming: on
    tx-checksumming: off
    scatter-gather: off
    tcp-segmentation-offload: off
    udp-fragmentation-offload: off
    generic-segmentation-offload: off
    generic-receive-offload: on
    large-receive-offload: off
    rx-vlan-offload: on
    tx-vlan-offload: on
    ntuple-filters: off
    receive-hashing: off
```

Modern NICs, particularly high-performance NICs, have various offload capabilities. Cloudera recommends enabling them.

In particular, tcp-segmentation-offload (TSO), scatter-gather (SG), and generic-segmentation-offload (GSO) should be enabled if not enabled by default.

NIC Ring Buffer Configurations

Check existing ring buffer sizes by running:

```
$ ethtool -g eth0
Ring parameters for eth0:
Pre-set maximums:
RX:     4096
RX Mini:    0
RX Jumbo:   0
TX:     4096
Current hardware settings:
RX:     256
RX Mini:    0
RX Jumbo:   0
TX:       256
```

After checking the preset maximum values and the current hardware settings, use the following commands to resize the ring buffers:

```
# ethtool –G <interface> rx <newsize>
```

- or -

```
# ethtool –G <interface> tx <newsize>
```

NOTE: The ring buffer sizes depend to a certain degree on network topology and might need to be tuned, depending on the nature of the workload. For 10 Gbps NICs, consider setting the RX and TX buffers to maximum. This setting may require tuning, depending on the network architecture and type of traffic.

Storage

Disk/FS Mount Options

Disable "atime" from the data disks and root FS by using the `noatime` option when mounting the FS.

In the `/etc/fstab` file, ensure that the appropriate filesystems have the `noatime` mount option specified:

```
LABEL=ROOT /        ext4    noatime        0 0
```

FS Creation Options

For FS creation:

- Enable journal mode
- Reduce superuser block reservation from 5% to 1% for root, using the `-m1` option
- Use the `sparse_super`, `dir_index`, and `extent` options to minimize number of super block backups and use b-tree indexes for directory trees and extent-based allocations)

```
# mkfs –t ext4 –m1 –O
sparse_super,dir_index,extent,has_journal /dev/sdb1
```

Cloudera Software Stack

Guidelines for installing the Cloudera stack on this platform are nearly identical to those for direct attached storage. This is addressed in various documents on the Cloudera website.

To configure the Isilon service (instead of HDFS), follow the instructions at Managing Isilon.

References

1. Managing The Isilon Service
2. Cloudera Documentation
3. EMC Hadoop Starter Kit -- Step By Step Guide To Quickly And Easily Deploy Hadoop
4. EMC HSK 3.0 For Cloudera Enterprise