Cloudera Manager 7.9.5

# Cloudera Manager Overview

**Date published: 2020-11-30**
**Date modified: 2023-01-25**

**https://docs.cloudera.com/**

# Legal Notice

© Cloudera Inc. 2025. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

# Contents

**Overview** ..... 5

**Cloudera Manager Terminology** ..... 8

**Cloudera Manager Architecture** ..... 10

**State Management** ..... 12

**Cloudera Manager Admin Console** ..... 13

Cloudera Manager Admin Console Home Page ..... 15

Automatic Logout ..... 19

**Software Distribution Management** ..... 21

**Process Management** ..... 22

**Host Management** ..... 22

**Cloudera Manager Agents** ..... 23

**Resource Management** ..... 23

**User Management** ..... 24

**Security Management** ..... 25

**Monitoring a Cluster Using Cloudera Manager** ..... 26

**Cloudera Management Service** ..... 27

**Cluster Configuration Overview** ..... 28

**Server and Client Configuration** ..... 29

**Cloudera Manager API** ..... 30

**Virtual Private Clusters and Cloudera SDX** ..... 30

Advantages of Separating Compute and Data Resources ..... 30
Architecture ..... 31
Performance Trade Offs ..... 33
Compatibility Considerations for Virtual Private Clusters ..... 33
CDH and Cloudera Runtime Version Compatibility ..... 33
Licensing Requirements ..... 34
Components ..... 34
Compute Cluster Services ..... 35
Cloudera Navigator Support ..... 35
Cloudera Manager Permissions ..... 35
Security ..... 35
Networking ..... 36
Networking Considerations for Virtual Private Clusters ..... 36
Minimum Network Performance Requirements ..... 36
Sizing and designing the Network Topology ..... 37

# Overview

Overview of Cloudera Manager functionality.

Cloudera Manager is an end-to-end application for managing clusters. With Cloudera Manager, you can easily deploy and centrally operate the complete Cloudera Runtime stack and other managed services. The application automates the installation and upgrade processes and gives you a cluster-wide, real-time view of hosts and running services. The Cloudera Manager Admin Console provides a single, central console where you can make configuration changes across your cluster and incorporates a full range of reporting and diagnostic tools to help you optimize performance and utilization. Cloudera Manager also manages security and encryption functionality. This overview introduces the basic concepts, structure, and functions of Cloudera Manager.

A single instance of Cloudera Manager can manage multiple clusters, including older versions of Cloudera Runtime and CDH.

**Note:** Not all combinations of Cloudera Manager, Cloudera Runtime, and CDP Private Cloud Data Services are supported. Ensure that the version of Cloudera Manager you are using supports the version of Cloudera Runtime and CDP Private Cloud Data Services you have selected. For more information, see the Cloudera Support Matrix,

The versions of Cloudera Runtime, CDP Private Cloud Data Services, and CDH clusters that can be managed by Cloudera Manager are limited to the following:

### Table 1: Cloudera Manager support for Cloudera Runtime, CDH, and CDP Private Cloud Data Services

| Cloudera Manager Version | Supported CDH/Runtime versions | Supported CDP Private Cloud Data Services versions |
|---|---|---|
| 7.9.5 | • Cloudera Runtime 7.1.8<br>• Cloudera Runtime 7.1.7 SP1<br>• Cloudera Runtime 7.1.6<br>• Cloudera Runtime 7.1.5<br>• Cloudera Runtime 7.1.4<br>• Cloudera Runtime 7.1.3<br>• Cloudera Runtime 7.1.2<br>• Cloudera Runtime 7.1.1<br>• Cloudera Runtime 7.0.3<br>• CDH 6.3<br>• CDH 6.2<br>• CDH 6.1<br>• CDH 6.0 | 1.5.0<br>Only supported with Cloudera Runtime 7.1.7 SP1 and 7.1.8. |
| 7.8.1 | • Cloudera Runtime 7.1.8<br>• Cloudera Runtime 7.1.7 SP1<br>• Cloudera Runtime 7.1.6<br>• Cloudera Runtime 7.1.5<br>• Cloudera Runtime 7.1.4<br>• Cloudera Runtime 7.1.3<br>• Cloudera Runtime 7.1.2<br>• Cloudera Runtime 7.1.1<br>• Cloudera Runtime 7.0.3<br>• CDH 6.3<br>• CDH 6.2<br>• CDH 6.1<br>• CDH 6.0 | 1.4.1<br>Only supported with Cloudera Runtime 7.1.7 SP1 and 7.1.8. |

| Cloudera Manager Version | Supported CDH/Runtime versions | Supported CDP Private Cloud Data Services versions |
|---|---|---|
| 7.7.3 / 7.7.3-CHF1<br><br>Cloudera Manager 7.7.3 should only be used when you need to use Python 3.8 for the Cloudera Manager agents. You must install Python 3.8 on all hosts before installing or upgrading to Cloudera Manager 7.7.3. Cloudera Manager 7.7.3 is only supported with RHEL 8.4 and 8.6. Cloudera Manager 7.7.3-CHF1 adds support for RHEL 7.9. See the CDP Private Cloud Base Installation Guide for more information. | • Cloudera Runtime 7.1.8 | None |
| 7.7.1 | • Cloudera Runtime 7.1.8<br>• Cloudera Runtime 7.1.7 SP1<br>• Cloudera Runtime 7.1.7<br>• Cloudera Runtime 7.1.6<br>• Cloudera Runtime 7.1.5<br>• Cloudera Runtime 7.1.4<br>• Cloudera Runtime 7.1.3<br>• Cloudera Runtime 7.1.2<br>• Cloudera Runtime 7.1.1<br>• Cloudera Runtime 7.0.3<br>• CDH 6.3<br>• CDH 6.2<br>• CDH 6.1<br>• CDH 6.0 | None |
| 7.6.5 | • Supported with Cloudera Runtime 7.1.6 , 7.1.7, and 7.1.7 SP1 only when CDP Private Cloud Data Services is deployed. | 1.3.1, 1.3.2, 1.3.3. 1.3.4 are supported with Cloudera Runtime 7.1.6 , 7.1.7.<br><br>1.4.0 is supported with Cloudera Runtime 7.1.7 SP1 only. |
| 7.6.1<br><br>**Important:** Do not upgrade to Cloudera Manager 7.6.1 if you are running CDP Private Cloud Data Services in your deployment. | • Cloudera Runtime 7.1.7 SP1<br>• Cloudera Runtime 7.1.7<br>• Cloudera Runtime 7.1.6<br>• Cloudera Runtime 7.1.5<br>• Cloudera Runtime 7.1.4<br>• Cloudera Runtime 7.1.3<br>• Cloudera Runtime 7.1.2<br>• Cloudera Runtime 7.1.1<br>• Cloudera Runtime 7.0.3<br>• CDH 6.3<br>• CDH 6.2<br>• CDH 6.1<br>• CDH 6.0<br>• CDH 5.13 - 5.16 | None |
| 7.5.5 | • **Note:** Cloudera Manager 7.5.5 is not compatible with the Spark 3 CDS parcel.<br>• Cloudera Runtime 7.1.7<br>• Cloudera Runtime 7.1.6 | 1.3.1, 1.3.2, 1.3.3. 1.3.4<br><br>Supported with Cloudera Runtime 7.1.6 and 7.1.7 only |
| 7.5.4 | • **Note:** Cloudera Manager 7.5.4 is not compatible with the Spark 3 CDS parcel.<br>• Cloudera Runtime 7.1.7<br>• Cloudera Runtime 7.1.6 | 1.3.1, 1.3.2, 1.3.3<br><br>Supported with Cloudera Runtime 7.1.6 and 7.1.7 only |

| Cloudera Manager Version | Supported CDH/Runtime versions | Supported CDP Private Cloud Data Services versions |
|---|---|---|
| 7.5.1 | • **Note:** Cloudera Manager 7.5.1 is not compatible with the Spark 3 CDS parcel.<br>•<br>• Cloudera Runtime 7.1.7<br>• Cloudera Runtime 7.1.6 | 1.3.1<br>Supported with Cloudera Runtime 7.1.6 and 7.1.7 only |
| 7.4.4 | • Cloudera Runtime 7.1.7<br>• Cloudera Runtime 7.1.6<br>• Cloudera Runtime 7.1.5<br>• Cloudera Runtime 7.1.4<br>• Cloudera Runtime 7.1.3<br>• Cloudera Runtime 7.1.2<br>• Cloudera Runtime 7.1.1<br>• Cloudera Runtime 7.0.3<br>• CDH 6.3<br>• CDH 6.2<br>• CDH 6.1<br>• CDH 6.0<br>• CDH 5.13 - 5.16 | None |
| 7.3.1 | • Cloudera Runtime 7.1.6<br>• Cloudera Runtime 7.1.5<br>• Cloudera Runtime 7.1.4<br>• Cloudera Runtime 7.1.3<br>• Cloudera Runtime 7.1.2<br>• Cloudera Runtime 7.1.1<br>• Cloudera Runtime 7.0.3<br>• CDH 6.3<br>• CDH 6.2<br>• CDH 6.1<br>• CDH 6.0<br>• CDH 5.13 - 5.16 | None |
| 7.2.4 | • Cloudera Runtime 7.1.5<br>• Cloudera Runtime 7.1.4<br>• Cloudera Runtime 7.1.3<br>• Cloudera Runtime 7.1.2<br>• Cloudera Runtime 7.1.1<br>• Cloudera Runtime 7.0.3<br>• CDH 6.3<br>• CDH 6.2<br>• CDH 6.1<br>• CDH 6.0<br>• CDH 5.13 - 5.16 | 1.2<br>Supported with Cloudera Runtime 7.1.5 only |
| 7.1.4 | • Cloudera Runtime 7.1.4<br>• Cloudera Runtime 7.1.3<br>• Cloudera Runtime 7.1.2<br>• Cloudera Runtime 7.1.1<br>• Cloudera Runtime 7.0.3<br>• CDH 6.3<br>• CDH 6.2<br>• CDH 6.1<br>• CDH 6.0<br>• CDH 5.13 - 5.16 | |

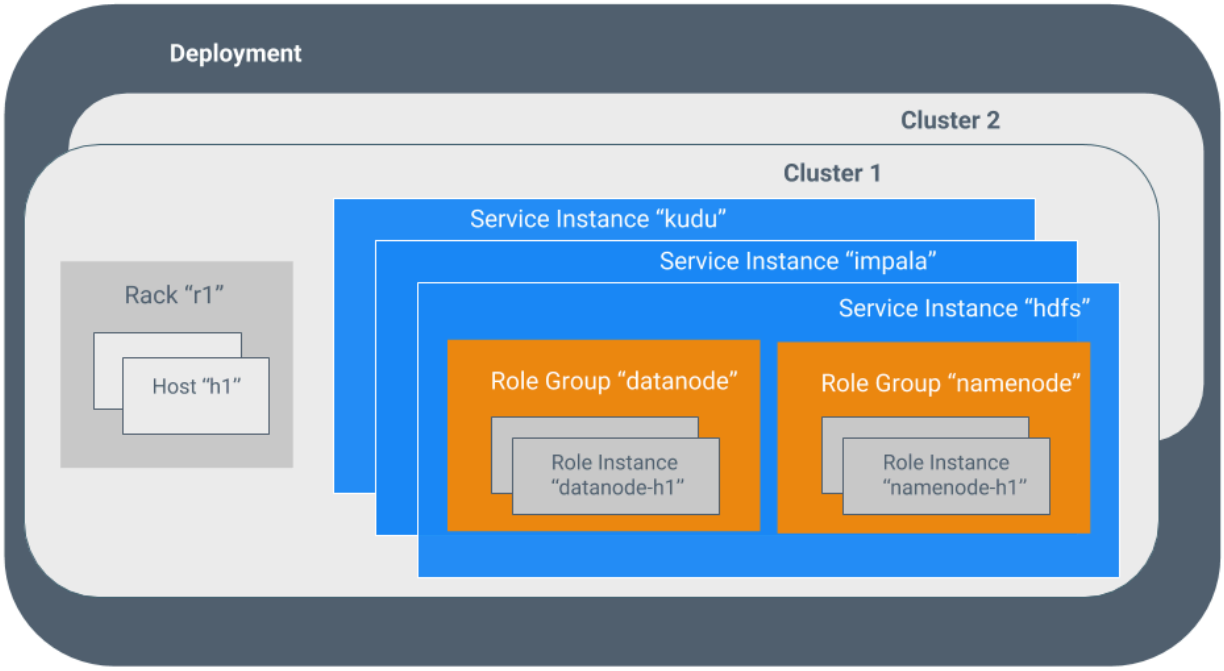
| Cloudera Manager Version | Supported CDH/Runtime versions | Supported CDP Private Cloud Data Services versions |
|---|---|---|
| 7.1.3 | • Cloudera Runtime 7.1.3<br>• Cloudera Runtime 7.1.2<br>• Cloudera Runtime 7.1.1<br>• Cloudera Runtime 7.0.3<br>• CDH 6.3<br>• CDH 6.2<br>• CDH 6.1<br>• CDH 6.0<br>• CDH 5.13 - 5.16 | 1.1 |
| 7.1.2 | • Cloudera Runtime 7.1.2<br>• Cloudera Runtime 7.1.1<br>• Cloudera Runtime 7.0.3<br>• CDH 6.3<br>• CDH 6.2<br>• CDH 6.1<br>• CDH 6.0<br>• CDH 5.13 - 5.16 | 1.0 |
| 7.1.1 | • Cloudera Runtime 7.1.1<br>• Cloudera Runtime 7.0.3<br>• CDH 6.3<br>• CDH 6.2<br>• CDH 6.1<br>• CDH 6.0<br>• CDH 5.13 - 5.16 | |
| 7.0.3 | • Cloudera Runtime 7.0.3 | |

# Cloudera Manager Terminology

Terminology used in Cloudera Manager.

To effectively use Cloudera Manager, you should first understand its terminology.

The relationship between the terms is illustrated below and their definitions follow:




Some of the terms, such as cluster and service, are used without further explanation. Other terms, such as role group, gateway, host template, and parcel are explained in the below sections.

Sometimes the terms *service* and *role* are used to refer to both types and instances, which can be confusing. Cloudera Manager and this section sometimes use the same term for type and instance. For example, the Cloudera Manager Admin Console Home Status tab and the Clusters *CLUSTERNAME* menu list service instances. This is similar to the practice in programming languages where the term "string" might indicate a type (java.lang.String) or an instance of that type ("hi there"). Where it is necessary to distinguish between types and instances, the word "type" is appended to indicate a type and the word "instance" is appended to explicitly indicate an instance.

### deployment

A configuration of Cloudera Manager and all the clusters it manages.

### dynamic resource pool

In Cloudera Manager, a named configuration of resources and a policy for scheduling the resources among YARN applications or Impala queries running in the pool.

### cluster

- A set of computers or racks of computers that contains an HDFS filesystem and runs MapReduce and other processes on that data.
- In Cloudera Manager, a logical entity that contains a set of hosts, a single version of Cloudera Runtime installed on the hosts, and the service and role instances running on the hosts. A host can belong to only one cluster. Cloudera Manager can manage multiple clusters, however each cluster can only be associated with a single Cloudera Manager Server.

### host

In Cloudera Manager, a physical or virtual machine that runs role instances. A host can belong to only one cluster.

### rack

In Cloudera Manager, a physical entity that contains a set of physical hosts typically served by the same switch.

### service

- A Linux command that runs a System V init script in /etc/init.d/ in as predictable an environment as possible, removing most environment variables and setting the current working directory to /.
- A category of managed functionality in Cloudera Manager, which may be distributed or not, running in a cluster. Sometimes referred to as a service type. For example: Hive, HBase, HDFS, YARN, and Spark.

### service instance

In Cloudera Manager, an instance of a service running on a cluster. For example: "HDFS-1" and "yarn". A service instance spans many role instances.

### role

In Cloudera Manager, a category of functionality within a service. For example, the HDFS service has the following roles: NameNode, SecondaryNameNode, DataNode, and Balancer. Sometimes referred to as a role type.

### role instance

In Cloudera Manager, an instance of a role running on a host. It typically maps to a Unix process. For example: "NameNode-h1" and "DataNode-h1".

### role group

In Cloudera Manager, a set of configuration properties for a set of role instances.

### host template

A set of role groups in Cloudera Manager. When a template is applied to a host, a role instance from each role group is created and assigned to that host.

### gateway

A type of role that typically provides client access to specific cluster services. For example, HDFS, Hive, Kafka, MapReduce, Solr, and Spark each have gateway roles to provide access for their clients to their respective services. Gateway roles do not always have "gateway" in their names, nor are they exclusively for client access. For example, Hue Kerberos Ticket Renewer is a gateway role that proxies tickets from Kerberos.

The node supporting one or more gateway roles is sometimes referred to as the *gateway node* or *edge node*, with the notion of "edge" common in network or cloud environments. In terms of the Cloudera cluster, the gateway nodes in the cluster receive the appropriate client configuration files when Deploy Client Configuration is selected from the Actions menu in Cloudera Manager Admin Console.

### parcel

A binary distribution format that contains compiled code and meta-information such as a package description, version, and dependencies.

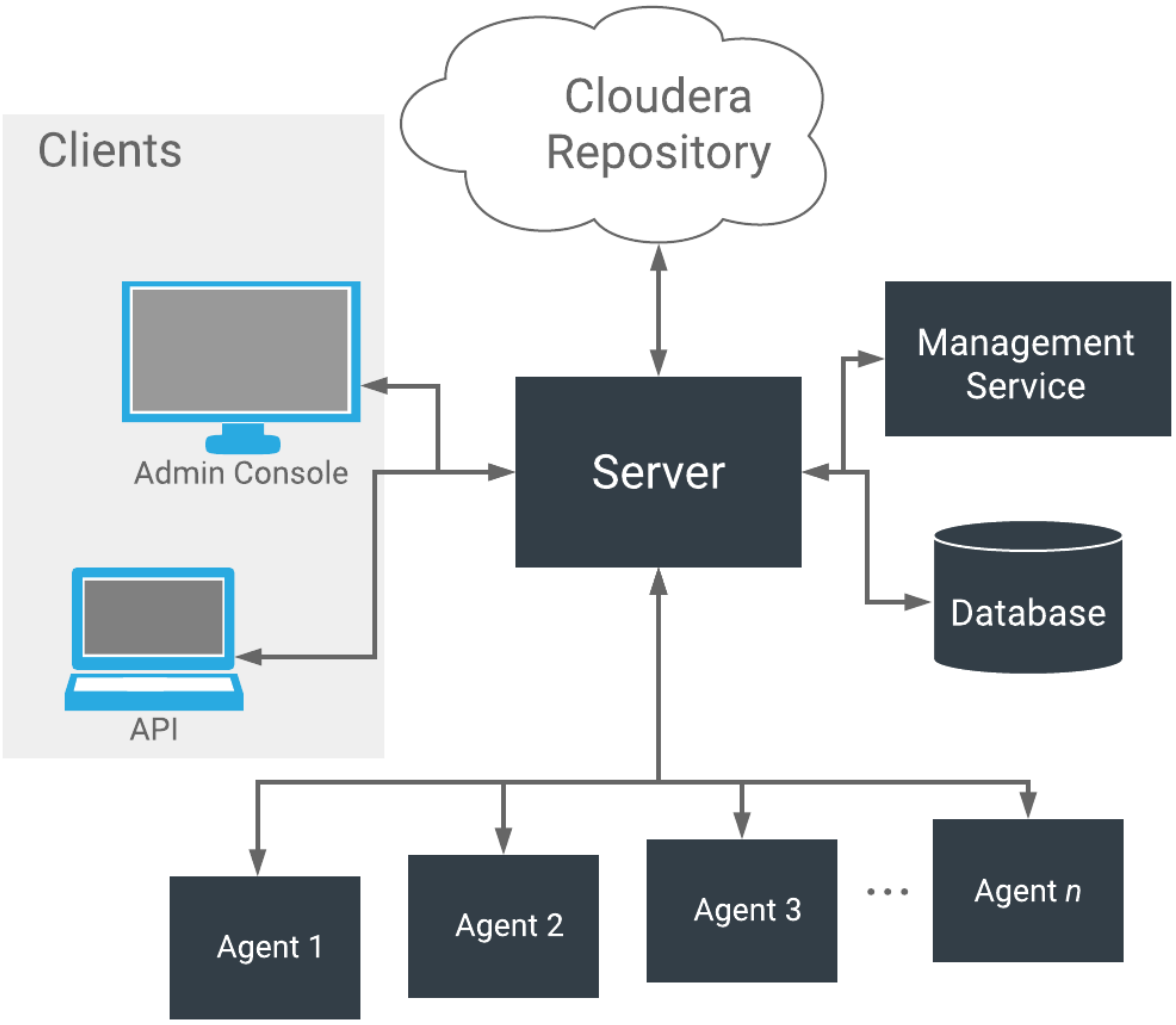
### static service pool

In Cloudera Manager, a static partitioning of total cluster resources—CPU, memory, and I/O weight—across a set of services.

# Cloudera Manager Architecture

Description of the components that comprise Cloudera Manager.

As depicted below, the heart of Cloudera Manager is the Cloudera Manager Server. The Server hosts the Cloudera Manager Admin Console, the Cloudera Manager API, and the application logic, and is responsible for installing software, configuring, starting, and stopping services, and managing the cluster on which the services run.




The Cloudera Manager Server works with several other components:

- Agent - installed on every host. The agent is responsible for starting and stopping processes, unpacking configurations, triggering installations, and monitoring the host.
- Management Service - a service consisting of a set of roles that perform various monitoring, alerting, and reporting functions.
- Database - stores configuration and monitoring information. Typically, multiple logical databases run across one or more database servers. For example, the Cloudera Manager Server and the monitoring roles use different logical databases.
- Cloudera Repository - repository of software for distribution by Cloudera Manager.
- Clients - are the interfaces for interacting with the server:
  - Cloudera Manager Admin Console - Web-based user interface that administrators use to manage clusters and Cloudera Manager.
  - Cloudera Manager API - API developers use to create custom Cloudera Manager applications.

### Heartbeating

Heartbeats are a primary communication mechanism in Cloudera Manager. By default Agents send heartbeats every 15 seconds to the Cloudera Manager Server. However, to reduce user latency the frequency is increased when state is changing.

During the heartbeat exchange, the Agent notifies the Cloudera Manager Server of its activities. In turn the Cloudera Manager Server responds with the actions the Agent should be performing. Both the Agent and the Cloudera Manager Server end up doing some reconciliation. For example, if you start a service, the Agent attempts to start the relevant processes; if a process fails to start, the Cloudera Manager Server marks the start command as having failed.

### Related Information

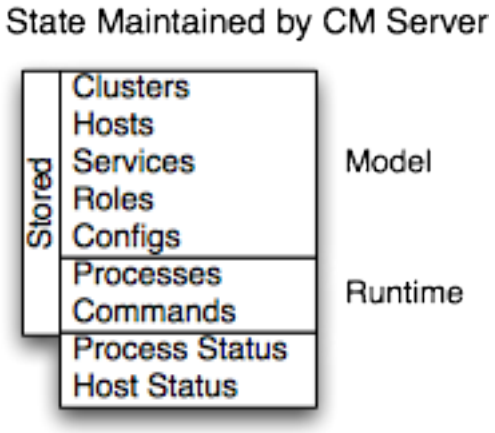
Cloudera Management Service

Cloudera Manager Admin Console

Cloudera Manager API

# State Management

The Cloudera Manager Server maintains the state of the cluster. This state can be divided into two categories: "model" and "runtime", both of which are stored in the Cloudera Manager Server database.




Cloudera Manager models clusters and managed services: their roles, configurations, and inter-dependencies. Model state captures what is supposed to run where, and with what configurations. For example, model state captures the fact that a cluster contains 17 hosts, each of which is supposed to run a DataNode. You interact with the model through the Cloudera Manager Admin Console configuration screens and API and operations such as "Add Service".

Runtime state is what processes are running where, and what commands (for example, rebalance HDFS or run a Backup/Disaster Recovery schedule or rolling restart or stop) are currently running. The runtime state includes the exact configuration files needed to run a process. When you select Start in the Cloudera Manager Admin Console, the server gathers up all the configuration for the relevant services and roles, validates it, generates the configuration files, and stores them in the database.

When you update a configuration (for example, the Hue Server web port), you have updated the model state. However, if Hue is running while you do this, it is still using the old port. When this kind of mismatch occurs, the role is marked as having an "outdated configuration". To resynchronize, you restart the role (which triggers the configuration re-generation and process restart).

While Cloudera Manager models all of the reasonable configurations, some cases inevitably require special handling. To allow you to workaround, for example, a bug or to explore unsupported options, Cloudera Manager supports an "advanced configuration snippet" mechanism that lets you add properties directly to the configuration files.

**Related Information**

Advanced Configuration Snippets

# Cloudera Manager Admin Console

Cloudera Manager Admin Console is the web-based interface that you use to configure, manage, and monitor Cloudera Runtime.

The Cloudera Manager Admin Console side navigation bar provides the following tabs and menus:

**Note:** Depending on the user role used to log in, some items may not appear in the Cloudera Manager Admin Console.

- Search - Supports searching for services, roles, hosts, configuration properties, and commands. You can enter a partial string and a drop-down list with up to sixteen entities that match will display.
- Clusters*CLUSTER_NAME*
  - Services - Display individual services, and the Cloudera Management Service. In these pages you can:
    - View the status and other details of a service instance or the role instances associated with the service
    - Make configuration changes to a service instance, a role, or a specific role instance
    - Add and delete a service or role
    - Stop, start, or restart a service or role.
    - View the commands that have been run for a service or a role
    - View an audit event history
    - Deploy and download client configurations
    - Decommission and recommission role instances
    - Enter or exit maintenance mode
    - Perform actions unique to a specific type of service. For example:
      - Enable HDFS high availability or NameNode federation
      - Run the HDFS Balancer
      - Create HBase, Hive, and Sqoop directories
  - Cloudera Manager Management Service - Manage and monitor the Cloudera Manager Management Service. This includes the following roles: Activity Monitor, Alert Publisher, Event Server, Host Monitor, Navigator Audit Server, Navigator Metadata Server, Reports Manager, and Service Monitor.
  - Hosts - Displays the hosts in the cluster.
  - Reports - Create reports about the HDFS, MapReduce, YARN, and Impala usage and browse HDFS files, and manage quotas for HDFS directories.
  - Utilization Report - Opens the Cluster Utilization Report. displays aggregated utilization information for YARN and Impala jobs.
  - *MAPREDUCE_SERVICE_NAME* Jobs - Query information about MapReduce jobs running on your cluster.
  - *YARN_SERVICE_NAME* Applications - Query information about YARN applications running on your cluster.
  - *IMPALA_SERVICE_NAME* Queries - Query information about Impala queries running on your cluster.
  - Dynamic Resource Pools - Manage dynamic allocation of cluster resources to YARN and Impala services by specifying the relative weights of named pools.
  - Static Service Pools - Manage static allocation of cluster resources to HBase, HDFS, Impala, MapReduce, and YARN services.

- Hosts - Display the hosts managed by Cloudera Manager.
  - All Hosts - Displays a list of manage hosts in the cluster.
  - Add Hosts - Launches the Add Hosts wizard.
  - Parcels - Displays parcels available in the cluster and allows you to download, distribute, and activate new parcels.
  - Hosts Configuration - Opens the Hosts Configuration page where you can configure hosts and specify overrides for globally-configured properties for one or more hosts.
  - Roles - Displays the roles deployed on each host.
  - Host Templates - Create and manage Host Templates, which define sets of role groups that can be used to easily expand a cluster.
  - Disks Overview - Displays the status of all disks in the cluster.

  In this page you can:

  - View the status and a variety of detail metrics about individual hosts
  - Make configuration changes for host monitoring
  - View all the processes running on a host
  - Run the Host Inspector
  - Add and delete hosts
  - Create and manage host templates
  - Manage parcels
  - Decommission and recommission hosts
  - Make rack assignments
  - Run the host upgrade wizard
- Diagnostics - Review logs, events, and alerts to diagnose problems. The subpages are:
  - Events - Search for and displaying events and alerts that have occurred.
  - Logs - Search logs by service, role, host, and search phrase as well as log level (severity).
  - Server Log -Display the Cloudera Manager Server log.
- Audits - Query and filter audit events across clusters, including logins, across clusters.
- Charts - Query for metrics of interest, display them as charts, and display personalized chart dashboards.
- Replication - Manage replication schedules and snapshot policies.
- Administration - Administer Cloudera Manager. The subpages are:
  - Settings - Configure Cloudera Manager.
  - Alerts - Display when alerts will be generated, configure alert recipients, and send test alert email.
  - Users & Roles - Manage Cloudera Manager users and their assigned roles, and sessions.
  - Security - Generate Kerberos credentials and inspect hosts.
  - License - Manage Cloudera licenses.
  - Language - Set the language used for the content of activity events, health events, and alert email messages.
  - External Accounts - Configure connectivity from cloud services to Cloudera Manager.
- Parcels Opens the Parcels page where you can view the status of installed and available parcels.
- Recent Commands Indicator - displays the status commands currently or recently running for all services or roles.

- Support - Displays various support actions. The subcommands are:
  - Send Diagnostic Data - Sends data to Cloudera Support to support troubleshooting.
  - Support Portal (Cloudera Enterprise) - Displays the Cloudera Support portal.
  - Scheduled Diagnostics: Weekly - Configure the frequency of automatically collecting diagnostic data and sending to Cloudera support.
  - The following links open the latest documentation on the Cloudera web site:
    - Help
    - Installation Guide
    - API Documentation
    - API Explorer (Cloudera Manager Swagger interface)
    - Release Notes
  - About - Version number and build details of Cloudera Manager and the current date and time stamp of the Cloudera Manager server.
- Logged-in User Menu - The currently logged-in user. The subcommands are:
  - My Profile - Displays the role and login information for the current user.
  - Change Password - Change the password of the currently logged in user.
  - Logout

## Cloudera Manager Admin Console Home Page

When you start the Cloudera Manager Admin Console, the HomeStatus tab displays. You can also go to the HomeStatus tab by clicking the Cloudera Manager logo in the top navigation bar.

The **Status** tab has two potential views: Table View and Classic View. The Classic View contains a set of charts for the selected cluster, while the Table View separates regular clusters, compute clusters, and other services into summary tables. You can use the Switch to Table View and Switch to Classic View links on each view to switch between the two views. Cloudera Manager remembers which view you select and remains in that view.

**Figure 1: Cloudera Manager Admin Console: Classic View**

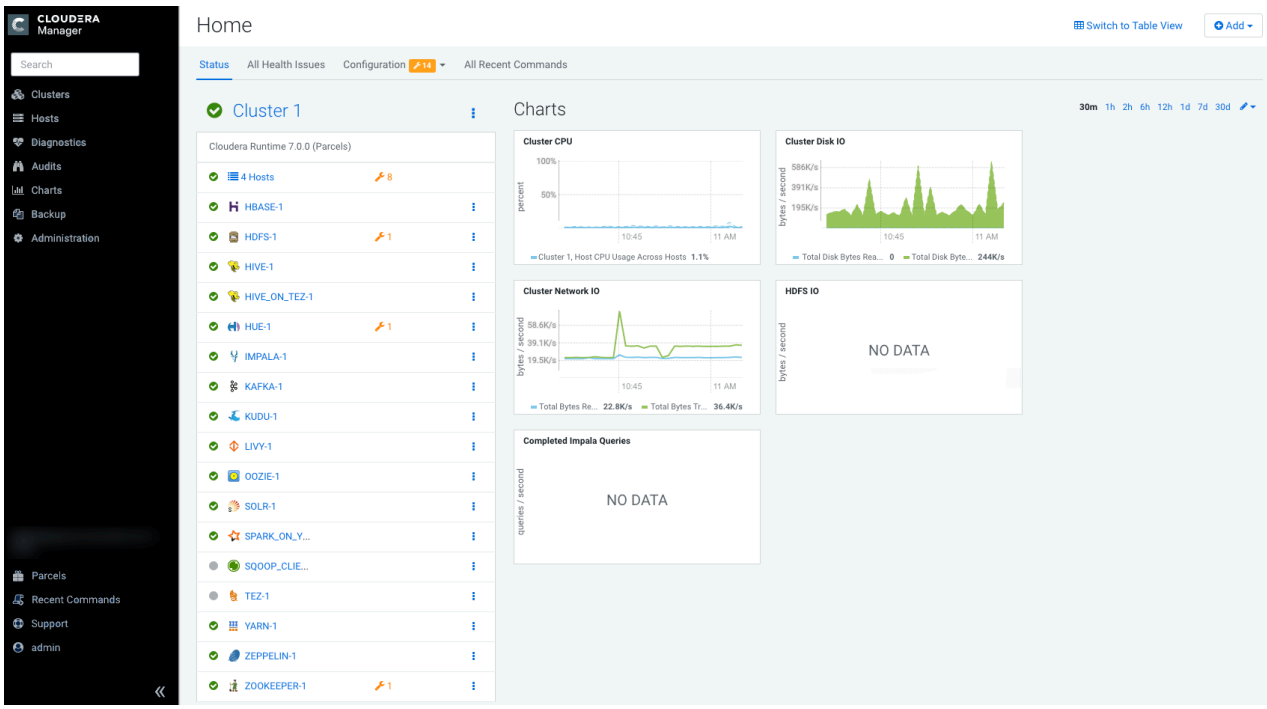



**Figure 2: Cloudera Manager Admin Console**

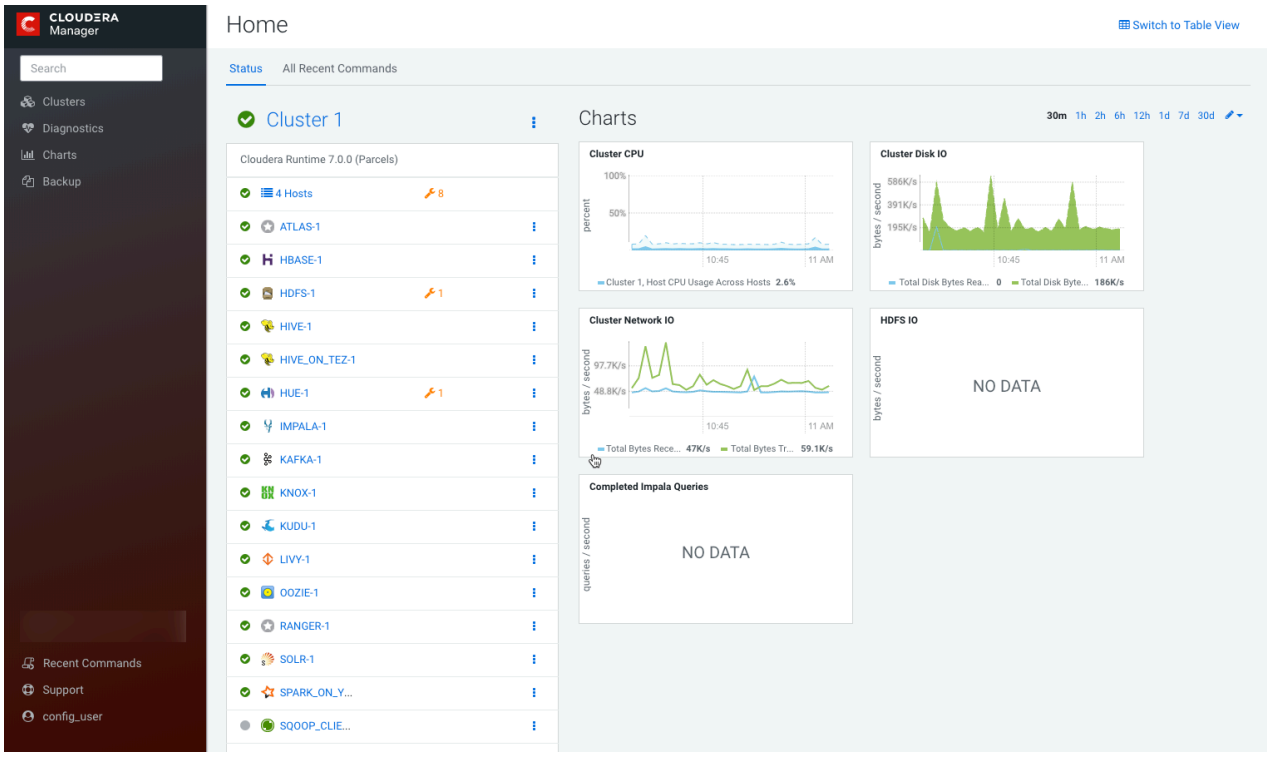



**Figure 3: Cloudera Manager Admin Console: Table View**

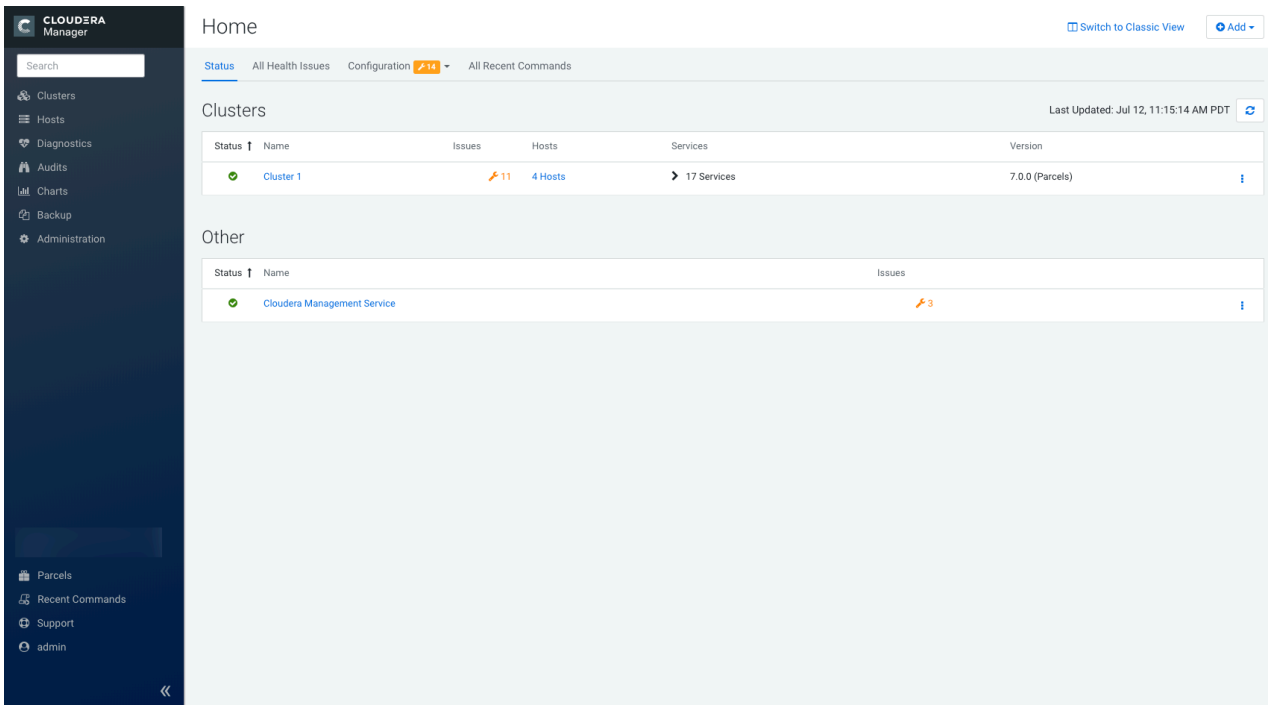
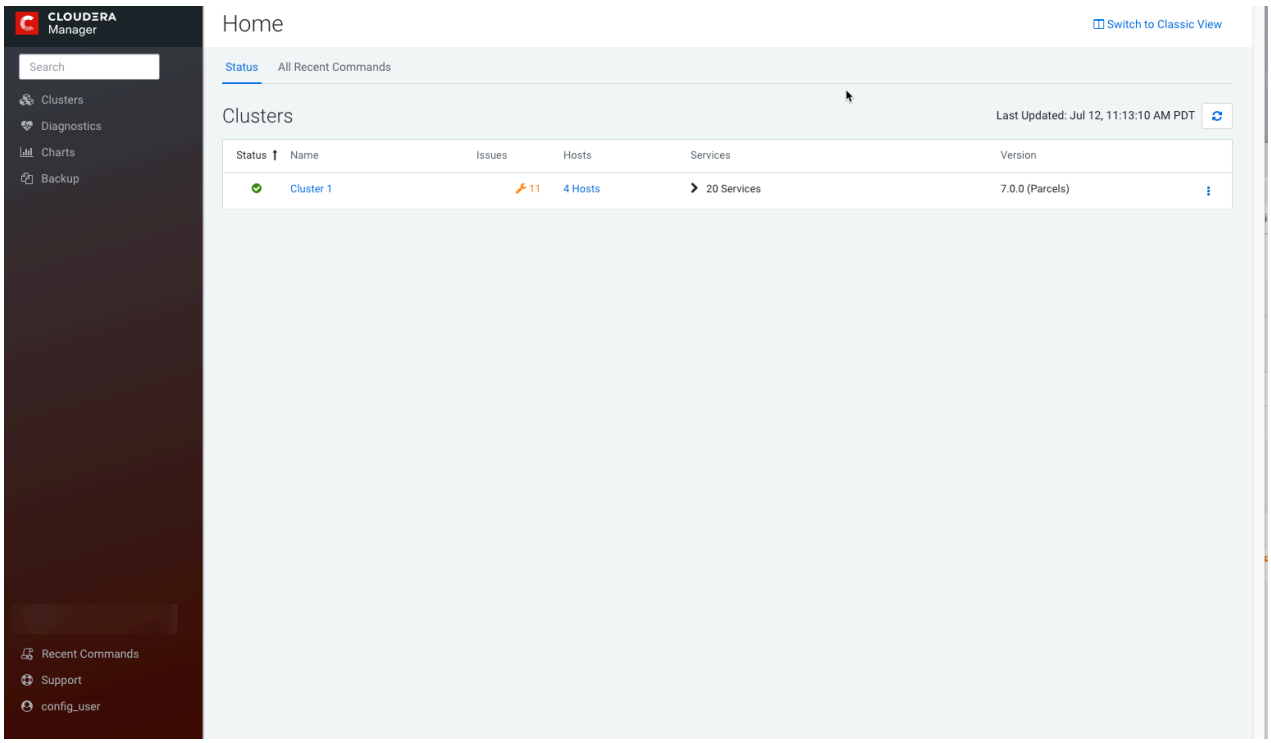



**Figure 4: Cloudera Manager Admin Console: Table View**




## Status

The Status tab contains:

- Clusters - The clusters being managed by Cloudera Manager. Each cluster is displayed either in summary form or in full form depending on the configuration of the AdministrationSettingsOtherMaximum Cluster Count Shown In

Full property. When the number of clusters exceeds the value of the property, only cluster summary information displays.

- Summary Form - A list of links to cluster status pages. Click Customize to jump to the AdministrationSettingsOtherMaximum Cluster Count Shown In Full property.
- Full Form - A separate section for each cluster containing a link to the cluster status page and a table containing links to the Hosts page and the status pages of the services running in the cluster.

Each service row in the table has a menu of actions that you select by clicking the Actions Menu ( ) and can contain one or more of the following indicators:

| Indicator | Meaning | Description |
|---|---|---|
| 2 | Health issue | Indicates that the service has at least one health issue. The indicator shows the number of health issues at the highest severity level. If there are Bad health test results, the indicator is red. If there are no Bad health test results, but Concerning test results exist, then the indicator is yellow. No indicator is shown if there are no Bad or Concerning health test results.<br>**Important:** If there is one Bad health test result and two Concerning health results, there will be three health issues, but the number will be one.<br>Click the indicator to display the Health Issues pop-up dialog box.<br>By default only Bad health test results are shown in the dialog box. To display Concerning health test results, click the Also show *N* concerning issue(s) link.Click the link to display the Status page containing with details about the health test result. |
| 4 | Configuration issue | Indicates that the service has at least one configuration issue. The indicator shows the number of configuration issues at the highest severity level. If there are configuration errors, the indicator is red. If there are no errors but configuration warnings exist, then the indicator is yellow. No indicator is shown if there are no configuration notifications.<br>**Important:** If there is one configuration error and two configuration warnings, there will be three configuration issues, but the number will be one.<br>Click the indicator to display the Configuration Issues pop-up dialog box.<br>By default only notifications at the Error severity level are listed, grouped by service name are shown in the dialog box. To display Warning notifications, click the Also show *N* warning(s) link.Click the message associated with an error or warning to be taken to the configuration property for which the notification has been issued where you can address the issue. |
| Restart Needed<br>Refresh Needed | Configuration modified | Indicates that at least one of a service's roles is running with a configuration that does not match the current configuration settings in Cloudera Manager.<br>Click the indicator to display the Stale Configurations page.To bring the cluster up-to-date, click the Refresh or Restart button on the Stale Configurations page. You can also Refresh or Restart, the cluster or restart the services. |
| | Client configuration redeployment required | Indicates that the client configuration for a service should be redeployed.<br>Click the indicator to display the Stale Configurations page. To bring the cluster up-to-date, click the Deploy Client Configuration button on the Stale Configurations page or manually redeploy the client configuration. |

- Cloudera Management Service - A table containing a link to the Cloudera Manager Service. The Cloudera Manager Service has a menu of actions that you select by clicking .

- Charts - A set of charts (dashboards) that summarize resource utilization (IO, CPU usage) and processing metrics.

  Click a line, stack area, scatter, or bar chart to expand it into a full-page view with a legend for the individual charted entities as well more fine-grained axes divisions.

  By default the time scale of a dashboard is 30 minutes. To change the time scale, click a duration link

  
  at the top-right of the dashboard.

  To set the dashboard type, click and select one of the following:

  - Custom - displays a custom dashboard.
  - Default - displays a default dashboard.
  - Reset - resets the custom dashboard to the predefined set of charts, discarding any customizations.

### All Health Issues

Displays all health issues by cluster. The number badge has the same semantics as the per service health issues reported on the Status tab.

- By default only Bad health test results are shown in the dialog box. To display Concerning health test results, click the Also show *N* concerning issue(s) link.
- To group the health test results by entity or health test, click the buttons on the Organize by Entity/Organize by Health Test switch.
- Click the link to display the Status page containing with details about the health test result.

### All Configuration Issues

Displays all configuration issues by cluster. The number badge has the same semantics as the per service configuration issues reported on the Status tab. By default only notifications at the Error severity level are listed, grouped by service name are shown in the dialog box. To display Warning notifications, click the Also show *N* warning(s) link. Click the message associated with an error or warning to be taken to the configuration property for which the notification has been issued where you can address the issue.

### All Recent Commands

Displays all commands run recently across the clusters. A badge indicates how many recent commands are still running. Click the command link to display details about the command and child commands.

### Displaying the Cloudera Manager Server Version and Server Time

To display the version, build number, and time for the Cloudera Manager Server:

1. Open the Cloudera Manager Admin Console.
2. Select SupportAbout.

### Related Information

Managing Cloudera Runtime Services

Viewing and Running Recent Commands

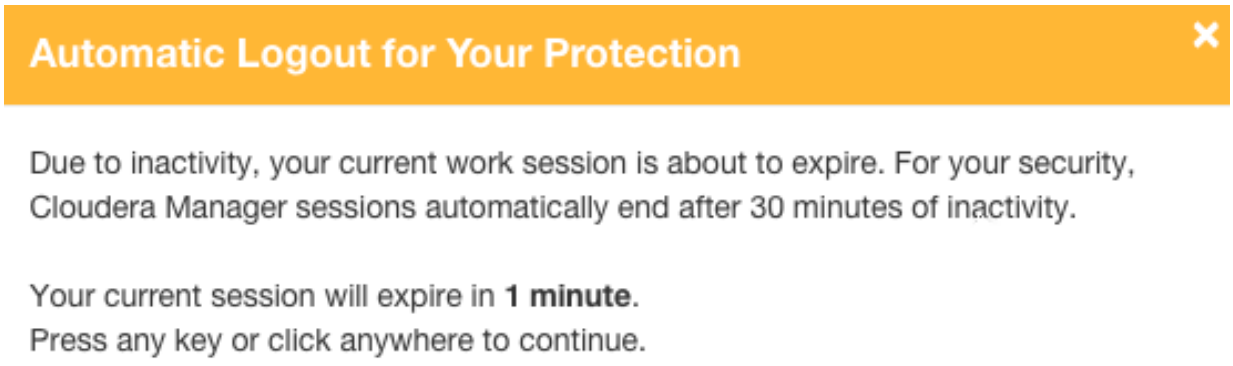
## Automatic Logout

For security purposes, Cloudera Manager automatically logs out a user session after 30 minutes. You can change this session logout period.

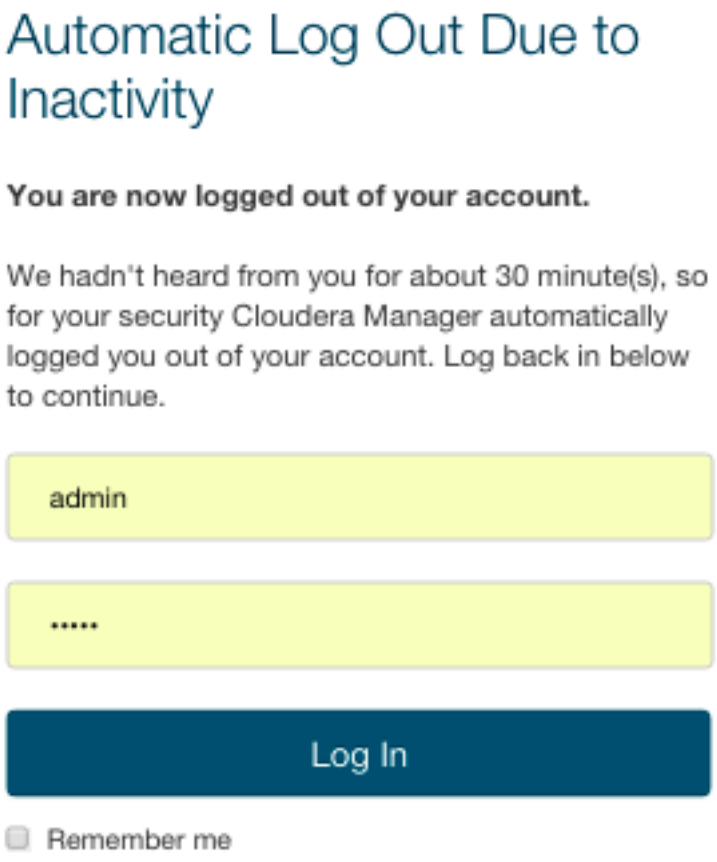
### Procedure

1. Click AdministrationSettings.
2. Click CategorySecurity.
3. Edit the Session Timeout property.
4. Enter a Reason for change, and then click Save Changes to commit the changes.

   When the timeout is one minute from triggering, the user sees the following message:




   If the user does not click the mouse or press a key, the user is logged out of the session and the following message appears:



# Software Distribution Management

A major function of Cloudera Manager is to install and upgrade Cloudera Runtime and other managed services. Cloudera Manager supports two software distribution formats: packages and parcels.

A *package* is a binary distribution format that contains compiled code and meta-information such as a package description, version, and dependencies. Package management systems evaluate this meta-information to allow package searches, perform upgrades to a newer version, and ensure that all dependencies of a package are fulfilled. Cloudera Manager uses the built-in system package manager for each supported OS to install and upgrade Cloudera Manager.

A *parcel* is a binary distribution format containing the program files, along with additional metadata used by Cloudera Manager. Parcels provide the following advantages:

- Parcels are self-contained and installed in a versioned directory, which means that multiple versions of a given parcel can be installed side-by-side. You can then designate one of these installed versions as the active one. With packages, only one package can be installed at a time so there is no distinction between what is installed and what is active.
- Parcels are required for rolling upgrades.
- You can install parcels at any location in the filesystem. They are installed by default in /opt/cloudera/parcels. In contrast, packages are installed in /usr/lib.
- When you install from the Parcels page, Cloudera Manager automatically downloads, distributes, and activates the correct parcel for the operating system running on each host in the cluster. All Cloudera Runtime hosts that make up a logical cluster must run on the same major OS release to be covered by Cloudera Support. Cloudera Manager must run on the same major OS release as at least one of the Cloudera Runtime clusters it manages, to be covered by Cloudera Support. The risk of issues caused by running different minor OS releases is considered lower than the risk of running different major OS releases. Cloudera recommends running the same minor release cross-cluster, because it simplifies issue tracking and supportability..

Because of their unique properties, parcels offer the following advantages over packages:

- Distribution of Cloudera Runtime as a single object - Instead of having a separate package for each component of Cloudera Runtime, parcels are distributed as a single object. This makes it easier to distribute software to a cluster that is not connected to the Internet.
- Internal consistency - All Cloudera Runtime components are matched, eliminating the possibility of installing components from different Cloudera Runtime versions.
- Installation outside of /usr - In some environments, Hadoop administrators do not have privileges to install system packages. With parcels, administrators can install to /opt, or anywhere else.

  **Note:** With parcels, the path to the Cloudera Runtime libraries is /opt/cloudera/parcels/Cloudera Runtime/lib instead of the usual /usr/lib. Do not link /usr/lib/ elements to parcel-deployed paths, because the links can cause scripts that distinguish between the two paths to not work.
- Installation of Cloudera Runtime without sudo - Parcel installation is handled by the Cloudera Manager Agent running as root or another user, so you can install Cloudera Runtime without sudo.
- Decoupled distribution from activation - With side-by-side install capabilities, you can stage a new version of Cloudera Runtime across the cluster before switching to it. This allows the most time-consuming part of an upgrade to be done ahead of time without affecting cluster operations, thereby reducing downtime.
- Rolling upgrades - Using packages requires you to shut down the old process, upgrade the package, and then start the new process. Errors can be difficult to recover from, and upgrading requires extensive integration with the package management system to function seamlessly. With parcels, when a new version is staged side-by-side, you can switch to a new minor version by simply changing which version of Cloudera Runtime is used when restarting each process. You can then perform upgrades with rolling restarts, in which service roles are restarted in the correct order to switch to the new version with minimal service interruption. Your cluster can continue to run on the existing installed components while you stage a new version across your cluster, without impacting your current operations. Major version upgrades (for example, CDH 5 to Cloudera Runtime 7) require full service

restarts because of substantial changes between the versions. Finally, you can upgrade individual parcels or multiple parcels at the same time.
- Upgrade management - Cloudera Manager manages all the steps in a Cloudera Runtime cluster upgrade.
- Additional components - Parcels are not limited to Cloudera Runtime. LZO and add-on service parcels are also available.
- Compatibility with other distribution tools - Cloudera Manager works with other tools you use for download and distribution, such as Puppet. Or, you can download the parcel to Cloudera Manager Server manually if your cluster has no Internet connectivity and then have Cloudera Manager distribute the parcel to the cluster.

For more information, see Overview of Parcels.

# Process Management

Starting and stopping processes using Cloudera Manager.

In a Cloudera Manager managed cluster, you can only start or stop role instance processes using Cloudera Manager. Cloudera Manager uses an open source process management tool called supervisord, that starts processes, takes care of redirecting log files, notifying of process failure, setting the effective user ID of the calling process to the right user, and so on. Cloudera Manager supports automatically restarting a crashed process. It will also flag a role instance with a bad health flag if its process crashes repeatedly right after start up.

Stopping the Cloudera Manager Server and the Cloudera Manager Agents will not bring down your services; any running role instances keep running.

The Agent is started by init.d at start-up. It, in turn, contacts the Cloudera Manager Server and determines which processes should be running. The Agent is monitored as part of Cloudera Manager's host monitoring. If the Agent stops heartbeating, the host is marked as having bad health.

One of the Agent's main responsibilities is to start and stop processes. When the Agent detects a new process from the Server heartbeat, the Agent creates a directory for it in /var/run/cloudera-scm-agent and unpacks the configuration. It then contacts supervisord, which starts the process.

These actions reinforce an important point: a Cloudera Manager process never travels alone. In other words, a process is more than just the arguments to `exec()`—it also includes configuration files, directories that need to be created, and other information.

### Related Information

Supervisor: A Process Control System

# Host Management

Cloudera Manager provides several features to manage the hosts in your clusters

The first time you run Cloudera Manager Admin Console you can search for hosts to add to the cluster and once the hosts are selected you can map the assignment of roles to hosts. Cloudera Manager automatically deploys all software required to participate as a managed host in a cluster: JDK, Cloudera Manager Agent, Impala, Solr, and so on to the hosts.

Once the services are deployed and running, the Hosts area within the Admin Console shows the overall status of the managed hosts in your cluster. The information provided includes the version of Cloudera Runtime running on the host, the cluster to which the host belongs, and the number of roles running on the host. Cloudera Manager provides operations to manage the lifecycle of the participating hosts and to add and delete hosts. The Cloudera Management Service Host Monitor role performs health tests and collects host metrics to allow you to monitor the health and performance of the hosts.

# Cloudera Manager Agents

The Cloudera Manager Agent is a Cloudera Manager component that works with the Cloudera Manager Server to manage the processes that map to role instances.

In a Cloudera Manager managed cluster, you can only start or stop role instance processes using Cloudera Manager. Cloudera Manager uses an open source process management tool called supervisord, that starts processes, takes care of redirecting log files, notifying of process failure, setting the effective user ID of the calling process to the right user, and so on. Cloudera Manager supports automatically restarting a crashed process. It will also flag a role instance with a bad health flag if its process crashes repeatedly right after start up.

The Agent is started by init.d at start-up. It, in turn, contacts the Cloudera Manager Server and determines which processes should be running. The Agent is monitored as part of Cloudera Manager's host monitoring. If the Agent stops heartbeating, the host is marked as having bad health.

One of the Agent's main responsibilities is to start and stop processes. When the Agent detects a new process from the Server heartbeat, the Agent creates a directory for it in /var/run/cloudera-scm-agent and unpacks the configuration. It then contacts supervisord, which starts the process.

### cm_processes

To enable Cloudera Manager to run scripts in subdirectories of /var/run/cloudera-scm-agent, (because /var/run is mounted noexec in many Linux distributions), Cloudera Manager mounts a tmpfs (temporary file storage), named cm_processes, for process subdirectories.

A tmpfs defaults to a max size of 50% of physical RAM but this space is not allocated until its used, and tmpfs is paged out to swap if there is memory pressure.

The lifecycle actions of cmprocesses can be described by the following statements:

- Created when the Agent starts up for the first time with a new supervisord process.
- If it already exists without noexec, reused when the Agent is started using start and not recreated.
- Remounted if Agent is started using clean_restart.
- Unmounting and remounting cleans out the contents (since it is mounted as a tmpfs).
- Unmounted when the host is rebooted.
- Not unmounted when the Agent is stopped.

**Important:**

Cloudera Manager is designed to operate on OS platforms or containers where the root PID is an INIT process. This INIT process should possess the necessary privileges to effectively terminate any zombie processes. This helps to maintain a clean process table, preventing any unwanted clutter from lingering zombie processes that were earlier managed by Cloudera Manager.

### Related Information

Supervisor: A Process Control System

# Resource Management

Resource management helps ensure predictable behavior by defining the impact of different services on cluster resources.

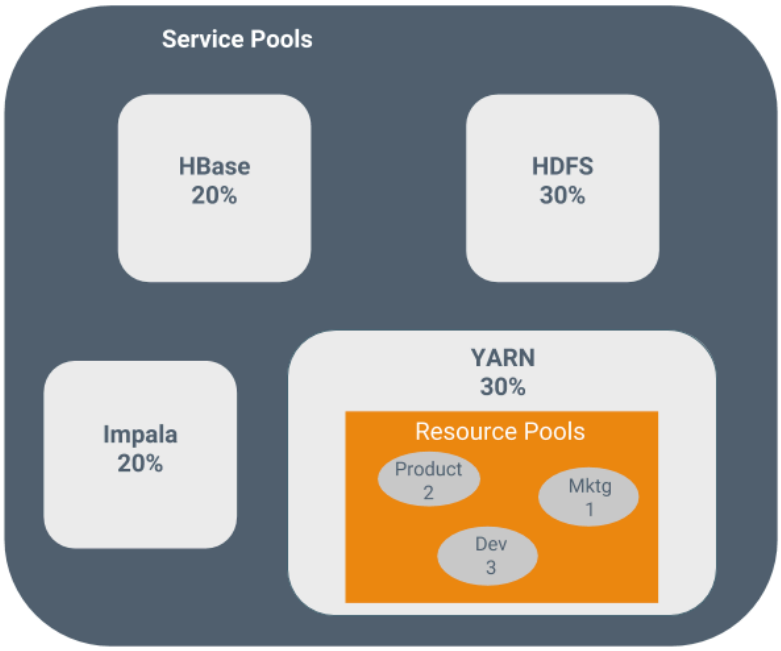
Use resource management to:

- Guarantee completion in a reasonable time frame for critical workloads.
- Support reasonable cluster scheduling between groups of users based on fair allocation of resources per group.

- Prevent users from depriving other users access to the cluster.

Statically allocating resources using cgroups is configurable through a single static service pool wizard. You allocate services as a percentage of total resources, and the wizard configures the cgroups.

For example, the following figure illustrates static pools for HBase, HDFS, Impala, and YARN services that are respectively assigned 20%, 30%, 20%, and 30% of cluster resources.




You can dynamically apportion resources that are statically allocated to YARN and Impala by using dynamic resource pools.

Depending on the version of Cloudera Runtime you are using, dynamic resource pools in Cloudera Manager support the following scenarios:

- YARN - YARN manages the virtual cores, memory, running applications, maximum resources for undeclared children (for parent pools), and scheduling policy for each pool. In the preceding diagram, three dynamic resource pools—Dev, Product, and Mktg with weights 3, 2, and 1 respectively—are defined for YARN. If an application starts and is assigned to the Product pool, and other applications are using the Dev and Mktg pools, the Product resource pool receives 30% x 2/6 (or 10%) of the total cluster resources. If no applications are using the Dev and Mktg pools, the YARN Product pool is allocated 30% of the cluster resources.
- Impala - Impala manages memory for pools running queries and limits the number of running and queued queries in each pool.

# User Management

Access to Cloudera Manager features is controlled by user accounts. A user account identifies how a user is authenticated and determines what privileges are granted to the user.

Cloudera Manager provides several mechanisms for authenticating users. You can configure Cloudera Manager to authenticate users against the Cloudera Manager database or against an external authentication service. The external authentication service can be an LDAP server (Active Directory or an OpenLDAP compatible directory), or you can specify another external service. Cloudera Manager also supports using the Security Assertion Markup Language (SAML) to enable single sign-on.

For information about the privileges associated with each of the Cloudera Manager user roles, see Cloudera Manager User Roles.

# Security Management

Cloudera Manager consolidates security configurations and management for all components in your clusters.

## Authentication

Authentication is a process that requires users and services to prove their identity when trying to access a system resource. Organizations typically manage user identity and authentication through various time-tested technologies, including Lightweight Directory Access Protocol (LDAP) for identity, directory, and other services, such as group management, and Kerberos for authentication.

Cloudera clusters support integration with both of these technologies. For example, organizations with existing LDAP directory services such as Active Directory (included in Microsoft Windows Server as part of its suite of Active Directory Services) can leverage the organization's existing user accounts and group listings instead of creating new accounts throughout the cluster. Using an external system such as Active Directory or OpenLDAP is required to support the user role authorization mechanism implemented in Cloudera Navigator.

For authentication, Cloudera supports integration with MIT Kerberos and with Active Directory. Microsoft Active Directory supports Kerberos for authentication in addition to its identity management and directory functionality, that is, LDAP.

These systems are not mutually exclusive. For example, Microsoft Active Directory is an LDAP directory service that also provides Kerberos authentication services, and Kerberos credentials can be stored and managed in an LDAP directory service. Cloudera Manager Server, cluster nodes, and components, such as Apache Hive, Hue, and Impala, can all make use of Kerberos authentication.

## Authorization

Authorization is concerned with who or what has access or control over a given resource or service. Since Hadoop merges together the capabilities of multiple varied, and previously separate IT systems as an enterprise data hub that stores and works on all data within an organization, it requires multiple authorization controls with varying granularities. In such cases, Hadoop management tools simplify setup and maintenance by:

- Tying all users to groups, which can be specified in existing LDAP or AD directories.
- Providing role-based access control for similar interaction methods, like batch and interactive SQL queries.

Cloudera Runtime currently provides the following forms of access control:

- Traditional POSIX-style permissions for directories and files, where each directory and file is assigned a single owner and group. Each assignment has a basic set of permissions available; file permissions are simply read, write, and execute, and directories have an additional permission to determine access to child directories.
- HDFS ACLs that provide fine-grained control of permissions for HDFS files by allowing you to set different permissions for specific named users or named groups.
- Apache HBase uses ACLs to authorize various operations (READ, WRITE, CREATE, ADMIN) by column, column family, and column family qualifier. HBase ACLs are granted and revoked to both users and groups.
- Role-based access control with Apache Ranger.

## Encryption

Data at rest and data in transit encryption function at different technology layers of the cluster:

| Layer | Description |
|---|---|
| Application | Applied by the HDFS client software, HDFS Transparent Encryption lets you encrypt specific folders contained in HDFS. To securely store the required encryption keys, Cloudera recommends using the Key Trustee Server in conjunction with HDFS encryption. Data stored temporarily on the local filesystem outside HDFS by CDP components (including Impala, MapReduce, YARN, or HBase) can also be encrypted. |
| Operating System | At the Linux OS file system layer, encryption can be applied to an entire volume. |
| Network | Network communications between client processes and server processes (HTTP, RPC, or TCP/IP services) can be encrypted using industry-standard TLS/SSL. |

# Monitoring a Cluster Using Cloudera Manager

Cloudera Manager provides many features for monitoring the health and performance of the components of your clusters (hosts, service daemons) as well as the performance and resource demands of the jobs running on your clusters.

The following monitoring features are available in Cloudera Manager:

- Monitoring Cloudera Runtime Services - describes how to view the results of health tests at both the service and role instance level. Various types of metrics are displayed in charts that help with problem diagnosis. Health tests include advice about actions you can take if the health of a component becomes concerning or bad. You can also view the history of actions performed on a service or role, and can view an audit log of configuration changes.
- Monitoring Hosts - describes how to view information pertaining to all the hosts on your cluster: which hosts are up or down, current resident and virtual memory consumption for a host, what role instances are running on a host, which hosts are assigned to different racks, and so on. You can look at a summary view for all hosts in your cluster or drill down for extensive details about an individual host, including charts that provide a visual overview of key metrics on your host.
- Activities - describes how to view the activities running on the cluster, both at the current time and through dashboards that show historical activity, and provides many statistics, both in tabular displays and charts, about the resources used by individual jobs. You can compare the performance of similar jobs and view the performance of individual task attempts across a job to help diagnose behavior or performance problems.
- Events - describes how to view events and make them available for alerting and for searching, giving you a view into the history of all relevant events that occur cluster-wide. You can filter events by time range, service, host, keyword, and so on.
- Alerts - describes how to configure Cloudera Manager to generate alerts from certain events. You can configure thresholds for certain types of events, enable and disable them, and configure alert notifications by email or using SNMP trap for critical events. You can also suppress alerts temporarily for individual roles, services, hosts, or even the entire cluster to allow system maintenance/troubleshooting without generating excessive alert traffic.
- Lifecycle and Security Auditing - describes how to view service, role, and host lifecycle events such as creating a role or service, making configuration revisions for a role or service, decommissioning and recommissioning hosts, and running commands recorded by Cloudera Manager management services. You can filter audit event entries by time range, service, host, keyword, and so on.
- Charting Time-Series Data - describes how to search metric data, create charts of the data, group (facet) the data, and save those charts to user-defined dashboards.
- Logs - describes how to access logs in a variety of ways that take into account the current context you are viewing. For example, when monitoring a service, you can easily click a single link to view the log entries related to that specific service, through the same user interface. When viewing information about a user's activity, you can easily view the relevant log entries that occurred on the hosts used by the job while the job was running.
- Reports - describes how to view historical information about disk utilization by user, user group, and by directory and view cluster job activity user, group, or job ID. These reports are aggregated over selected time periods

(hourly, daily, weekly, and so on) and can be exported as XLS or CSV files. You can also manage HDFS directories as well, including searching and setting quotas.

- Troubleshooting Cluster Configuration and Operation - contains solutions to some common problems that prevent you from using Cloudera Manager and describes how to use Cloudera Manager log and notification management tools to diagnose problems.

# Cloudera Management Service

The Cloudera Management Service is a set of roles used by Cloudera Manager to manage and monitor clusters.

The Cloudera Management Service implements various management features as a set of roles:

- Host Monitor - collects health and metric information about hosts
- Service Monitor - collects health and metric information about services and activity information from the YARN and Impala services
- Event Server - aggregates relevant Hadoop events and makes them available for alerting and searching
- Alert Publisher - generates and delivers alerts for certain types of events
- Reports Manager - generates reports that provide an historical view into disk utilization by user, user group, and directory, processing activities by user and YARN pool, and HBase tables and namespaces. This role is not added in Cloudera Express.

You can view the status of the Cloudera Management Service by doing one of the following:

- Select  Clusters Cloudera Management Service .
- On the HomeStatus tab, in Cloudera Management Service table, click the Cloudera Management Service link.

## Health Tests

Cloudera Manager monitors the health of the services, roles, and hosts that are running in your clusters using *health tests*. The Cloudera Management Service also provides health tests for its roles. Role-based health tests are enabled by default. For example, a simple health test is whether there's enough disk space in every NameNode data directory. A more complicated health test may evaluate when the last checkpoint for HDFS was compared to a threshold or whether a DataNode is connected to a NameNode. Some of these health tests also aggregate other health tests: in a distributed system like HDFS, it's normal to have a few DataNodes down (assuming you've got dozens of hosts), so we allow for setting thresholds on what percentage of hosts should color the entire service down.

Health tests can return one of three values: Good, Concerning, and Bad. A test returns Concerning health if the test falls below a warning threshold. A test returns Bad if the test falls below a critical threshold. The overall health of a service or role instance is a roll-up of its health tests. If any health test is Concerning (but none are Bad) the role's or service's health is Concerning; if any health test is Bad, the service's or role's health is Bad.

In the Cloudera Manager Admin Console, health tests results are indicated with colors: Good , Concerning , and Bad .

One common question is whether monitoring can be separated from configuration. One of the goals for monitoring is to enable it without needing to do additional configuration and installing additional tools (for example, Nagios). By having a deep model of the configuration, Cloudera Manager is able to know which directories to monitor, which ports to use, and what credentials to use for those ports. This tight coupling means that, when you install Cloudera Manager all the monitoring is enabled.

## Metric Collection and Display

To perform monitoring, the Service Monitor and Host Monitor collects metrics. A *metric* is a numeric value, associated with a name (for example, "CPU seconds"), an entity it applies to ("host17"), and a timestamp. Most metric collection is performed by the Agent. The Agent communicates with a supervised process, requests the metrics, and forwards them to the Service Monitor. In most cases, this is done once per minute.

A few special metrics are collected by the Service Monitor. For example, the Service Monitor hosts an HDFS canary, which tries to write, read, and delete a file from HDFS at regular intervals, and measure whether it succeeded, and how long it took. Once metrics are received, they're aggregated and stored.

Using the Charts page in the Cloudera Manager Admin Console, you can query and explore the metrics being collected. Charts display *time series*, which are streams of metric data points for a specific entity. Each metric data point contains a timestamp and the value of that metric at that timestamp.

Some metrics (for example, total_cpu_seconds) are counters, and the appropriate way to query them is to take their rate over time, which is why a lot of metrics queries contain the dt0 function. For example, dt0(total_cpu_seconds). (The dt0 syntax is intended to remind you of derivatives. The `0` indicates that the rate of a monotonically increasing counter should never have negative rates.)

### Events, Alerts, and Triggers

An *event* is a record that something of interest has occurred – a service's health has changed state, a log message (of the appropriate severity) has been logged, and so on. Many events are enabled and configured by default.

An *alert* is an event that is considered especially noteworthy and is triggered by a selected event. Alerts are shown with an

badge when they appear in a list of events. You can configure the Alert Publisher to send alert notifications by email or by SNMP trap to a trap receiver.

A *trigger* is a statement that specifies an action to be taken when one or more specified conditions are met for a service, role, role configuration group, or host. The conditions are expressed as a tsquery statement, and the action to be taken is to change the health for the service, role, role configuration group, or host to either Concerning (yellow) or Bad (red).

### Related Information

tsquery Language

# Cluster Configuration Overview

Cloudera Manager manages the configuration of all roles running in a cluster.

When Cloudera Manager configures a service, it allocates *roles* that are required for that service to the hosts in your cluster. The role determines which service daemons run on a host.

For example, for an HDFS service instance, Cloudera Manager configures:

- One host to run the NameNode role.
- One host to run as the secondary NameNode role.
- One host to run the Balancer role.
- Remaining hosts as to run DataNode roles.

A role group is a set of configuration properties for a role type, as well as a list of role instances associated with that group. Cloudera Manager automatically creates a default role group named Role Type Default Group for each role type.

When you run the installation or upgrade wizard, Cloudera Manager configures the default role groups it adds, and adds any other required role groups for a given role type. For example, a DataNode role on the same host as the NameNode might require a different configuration than DataNode roles running on other hosts. Cloudera Manager creates a separate role group for the DataNode role running on the NameNode host and uses the default configuration for DataNode roles running on other hosts.

Cloudera Manager wizards autoconfigure role group properties based on the resources available on the hosts. For properties that are not dependent on host resources, Cloudera Manager default values typically align with Cloudera Runtime default values for that configuration. Cloudera Manager deviates when the Cloudera Runtime default is not a recommended configuration or when the default values are invalid.

**Related Information**

Cloudera Runtime Configuration Properties Reference

# Server and Client Configuration

Cloudera Manager generates server and client configuration files from its database.

Administrators are sometimes surprised that modifying /etc/hadoop/conf and then restarting HDFS has no effect. That is because service instances started by Cloudera Manager do not read configurations from the default locations. To use HDFS as an example, when not managed by Cloudera Manager, there would usually be one HDFS configuration per host, located at /etc/hadoop/conf/hdfs-site.xml. Server-side daemons and clients running on the same host would all use that same configuration.

Cloudera Manager distinguishes between server and client configuration. In the case of HDFS, the file /etc/hadoop/conf/hdfs-site.xml contains only configuration relevant to an HDFS client. That is, by default, if you run a program that needs to communicate with Hadoop, it will get the addresses of the NameNode and JobTracker, and other important configurations, from that directory. A similar approach is taken for /etc/hbase/conf and /etc/hive/conf.

In contrast, the HDFS role instances (for example, NameNode and DataNode) obtain their configurations from a private per-process directory, under /var/run/cloudera-scm-agent/process/*UNIQUE-PROCESS-NAME*. Giving each process its own private execution and configuration environment allows Cloudera Manager to control each process independently. For example, here are the contents of an example 879-hdfs-NAMENODE process directory:

```
$ tree -a /var/run/cloudera-scm-Agent/process/879-hdfs-NAMENODE/
  /var/run/cloudera-scm-Agent/process/879-hdfs-NAMENODE/
  ### cloudera_manager_Agent_fencer.py
  ### cloudera_manager_Agent_fencer_secret_key.txt
  ### cloudera-monitor.properties
  ### core-site.xml
  ### dfs_hosts_allow.txt
  ### dfs_hosts_exclude.txt
  ### event-filter-rules.json
  ### hadoop-metrics2.properties
  ### hdfs.keytab
  ### hdfs-site.xml
  ### log4j.properties
  ### logs
  #   ### stderr.log
  #   ### stdout.log
  ### topology.map
  ### topology.py
```

Distinguishing between server and client configuration provides several advantages:

- Sensitive information in the server-side configuration, such as the password for the Hive Metastore RDBMS, is not exposed to the clients.
- A service that depends on another service may deploy with customized configuration. For example, to get good HDFS read performance, Impala needs a specialized version of the HDFS client configuration, which may be harmful to a generic client. This is achieved by separating the HDFS configuration for the Impala daemons (stored in the per-process directory mentioned above) from that of the generic client (/etc/hadoop/conf).
- Client configuration files are much smaller and more readable. This also avoids confusing non-administrator Hadoop users with irrelevant server-side properties.

# Cloudera Manager API

The Cloudera Manager API provides configuration and service lifecycle management, service health information and metrics, and allows you to configure Cloudera Manager itself.

The Cloudera Manager API is served on the same host and port as the Cloudera Manager Admin Console, and does not require an extra process or extra configuration. The API supports HTTP Basic Authentication, accepting the same users and credentials as the Cloudera Manager Admin Console.

You can also access the Cloudera Manager Swagger API user interface from the Cloudera Manager Admin Console. Go to SupportAPI Explorer to open Swagger.

You can view the Cloudera Manager REST API documentation from the Cloudera Manager Admin Console:

1. Open the Cloudera Manager Admin Console.
2. Select Support > API Documentation

### Resources

Cloudera Manager REST API documentation

Javadoc

Cloudera Manager API tutorial

Current API version: v51

# Virtual Private Clusters and Cloudera SDX

A Virtual Private Cluster uses the Cloudera Shared Data Experience (SDX) to simplify deployment of both on-premise and cloud-based applications and enable workloads running in different clusters to securely and flexibly share data.

The architecture of a Virtual Private Cluster provides many advantages for deploying workloads and sharing data among applications, including a shared catalog, unified security, consistent governance, and data lifecycle management.

In traditional cluster deployments, a Regular cluster contains storage nodes, compute nodes, and other services such as metadata services and security services that are collocated in a single cluster. This traditional architecture provides many advantages where computational services such as Impala and YARN can access collocated data sources such as HDFS or Hive.

With Virtual Private Clusters and the SDX framework, a new type of cluster is available in Cloudera Manager called a Compute cluster. A Compute cluster runs computational services such as Hive Execution Service, Spark, or YARN but you configure these services to access data hosted in another Regular cluster, called the Base cluster. Using this architecture, you can separate compute and storage resources in a variety of ways to flexibly maximize resources.

## Advantages of Separating Compute and Data Resources

Separating compute and data resources in a Virtual Private Cluster has important advantages for many workloads.

Architectures that separate compute resources from data resources can provide many advantages for a CDP deployment:

- More options for deploying computational and storage resources
  - You can selectively deploy resources using on-premise servers, containers, virtual machines, or cloud resources that are tailored for the workload. When you configure a Compute cluster, you can provision

hardware that is more appropriate for computational workloads while the Base cluster can use hardware that emphasizes storage capacity. Cloudera recommends that each cluster use similar hardware.
  - Software resources can be optimized to best use computational and storage resources.
- Ephemeral clusters

  When deploying clusters on cloud infrastructure, having separate clusters for compute and storage allows you to temporarily shut down the compute clusters and avoid unnecessary expense -- while still leaving the data available to other applications.
- Workload Isolation
  - Compute clusters can help to resolve resource conflicts among users accessing the cluster. Longer running or resource intensive workloads can be isolated to run in dedicated compute clusters that do not interfere with other workloads.
  - Resources can be grouped into clusters that allow IT to allocate costs to the teams that use the resources.

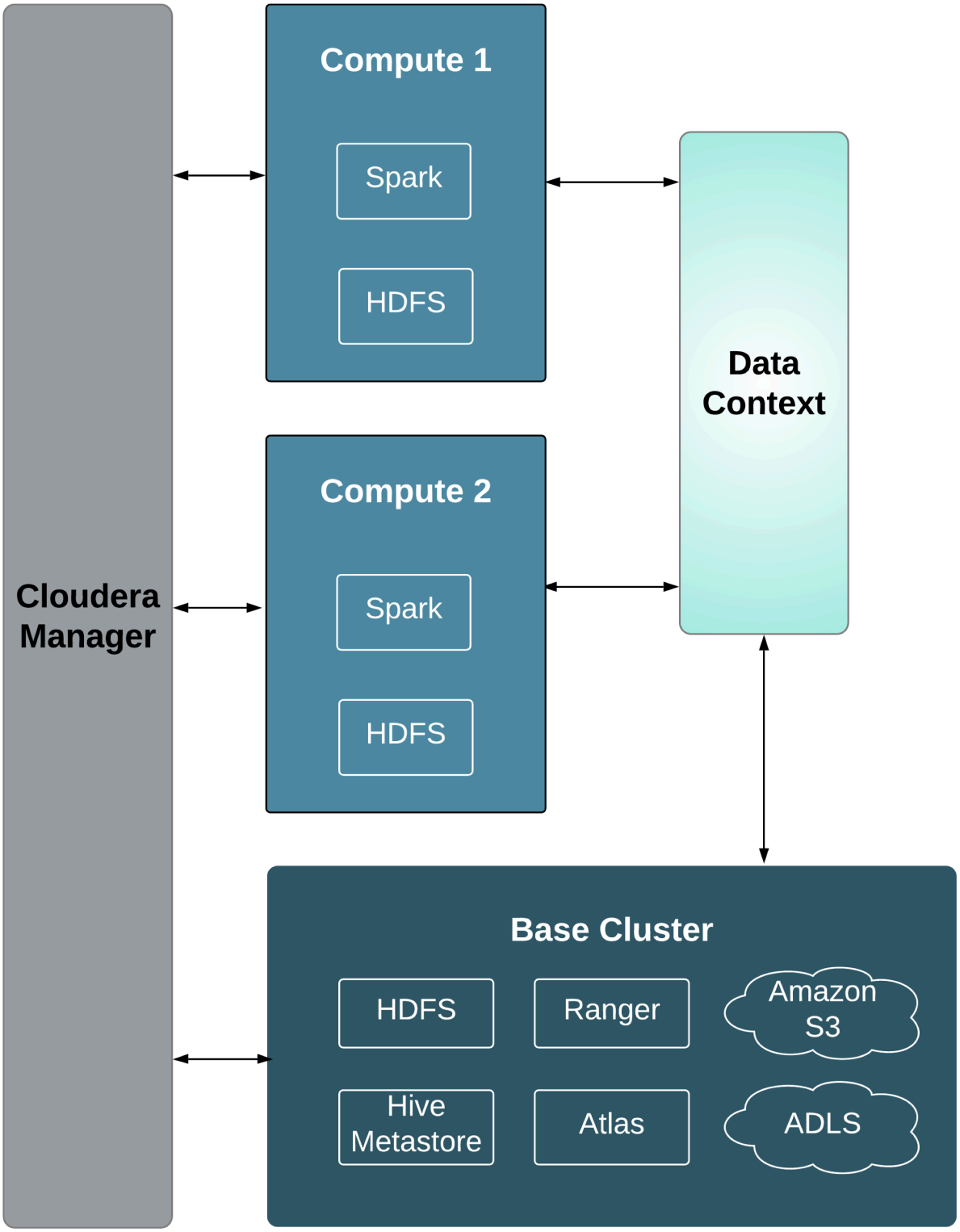
### Related Information

Compatibility Considerations for Virtual Private Clusters

## Architecture

If you are creating Virtual Private Clusters, it is important to understand the architecture of compute clusters and how they related to Data contexts.

A Compute cluster is configured with compute resources such as Spark or Hive Execution. Workloads running on these clusters access data by connecting to a Data Context for the Base cluster. A Data Context is a connector to a Regular cluster that is designated as the Base cluster. The Data context defines the data, metadata and security services deployed in the Base cluster that are required to access the data. Both the Compute cluster and Base cluster are managed by the same instance of Cloudera Manager. A Base cluster can contain any Cloudera Runtime services -- but only HDFS, Hive, Atlas, Ranger, Amazon S3, and Microsoft ADLS can be shared using the data context.




Either the Core Configuration service or the HDFS service is required on Compute clusters as temporary, persistent space. If you are using Hive-on-Tez in the Compute Cluster, the HDFS service is required. A compute cluster uses these services to store temporary files used in multi-stage MapReduce jobs. In addition, the following services may deployed as needed:

- Hive Execution Service (This service supplies the HiveServer2 role only.)

- Hue
- Kafka
- Spark 2
- Oozie (only when Hue is available, and is a requirement for Hue)
- YARN
- HDFS
- Stub DFS (Stub DFS replaces Core Settings and requires the Storage Operations role.)

The functionality of a Virtual Private cluster is a subset of the functionality available in Regular clusters, and the versions of Cloudera Runtime that you can use are limited. For more information, see Compatibility Considerations for Virtual Private Clusters on page 33.

### Related Information

Networking Considerations for Virtual Private Clusters

## Performance Trade Offs

Learning about the types of workloads appropriate for Virtual Private Clusters can help you decide if this architecture is appropriate for your needs.

### Throughput

Because data will be accessed over network connections to other clusters, this architecture may not be appropriate for workloads that scan large amounts of data. These types of workloads may run better on Regular clusters where compute and storage are collocated and features such as Impala short-circuit reads can provide improved performance.

You can evaluate the networking performance using the Network Performance Inspector.

### Ephemeral Clusters

For deployments where the Compute clusters are shut down or suspended when they are not needed, cluster services that collect historical data do not collect data when the Compute clusters are off-line, and the history is not available to users. This affects services such as the Spark History Server and the YARN JobHistory Server. When the Compute cluster restarts, the previous history will be available.

## Compatibility Considerations for Virtual Private Clusters

### Related Information

Networking Considerations for Virtual Private Clusters

## CDH and Cloudera Runtime Version Compatibility

The Cloudera Runtime version of the Compute cluster must match the major.minor version of the Base cluster. Support for additional combinations of Compute and Base cluster versions may be added at a later time. The following Cloudera Runtime and CDH versions are supported for creating Virtual Private Clusters:

- CDH 5.15
- CDH 5.16
- CDH 6.0
- CDH 6.1
- CDH 6.2
- CDH 6.3
- Cloudera Runtime 7.0.3
- Cloudera Runtime 7.1.1 and higher

## Licensing Requirements

A valid CDP Private Cloud Base Edition license is required to use Virtual Private Clusters. You cannot access this feature with a CDP Private Cloud Base Edition Trial license.

## Components

- SOLR – Not supported on Compute clusters
- Kudu – Not supported on Compute clusters.
- HDFS
  - Either the Core Configuration service or the HDFS service is required on Compute clusters as temporary, persistent space. If you are using Hive-on-Tez in the Compute Cluster, the HDFS service is required. The intent is to use this for Hive temporary queries and is also recommended for multi-stage Spark ETL jobs.
  - Cloudera recommends a minimum storage of 1TB per host, configured as the HDFS DataNode storage directory.
  - The Base cluster must have an HDFS service.
  - Isilon is not supported on the Base cluster.
  - S3 and ADLS connectors are supported only on the Base cluster; Compute clusters will use the supplied S3 or ADLS credentials from their associated Base cluster
  - The HDFS service on the Base cluster must be configured for high availability.
  - Cloudera highly recommends enabling high availability for the HDFS service on the Compute cluster, but it is not required.
  - The Base and compute nameservice names must be distinct.
  - The following configurations for the local HDFS service on a compute cluster must match the configurations on the Base cluster. This enables services on the Compute cluster to correctly access services on the Base cluster:
    - Hadoop RPC protection
    - Data Transfer protection
    - Enable Data Transfer Encryption
    - Kerberos Configurations
    - TLS/SSL Configuration (only when the cluster is not using Auto-TLS). See Security on page 35.
  - Do not override the nameservice configuration in the Compute cluster by using Advanced Configuration Snippets.
  - The HDFS path to compute cluster home directories use the following structure: /mc/<*CLUSTER_ID*>/fs/user

    You can find the <*CLUSTER_ID*> by clicking the cluster name in the Cloudera Manager Admin Console. The URL displayed by your browser includes the cluster id. For example, in the URL below, the cluster ID is 1.

    ```
    http://myco-1.prod.com:7180/cmf/clusters/1/status
    ```
- Backup and Disaster Recovery (BDR) – not supported when the source cluster is a Compute cluster and the target cluster is running a version of Cloudera Manager lower than 6.2.
- YARN and MapReduce
  - If both MapReduce (MR1, Deprecated in CM6) and YARN (MR2) are available on the Base cluster, dependent services in the Compute cluster such as Hive Execution Service will use MR1 because of the way service dependencies are handled in Cloudera Manager. To use YARN, you can update the configuration to make these Compute cluster services depend on YARN (MR2) before using YARN in your applications.
- Impala – Not supported on Compute clusters

- Hue
  - Only one Hue service instance is supported on a Compute cluster.
  - The Hue service on Compute clusters will not share user-specific query history with the Hue service on other Compute clusters or Hue services on the Base cluster
  - Hue examples may not install correctly due to differing permissions for creating tables and inserting data. You can work around this problem by deleting the sample tables and then re-adding them.
  - If you add Hue to a Compute cluster after the cluster has already been created, you will need to manually configure any dependencies on other services (such as Hive or Hive Execution Service) in the Compute cluster.
- Hive Execution Service

  The newly introduced “Hive execution service” is only supported on Compute clusters, and is not supported on Base or Regular clusters.

  To enable Hue to run Hive queries on a Compute cluster, you must install the Hive Execution Service on the Compute cluster.

## Compute Cluster Services

Only the following services can be installed on Compute clusters:

- Hive Execution Service (This service supplies the HiveServer2 role only.)
- Hue
- Kafka
- Spark 2
- Oozie (only when Hue is available, and is a requirement for Hue)
- YARN
- HDFS
- Stub DFS (Stub DFS replaces Core Settings and requires the Storage Operations role.)

## Cloudera Navigator Support

Navigator lineage and metadata, and Navigator KMS are not supported on Compute clusters.

## Cloudera Manager Permissions

Cluster administrators who are authorized to only see Base or Compute clusters can only see and administer these clusters, but cannot create, delete, or manage Data Contexts. Data context creation and deletion is only allowed with the Full Administrator use role or for unrestricted Cluster Administrators.

## Security

- KMS
  - Base cluster:
    - Hadoop KMS is not supported.
    - KeyTrustee KMS is supported on Base cluster.
  - Compute cluster: Any type of KMS is not supported.
- Authentication/User directory
  - Users should be identically configured on hosts on compute and Base clusters, as if the nodes are part of the same cluster. This includes Linux local users, LDAP, Active Directory, or other third-party user directory integrations.

- Kerberos
  - If a Base cluster has Kerberos installed, then the Compute and Base clusters must both use Kerberos in the same Kerberos realm. The Cloudera Manager Admin Console helps facilitate this configuration in the cluster creation process to ensure compatible Kerberos configuration.
- TLS
  - If the Base cluster has TLS configured for cluster services, Compute cluster services must also have TLS configured for services that access those Base cluster services.
  - Cloudera strongly recommends enabling Auto-TLS to ensure that services on Base and Compute clusters uniformly use TLS for communication.
  - If you have configured TLS, but are not using Auto-TLS, note the following:
    - You must create an identical configuration on the Compute cluster hosts before you add them to a Compute cluster using Cloudera Manager. Copy all files located in the directories specified by the following configuration properties, from the Base clusters to the Compute cluster hosts:
      - hadoop.security.group.mapping.ldap.ssl.keystore
      - ssl.server.keystore.location
      - ssl.client.truststore.location
    - Cloudera Manager will copy the following configurations to the compute cluster when you create the Compute cluster.
      - hadoop.security.group.mapping.ldap.use.ssl
      - hadoop.security.group.mapping.ldap.ssl.keystore
      - hadoop.security.group.mapping.ldap.ssl.keystore.password
      - hadoop.ssl.enabled
      - ssl.server.keystore.location
      - ssl.server.keystore.password
      - ssl.server.keystore.keypassword
      - ssl.client.truststore.location
      - ssl.client.truststore.password

### Networking

Workloads running on Compute clusters will communicate heavily with hosts on the Base cluster; Customers should have network monitoring in place for networking hardware (such as switches including top-of-rack, spine/leaf routers and so on) to track and adjust bandwidth between racks hosting hosts utilized in Compute and Base clusters.

You can also use the Cloudera Manager Network Performance Inspector to evaluate your network.

For more information on how to set up networking, see Networking Considerations for Virtual Private Clusters on page 36.

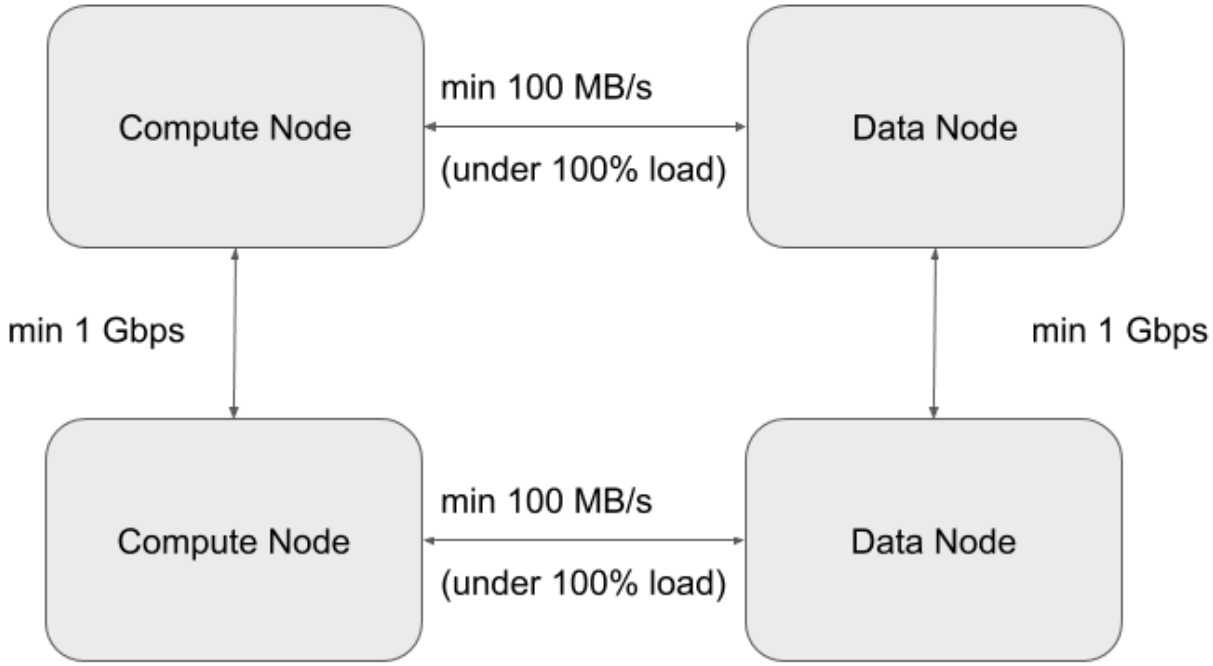
## Networking Considerations for Virtual Private Clusters

### Minimum Network Performance Requirements

A Virtual Private Cluster deployment has the following requirements for network performance:

- A worst case IO throughput of 100MB/s , i.e., 1 Gbp/s of network throughput (sustained) between any compute node and any storage node. To achieve the worst case, we will measure throughput when all computenodes are reading/writing from the storage nodes at the same time. This type of parallel execution is typical of big data applications.
- A worst case network bandwidth of 1Gbps between any two workload cluster Nodes or any two base cluster Nodes.

The following picture summarizes these requirements:




## Sizing and designing the Network Topology

### Minimum and Recommended Performance Characteristics

This section can be used to derive network throughput requirements, following the minimum network throughputs outlined in the table below.

**Note:** For the purposes of this guide, we are going to use the following terms:

- North-South (NS) traffic patterns indicate network traffic between Compute and Storage tiers.
- East-West (EW) traffic patterns indicate internal network traffic within the storage or compute clusters.

| Tier | Minimum Per-Node Storage IO throughput (MB/s) | Minimum per-Node network throughput (Gbps) (EW + NS) | Recommended per-Node Storage IO throughput (MB/s) | Recommended per-Node network throughput (Gbps) (EW + NS) |
|---|---|---|---|---|
| Compute (This can be virtualized or bare-metal) | 100 | 2 | 200 | 4 |
| Storage (This can be virtualized or bare-metal) | 400 | 8 | 800 | 16 |

**Attention:** The storage backend will predicate the maximum number of compute Nodes connecting to it. The backend is not just about capacity, but also throughput. Since the backend will be HDFS DataNodes, proper sizing of the backend nodes (or Nodes) is required. The following guidelines will help size the backend accordingly:

- Assuming SATA drives are being used for storage, each drive can generate about 100MB/s of throughput in bare-metal clusters and should expect to generate 70-80MB/s in virtualized clusters with directly attached storage.
- Each drive requires one physical CPU core. So a node with 12 drives should have at least 12 cores.
- The network bandwidth should be planned with 2x NS traffic in mind. For instance, if a node has 12 drives, then NS traffic expected should be 1200MB/s (1.2GB/s), which is ~ 10 Gbps in network throughput. Plan for 20 Gbps to address EW traffic.
- The inter-networking between the compute layer and the storage layer needs to factor in how much throughput will flow in the NS direction. The section below articulates the considerations for network design and illustrates the impact of having various degrees of oversubscription on the overall achievable throughput of the clusters.

### Network Topology Considerations

The preferred network topology is spine-leaf with as close to 1:1 over subscription between leaf and spine switches, ideally aiming for no over subscription. This is so we can ensure full line-rate between any combination of storage and compute nodes. Since this architecture calls for disaggregation of compute and storage, the network design must be more aggressive in order to ensure best performance.

There are two aspects to the minimum network throughput required, which will also determine the compute to storage nodes ratio.

- The network throughput and IO throughput capabilities of the storage backend.
- The network throughput and network over subscription between compute and storage tiers (NS).

Stepping through an example scenario can help to better understand this. Assuming that this is a greenfield setup, with compute nodes and storage nodes both being virtual machines (VMs):

- Factor that the total network bandwidth is shared between EW and NS traffic patterns. So for 1 Gbps of NS traffic, we should plan for 1 Gbps of EW traffic as well.
- Network oversubscription between compute and storage tiers is 1:1
- Backend comprises of 5 nodes (VMs), each with 8 SATA drives.
  - This implies that the ideal IO throughput (for planning) of the entire backend would be ~ 4GB/s, which is ~ 800MB/s per Node. which is the equivalent of ~ 32 Gbps network throughput for the cluster, and ~ 7 Gbps per node for NS traffic pattern
  - Factoring in EW + NS we would need 14 Gbps per Node to handle the 800MB/s IO throughput per Node.
- Compute Cluster would then ideally have the following:
  - 5 hypervisor nodes each with 7 Gbps NS + 7 gpbs EW = 14 Gbps total network throughput per hypervisor.
  - This scenario can handle ~ 6 Nodes with minimum throughput (100MB/s) provided they are adequately sized in terms of CPU and memory, in order to saturate the backend pipe (6 x 100 MB/s x 5 = 3000 MB/s). Each Node should have ~ 2 Gbps network bandwidth to accommodate NS + EW traffic patterns.
  - If we consider the recommended per Node throughput of 200MB/s then, it would be 3 such Nodes per hypervisor (3 x 200 MB/s x 5 = 3000 MB/s) Each Node should have ~ 4 Gbps network bandwidth to accommodate NS + EW traffic patterns.

Assume a Compute to Storage Node ratio - 4:1. This will vary depending on in-depth sizing and a more definitive sizing will be predicated by a good understanding of the workloads that are being intended to run on said infrastructure.

The two tables below illustrate the sizing exercise through a scenario that involves a storage cluster of 50 Nodes, and following the 4:1 assumption, 200 compute nodes.

- 50 Storage nodes

- 200 Compute nodes (4:1 ratio)

### Table 2: Storage-Compute Node Level Sizing

| Tier | Per node IO (MB/s) | Per node NS network (mbps) | per node EW (mbps) | Num of Nodes | Cluster IO (MB/s) | Cluster NS (Network) (mbps) | NS oversubscription |
|---|---|---|---|---|---|---|---|
| HDFS | 400 | 3200 | 3200 | 50 | 20000 | 160000 | 1 |
| Compute Node | 100 | 800 | 800 | 200 | 20000 | 160000 | 1 |
| Compute Node | 100 | 800 | 800 | 200 | 10000 | 80000 | 2 |
| Compute Node | 100 | 800 | 800 | 200 | 6667 | 53333 | 3 |
| Compute Node | 100 | 800 | 800 | 200 | 5000 | 40000 | 4 |

### Table 3: Storage and Compute Hypervisor level sizing

| Tier | Per node IO (MB/s) | Per node NS network (mbps) | per node EW (mbps) | Num of Nodes | Hypervisor Oversubscription |
|---|---|---|---|---|---|
| Compute Hypervisor | 600 | 4800 | 4800 | 33 | 6 |
| Compute Hypervisor | 400 | 3200 | 3200 | 50 | 4 |
| Compute Hypervisor | 300 | 2400 | 2400 | 67 | 3 |
| Storage Hypervisor | 1200 | 9600 | 9600 | 17 | 3 |
| Storage Hypervisor | 800 | 6400 | 6400 | 25 | 2 |
| Storage Hypervisor | 400 | 3200 | 3200 | 50 | 1 |

One can see that the table above provides the means to ascertain the capacity of the hardware for each tier of the private cloud, given different consolidation ratios and different throughput requirements.

### Physical Network Topology

The best network topology for Hadoop clusters is spine-leaf. Each rack of hardware has its own leaf switch and each leaf is connected to every spine switch. Ideally we would not like to have any oversubscription between spine and leaf. That would result in having full line rate between any two nodes in the cluster.

The choice of switches, bandwidth and so on would of course be predicated on the calculations in the previous section.

If the Storage nodes and the Workload nodes happen to reside in clearly separate racks, then it is necessary to ensure that between the workload rack switches and the Storage rack switches, there is at least as much uplink bandwidth as the theoretical max offered by the Storage cluster. In other words, the sum of the bandwidth of all workload-only racks needs to be equivalent to the sum of the bandwidth of all storage-only racks.

For instance, taking the example from the previous section, we should have at least 60 Gbps uplink between the Storage cluster and the workload cluster nodes.

If we are to build the desired network topology based on the sizing exercise from above, we would need the following.

| Layer | Network Per-port Bandwidth | Number of Ports required | Notes |
|---|---|---|---|
| Workload | 25 | 52 | Per sizing, we needed 20 Gbps per node. However, to simplify the design, let us pick single 25 Gbps ports instead of bonded pair of 10Gbps per node. |
| Storage | 25 | 43 | |
| Total Ports | | 95 | |

Assuming all the nodes are 2 RU in form factor, we would require 5 x 42 RU racks to house this entire set up.

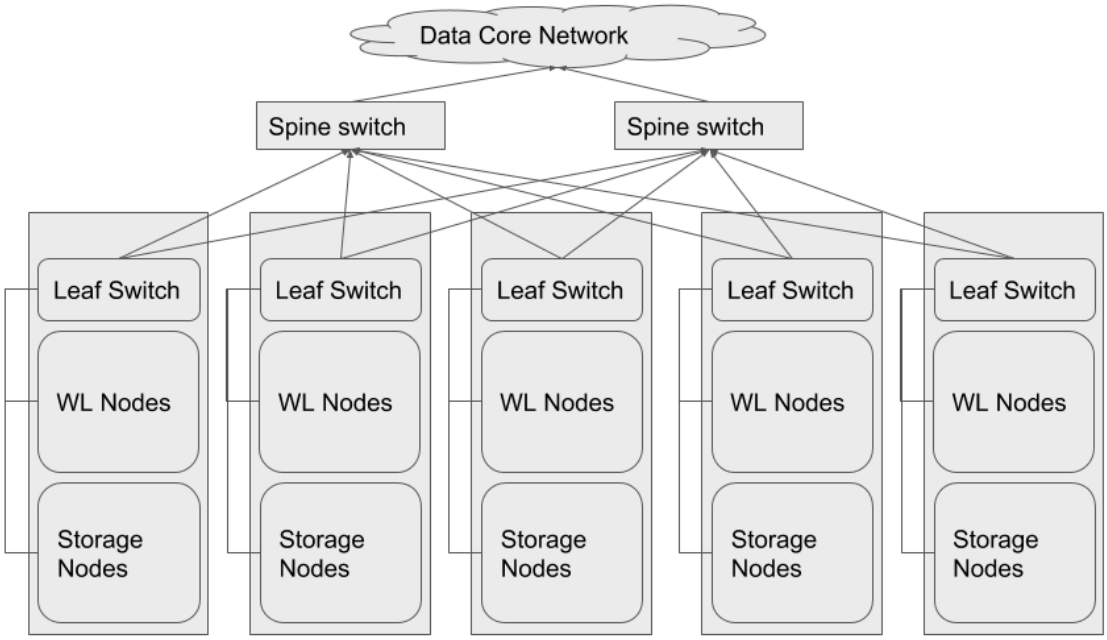
If we place the nodes from each layer as evenly distributed across the 5 Racks as we can, we would end up with following configuration.

| Rack1 | Rack2 | Rack3 | Rack4 | Rack5 |
|---|---|---|---|---|
| Spine ( 20 x 100 Gbps uplink + 2 x 100 Gbps to Core) | | | Spine (20 x 100 Gbps uplink + 2 x 100 Gbps to Core) | |
| ToR (18 x 25Gbps + 8 x 100 Gbps uplink) | ToR (20 x 25 Gbps + 8 x 100 Gbps uplink) | ToR (20 x 25 Gbps + 8 x 100 Gbps uplink ) | ToR (18 x 25 Gbps + 8 x 100 Gbps uplink) | ToR (20 x 25 Gbps + 8 x 100 Gbps uplink) |
| 10 x WL | 11 x WL | 11 x WL | 10 x WL | 11 x WL |
| 8 x Storage | 9 x Storage | 9 x Storage | 8 x Storage | 9 x Storage |

So that implies that we would have to choose ToR (Top of Rack) switches that have at least 20 x 25 Gbps ports and 8 x 100 Gbps uplink ports. Also the Spine switches would need at least 22 x 100 Gbps ports.

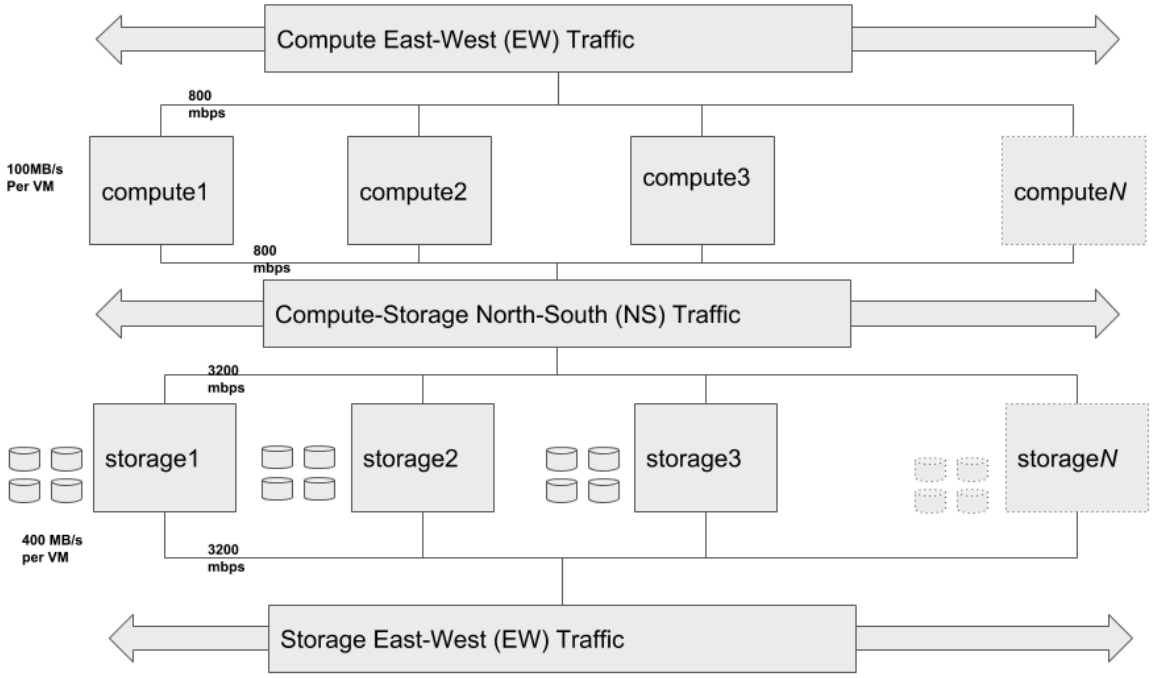
Using eight 100 Gbps uplinks from the leaf switches would result in almost 1:1 (1.125:1) oversubscription ratio between leaves (upto 20 x 25 Gbps ports) and the spine ( 4 x 100 Gbps per spine switch).

Mixing the Workload and Storage nodes in the way shown below, will help localize some traffic to each leaf and thereby reduce the pressure of N-S traffic (between Workload and Storage clusters) patterns.




**Note:** For sake of clarity, the spine switches have been shown outside the racks

The following diagram illustrates the logical topology at a virtual machine level.




The Storage E-W, Compute N-S and Compute E-W components shown above are not separate networks, but are the same network with different traffic patterns, which have been broken out in order clearly denote the different traffic patterns.