Data Hub

Data Hub Overview

Date published: 2019-12-17
Date modified: 2023-06-27

CLOUDERA

Legal Notice

Contents

Data Hub overview

Data Hub is a service for launching and managing workload clusters powered by Cloudera Runtime (Cloudera's unified open source distribution including the best of CDH and HDP). Data Hub clusters can be created on AWS, Microsoft Azure, and Google Cloud Platform.

Data Hub includes a set of cloud optimized built-in templates for common workload types, as well as a set of options allowing for extensive customization based on your enterprise's needs. Furthermore, it offers a set of convenient cluster management options such as cluster scaling, stop, restart, terminate, and more. All clusters are secured via wire encryption and strong authentication out of the box, and users can access cluster UIs and endpoints through a secure gateway powered by Apache Knox. Access to S3 cloud storage from Data Hub clusters is enabled by default (S3Guard is enabled and required in Runtime versions older than 7.2.2).

Data Hub provides complete workload isolation and full elasticity so that every workload, every application, or every department can have their own cluster with a different version of the software, different configuration, and running on different infrastructure. This enables a more agile development process.

Since Data Hub clusters are easy to launch and their lifecycle can be automated, you can create them on demand and when you don't need them, you can return the resources to the cloud.

The following diagram describes a simplified Data Hub architecture:
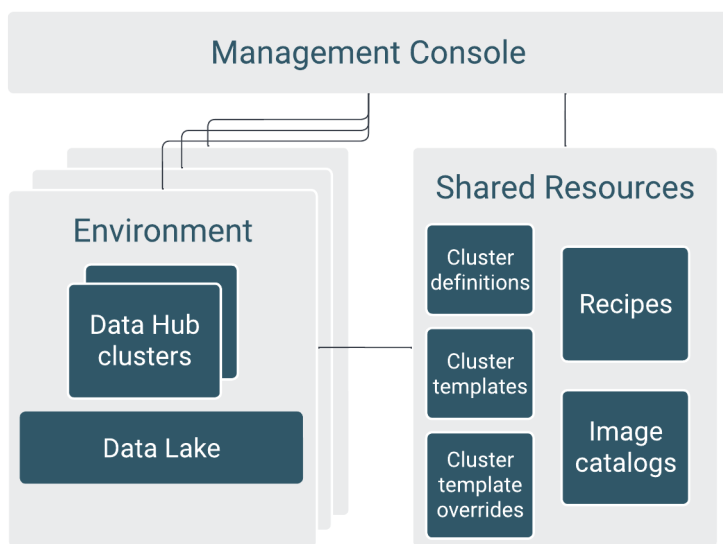
Data Hub clusters can be launched, managed, and accessed from the Management Console. All Data Hub clusters are attached to a Data Lake that runs within an environment and provides security and governance for the environment's clusters.

Data Hub provides a set of shared resources and allows you to register your own resources that can be reused between multiple Data Hub clusters. As illustrated in the following diagram, these resources (cluster definitions, cluster templates, cluster template overrides, recipes, and image catalogs) can be managed in the Management Console and shared between multiple Data Hub clusters:

- Default Cluster definitions (with cloud provider specific settings) and cluster templates (with Cloudera Runtime service configurations) allow you to quickly provision workload clusters for prescriptive use cases. You can also save your own cluster definitions and templates for future reuse.
- With cluster template overrides, you can easily specify custom configurations that override or append the properties in a built-in Data Hub template or a custom template.

- You can create and run your own pre- and post-deployment and startup scripts (called recipes) to simplify installation of third party components required by your enterprise security guidelines.
- Data Hub comes with a default image catalog that includes a set of prewarmed images (including Cloudera Manager and Cloudera Runtime). You can also customize default images and create custom image catalogs.

All of this functionality is available via the CDP web interface (as part of the Management Console service) and CDP CLI. While the CDP web interface allows you to get started quickly, the CLI allows you to create reusable scripts to automate cluster creation and cluster lifecycle management.

Related Information
Azure Load Balancers in Data Lakes and Data Hubs
Management Console Overview

Core concepts

The following concepts are key to understanding the Data Hub:

Workload clusters

All Data Hub clusters are workload clusters. These clusters are created for running specific workloads such as data engineering or data analytics.

Data Hub clusters are powered by Cloudera Runtime. They can be ephemeral or long-running. Once a Data Hub cluster is created it can be managed by using the Management Console and Cloudera Manager.

Each Data Hub cluster is created within an existing environment and is attached to a data lake, which provides the security and governance layer.

Cluster definitions

Data Hub provides a set of default cluster definitions for prescriptive use cases and allows you to create your own custom cluster definitions.

A cluster definition is a reusable cluster template in JSON format that can be used for creating multiple Data Hub clusters with identical cloud provider settings.

Data Hub includes a set of default cluster definitions for common data analytics and data engineering use cases, and allows you to save your own custom cluster definitions. While default cluster definitions can be used across all environments, custom cluster definitions are associated with one or more specific environments. Cluster definitions facilitate repeated cluster reuse and are therefore useful for creating automation around cluster creation.

The easiest way to create a custom cluster definition is to start with an existing cluster definition, modify it in the create cluster wizard, and then save it. Or you can open the custom create cluster wizard, provide all the parameters, and then save the cluster definition.

Note:

A cluster definition is not synonymous with a cluster template. While a cluster template primarily defines Cloudera Runtime services, a cluster definition primarily includes cloud provider settings. Furthermore, a cluster definition always references one specific cluster template.

Related Information
Cluster Definitions

Cluster templates

Data Hub uses cluster templates for defining cluster topology: defining host groups and components installed on each host group.

A cluster template is a reusable template in JSON format that can be used for creating multiple Data Hub clusters with identical Cloudera Runtime settings. It primarily defines the list of host groups and how components of various Cloudera Runtime services are distributed on these host groups. A cluster template allows you to specify stack, component layout, and configurations to materialize a cluster instance via Cloudera Manager REST API, without having to use the Cloudera Manager install wizard. After you provide the cluster template to Data Hub, the host groups in the JSON are mapped to a set of instances when starting the cluster, and the specified services and components are installed on the corresponding nodes.

> **Note:**
>
> A cluster template is not synonymous with a cluster definition, which primarily defines cloud provider settings. Each cluster definition must reference a specific cluster template.

Data Hub includes a few default cluster templates and allows you to upload your own cluster templates. Custom cluster templates can be uploaded and managed via the CDP web interface or CLI and then selected, when needed, for a specific cluster.

Related Information

Cluster Templates

Recipes

A recipe is a script that runs on all nodes of a selected host group at a specific time. You can use recipes to create and run scripts that perform specific tasks on your Data Hub cluster nodes.

You can use recipes for tasks such as installing additional software or performing advanced cluster configuration. For example, you can use a recipe to put a JAR file on the Hadoop classpath.

Recipes can be uploaded and managed via the CDP web interface or CLI and then selected, when needed, for a specific cluster and for a specific host group. If selected, they will be executed on a specific host group at a specified time.

Depending on the need, a recipe can be executed at various times. Available recipe execution times are:

- Before Cloudera Manager server start
- After Cloudera Manager server start
- After cluster installation
- Before cluster termination

Related Information

Recipes

Custom properties

Custom properties are configuration properties that can be set on a Cloudera Runtime cluster, but Data Hub allows you to conveniently set these during cluster creation.

When working with on premise clusters, you would typically set configuration properties once your cluster is already running by using Cloudera Manager or manually in a specific configuration file.

When creating a Data Hub cluster in CDP, you can specify a list of configuration properties that should be set during cluster creation and Data Hub sets these for you.

Note:

Alternatively, it is also possible to set configuration properties by using Cloudera Manager once the cluster is already running.

Related Information

Custom Properties

Image catalogs

Data Hub provides a set of default, prewarmed images. You can also customize the default images.

Data Hub provides a set of default prewarmed images. The prewarmed images include Cloudera Manager and Cloudera Runtime packages. Default images are available for each supported cloud provider and region as part of the default image catalog.

The following table lists the default images available:

Cloud provider	Operating system	Image type	Description
AWS	CentOS 7	Prewarmed	Includes Cloudera Manager and Cloudera Runtime packages.
Azure	CentOS 7	Prewarmed	Includes Cloudera Manager and Cloudera Runtime packages.
Google Cloud	CentOS 7	Prewarmed	Includes Cloudera Manager and Cloudera Runtime packages.

CDP does not support base images, but does support customizing the default images. You might need to customize an image for compliance or security reasons.

Default cluster configurations

Data Hub includes a set of prescriptive cluster configurations. Each of these default cluster configurations include a cloud-provider specific cluster definition, which primarily defines cloud provider settings. The cluster definition references a cluster template, which defines a number of Cloudera Runtime or Cloudera DataFlow components used for common data analytics and data engineering use cases.

Refer to the topic for each default cluster configuration to view the included services and compatible Runtime versions. These topics include links to documentation that will help you to understand the included components and use the workload cluster.

Many of the cluster components are included in the Cloudera Runtime software distribution. The Streams Messaging, Flow Management, and Streaming Analytics cluster configurations are part of Cloudera DataFlow for Data Hub and have distinct planning considerations and how-to information. See the Cloudera DataFlow for Data Hub documentation for more details.

You can access the default cluster definitions by clicking Environments, then selecting an environment and clicking the Cluster Definitions tab.

You can access the default cluster templates from Shared ResourcesCluster Templates.

To view details of a cluster definition or cluster template, click on its name. For each cluster definition, you can access a raw JSON file. For each cluster template, you can access a graphical representation ("list view") and a raw JSON file ("raw view") of all cluster host groups and their components.

Related Information

Cloudera DataFlow for Data Hub

Cloudera Runtime

Data Engineering clusters

Learn about the default Data Engineering clusters, including cluster definition and template names, included services, and compatible Runtime version.

Data Engineering provides a complete data processing solution, powered by Apache Spark and Apache Hive. Spark and Hive enable fast, scalable, fault-tolerant data engineering and analytics over petabytes of data.

Data Engineering cluster definition

This Data Engineering template includes a standalone deployment of Spark and Hive, as well as Apache Oozie for job scheduling and orchestration, Apache Livy for remote job submission, and Hue and Apache Zeppelin for job authoring and interactive analysis.

Cluster definition names

- Data Engineering for AWS
- Data Engineering for Azure
- Data Engineering for Google Cloud
- Data Engineering HA - Spark3 for AWS

  See the architectural information below for the Data Engineering HA clusters
- Data Engineering HA - Spark3 for Azure

  See the architectural information below for the Data Engineering HA clusters
- Data Engineering HA - Spark3 for Google Cloud
- Data Engineering Spark3 for AWS
- Data Engineering Spark3 for Azure
- Data Engineering Spark3 for Google Cloud

Cluster template name

- Data Engineering: Apache Spark3, Apache Hive, Apache Oozie

  Note: This cluster template was formerly named "Data Engineering: Apache Spark, Apache Hive, Apache Oozie."

- Data Engineering: HA: Apache Spark3, Apache Hive, Apache Oozie

  Note: This cluster template was formerly named "Data Engineering: HA: Apache Spark, Apache Hive, Apache Oozie."

  See the architectural information below for the Data Engineering HA clusters
- Data Engineering: Apache Spark3, Apache Hive, Apache Oozie

  Note: The "Data Engineering: Apache Spark3" cluster template is deleted. Therefore, the "Data Engineering: Apache Spark3, Apache Hive, Apache Oozie" cluster template can be used instead.

Included services

- HDFS
- Hive
- Hue
- Livy
- Spark 3
- Yarn
- Zeppelin
- ZooKeeper
- Oozie is supported for Spark 3 as of Runtime version 7.2.18

- Hive Warehouse Connector is supported as of Runtime version 7.2.16.

Compatible runtime version

7.2.16, 7.2.17, 7.2.18

Topology of the Data Engineering cluster

Topology is a set of host groups that are defined in the cluster template and cluster definition used by Data Engineering. Data Engineering uses the following topology:

Host group	Description	Node configuration
Master	The master host group runs the components for managing the cluster resources including Cloudera Manager (CM), Name Node, Resource Manager, as well as other master components such HiveServer2, HMS, Hue etc.	1 For Runtime versions earlier than 7.2.14: AWS : m5.4xlarge; gp2 - 100 GB Azure : Standard_D16_v3; StandardSSD_LRS - 100 GB GCP : e2-standard-16; pd-ssd - 100 GB For Runtime versions 7.2.14+ DE, DE Spark3, and DE HA: AWS : m5.4xlarge; gp2 - 100 GB Azure: Standard_D16_v3 GCP : e2-standard-16; pd-ssd - 100 GB
Worker	The worker host group runs the components that are used for executing processing tasks (such as NodeManager) and handling storing data in HDFS such as DataNode.	3 For Runtime versions earlier than 7.2.14: AWS : m5.2xlarge; gp2 - 100 GB Azure : Standard_D8_v3; StandardSSD_LRS - 100 GB GCP : e2-standard-8; pd-ssd - 100 GB For Runtime versions 7.2.14+ DE and DE Spark3: AWS: r5d.2xlarge - (gp2/EBS volumes) Azure: Standard_D5_v2 GCP : e2-standard-8; pd-ssd - 100 GB DE HA: AWS: r5d.4xlarge - (gp2/EBS volumes) Azure: Standard_D5_v2

Host group	Description	Node configuration
		GCP : e2-standard-8; pd-ssd - 100 GB
Compute	The compute host group can optionally be used for running data processing tasks (such as NodeManager). By default the number of compute nodes is set to 1 for proper configurations of YARN containers. This node group can be scaled down to 0 when there are no compute needs. Additionally, if load-based auto-scaling is enabled with minimum count set to 0, the compute nodegroup will be resized to 0 automatically.	0+ For Runtime versions earlier than 7.2.14: AWS : m5.2xlarge; gp2 - 100 GB Azure : Standard_D8_v3; StandardSSD_LRS - 100 GB GCP : e2-standard-8; pd-ssd - 100 GB For Runtime versions 7.2.14+ DE and DE Spark3: AWS: r5d.2xlarge - (ephemeral volumes) Azure: Standard_D5_v2 For Azure, the attached volume count for the compute host group is changed to 0. Only ephemeral/local volumes are used by default. GCP : e2-standard-8; pd-ssd - 100 GB DE HA: AWS: r5d.4xlarge - (ephemeral volumes) Azure: Standard_D5_v2 For Azure, the attached volume count for the compute host group is changed to 0. Only ephemeral/local volumes are used by default. GCP : e2-standard-8; pd-ssd - 100 GB **Note:** Compute nodes run YARN and require storage only for temporary data - this requirement is fulfilled by instance storage, so making the attached volumes count to 0 by default is more cost-efficient.
Gateway	The gateway host group can optionally be used for connecting to the cluster endpoints like Oozie, Beeline etc. This nodegroup does not run any critical services. This	0+ AWS : m5.2xlarge; gp2 - 100 GB Azure : Standard_D8_v3; StandardSSD_LRS - 100 GB

Host group	Description	Node configuration
	nodegroup resides in the same subnet as the rest of the nodegroups. If additional software binaries are required they could be installed using recipes.	GCP : e2-standard-8; pd-ssd - 100 GB

Service configurations			
Master host group CM, HDFS, Hive (on Tez), HMS, Yarn RM, Oozie, Hue, DAS, Zookeeper, Livy, Zeppelin and Sqoop	Gateway host group Configurations for the services on the master node	Worker host group Data Node and YARN NodeManager	Compute group YARN NodeManager

Configurations

Note the following:

- There is a Hive Metastore Service (HMS) running in the cluster that talks to the same database instance as the Data Lake in the environment.
- If you use CLI to create the cluster, you can optionally pass an argument to create an external database for the cluster use such as CM, Oozie, Hue, and DAS. This database is by default embedded in the master node external volume. If you specify the external database to be of type HA or NON_HA, the database will be provisioned in the cloud provider. For all these types of databases the lifecycle is still associated with the cluster, so upon deletion of the cluster, the database will also be deleted.
- The HDFS in this cluster is for storing the intermediary processing data. For resiliency, store the data in the cloud object stores.
- For high availability requirements choose the Data Engineering High Availability cluster shape.

Architecture of the Data Engineering HA for AWS cluster

The Data Engineering HA for AWS and Azure cluster shape provides failure resilience for several of the Data Engineering HA services, including Knox, Oozie, HDFS, HS2, Hue, Livy, YARN, and HMS.

Services that do not yet run in HA mode include Cloudera Manager, DAS, and Zeppelin.
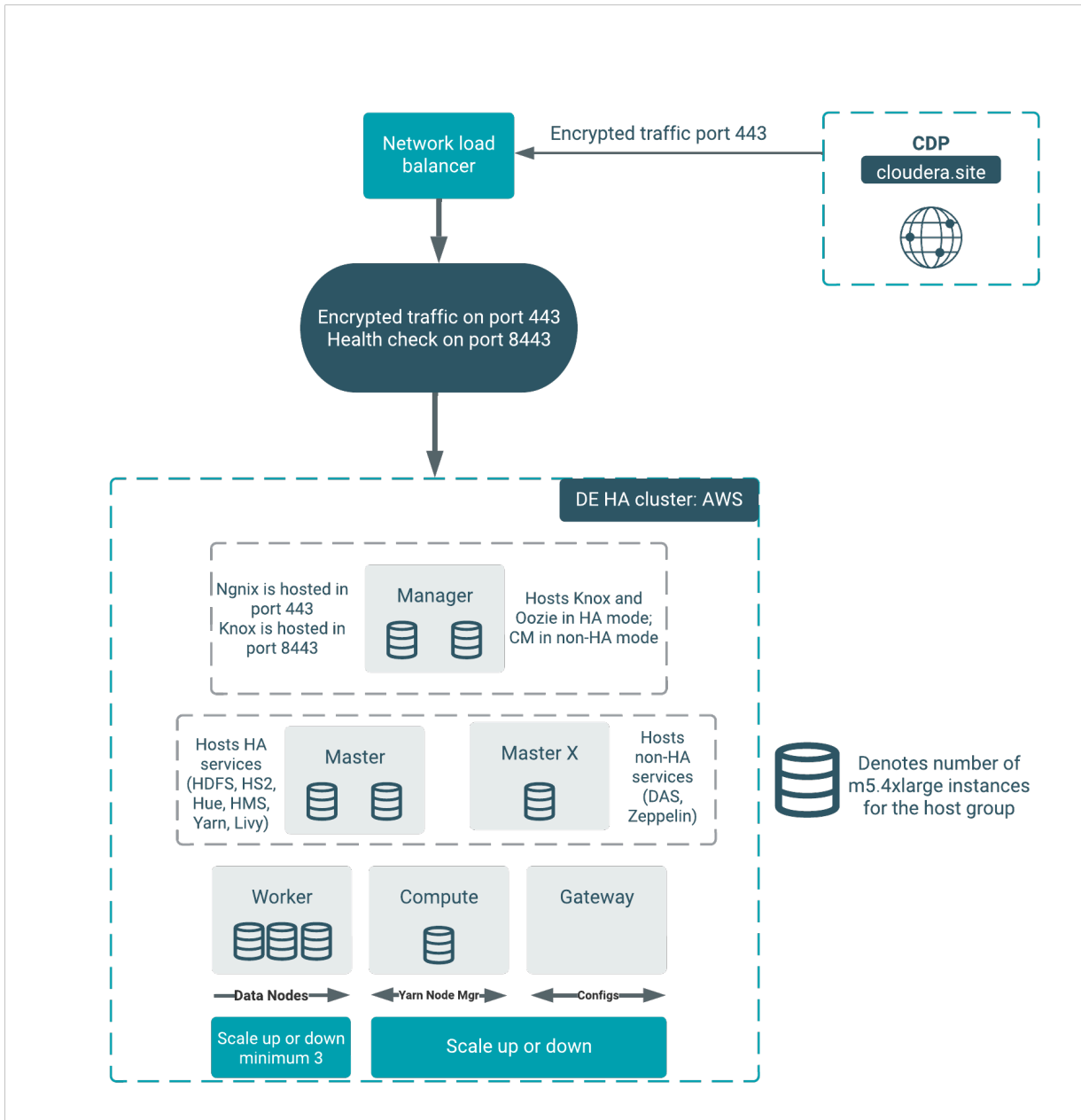
The architecture outlined in the diagram above handles the failure of one node in all of the host groups except for the "masterx" group. See the table below for additional details about the component interactions in failure mode:

Component	Failure	User experience
Knox	One of the Knox services is down	External users will still be able to access all of the UIs, APIs, and JDBC.
Cloudera Manager	The first node in manager host group is down	The cluster operations (such as repair, scaling, and upgrade) will not work.
Cloudera Manager	The second node in the manager host group is down	No impact.
HMS	One of the HMS services is down	No impact.
Hue	One of the Hue services is down in master host group	No impact.

HS2	One of the HS2 services is down in the master host group	External users will still be able to access the Hive service via JDBC. But if Hue was accessing that particular service it will not failover to the other host. The quick fix for Hue is to restart Hue to be able to use Hive functionality.
YARN	One of the YARN services is down	No impact.
HDFS	One of the HDFS services is down	No impact.
Nginx	Nginx in one of the manager hosts is down	Fifty percent of the UI, API, and JDBC calls will be affected. If the entire manager node is down, there is no impact. This is caused by the process of forwarding and health checking that is done by the network load-balancer.
Oozie	One of the Oozie servers is down in the manager host group.	No impact for AWS and Azure as of Cloudera Runtime version 7.2.11. If you create a custom template for DE HA, follow these two rules: 1. Oozie must be in single hostgroup. 2. Oozie and Hue must not be in the same hostgroup.

Important: If you are creating a DE HA cluster through the CDP CLI using the create-aws-cluster command, note that there is a CLI parameter to provision the network load-balancer in HA cluster shapes. Make sure to use the [--enable-load-balancer | --no-enable-load-balancer] parameter when provisioning a DE HA cluster via the CLI. For more information see the CDP CLI reference.

Architecture of the Data Engineering HA for Azure cluster

The Data Engineering HA for Azure cluster shape provides failure resilience for several of the Data Engineering HA services, including Knox, Oozie, HDFS, HS2, Hue, Livy, YARN, and HMS.

Services that do not yet run in HA mode include Cloudera Manager, DAS, and Zeppelin.

Component	Failure	User experience
Knox	One of the Knox services is down	External users will still be able to access all of the UIs, APIs, and JDBC.
Cloudera Manager	The first node in manager host group is down	The cluster operations (such as repair, scaling, and upgrade) will not work.
Cloudera Manager	The second node in the manager host group is down	No impact.
HMS	One of the HMS services is down	No impact.
Hue	One of the Hue services is down in master host group	No impact.

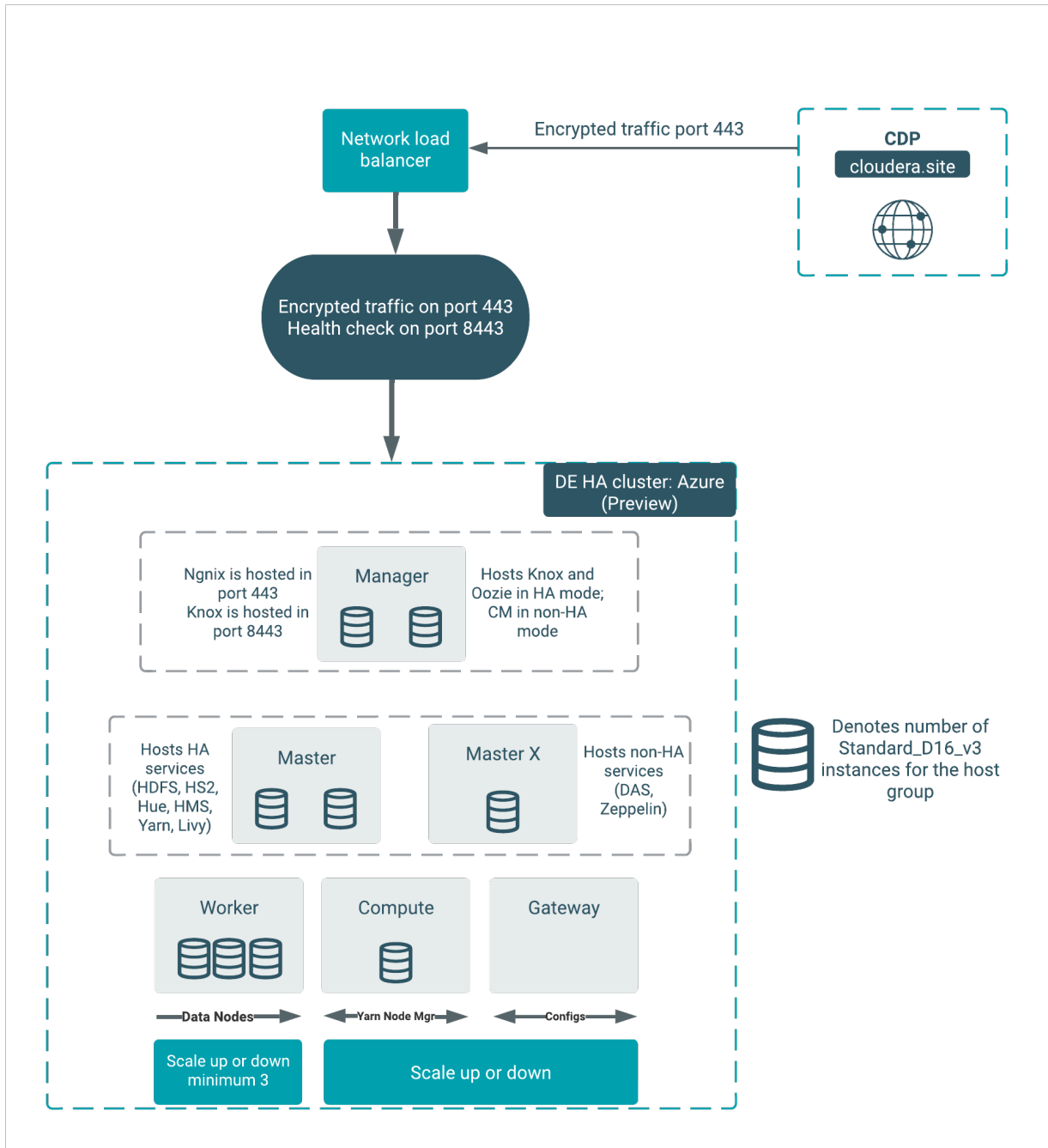| HS2	One of the HS2 services is down in the master host group	External users will still be able to access the Hive service via JDBC. But if Hue was accessing that particular service it will not failover to the other host. The quick fix for Hue is to restart Hue to be able to use Hive functionality.
YARN	One of the YARN services is down	No impact.
HDFS	One of the HDFS services is down	No impact.
Nginx	Nginx in one of the manager hosts is down	Fifty percent of the UI, API, and JDBC calls will be affected. If the entire manager node is down, there is no impact. This is caused by the process of forwarding and health checking that is done by the network load-balancer.
Oozie	One of the Oozie servers is down in the manager host group.	No impact for AWS and Azure as of Cloudera Runtime version 7.2.11. If you create a custom template for DE HA, follow these two rules: 1. Oozie must be in single hostgroup. 2. Oozie and Hue must not be in the same hostgroup.

Important: If you are creating a DE HA cluster through the CDP CLI using the create-azure-cluster command, note that there is a CLI parameter to provision the network load-balancer in HA cluster shapes. Make sure to use the [--enable-load-balancer | --no-enable-load-balancer] parameter when provisioning a DE HA cluster via the CLI. For more information see the CDP CLI reference.

GCP HA (Preview)

Note: HA for Oozie is not yet available in the GCP template.

Custom templates

Any custom DE HA template that you create must be forked from the default templates of the corresponding version. You must create a custom cluster definition for this with the JSON parameter "enableLoadBalancers": true , using the create-aws/azure/gcp-cluster CLI command parameter --request-template. Support for pre-existing custom cluster definitions will be added in a future release. As with the template, the custom cluster definition must be forked from the default cluster definition. You are allowed to modify the instance types and disks in the custom cluster definition. You must not change the placement of the services like Cloudera Manager, Oozie, and Hue. Currently the custom template is fully supported only via CLI.

The simplest way to change the DE HA definition is to create a custom cluster definition. In the Create Data Hub UI when you click Advanced Options, the default definition is not used fully, which will cause issues in the HA setup.

Related Information
HDFS
Hive
Hue
Livy
Oozie
Spark
YARN
Zeppelin
Zookeeper

Data Mart clusters

Learn about the default Data Mart and Real Time Data Mart clusters, including cluster definition and template names, included services, and compatible Runtime version.

Data Mart is an MPP SQL database powered by Apache Impala designed to support custom Data Mart applications at big data scale. Impala easily scales to petabytes of data, processes tables with trillions of rows, and allows users to store, browse, query, and explore their data in an interactive way.

Data Mart clusters

The Data Mart template provides a ready to use, fully capable, standalone deployment of Impala. Upon deployment, it can be used as a standalone Data Mart to which users point their BI dashboards using JDBC/ODBC end points. Users can also choose to author SQL queries in Cloudera's web-based SQL query editor, Hue, and run them with Impala providing a delightful end-user focused and interactive SQL/BI experience.

Cluster definition names

- Data Mart for AWS
- Data Mart for Azure
- Data Mart for Google Cloud

Cluster template name

CDP - Data Mart: Apache Impala, Hue

Included services

- HDFS
- Hue
- Impala

Compatible Runtime versions

7.1.0, 7.2.0, 7.2.1, 7.2.2, 7.2.6, 7.2.7, 7.2.8, 7.2.9, 7.2.10, 7.2.11, 7.2.12, 7.2.14, 7.2.15, 7.2.16, 7.2.17, 7.2.18

Real Time Data Mart clusters

The Real-Time Data Mart template provides a ready-to-use, fully capable, standalone deployment of Impala and Kudu. You can use a Real Time Data Mart cluster as a standalone Data Mart which allows high throughput streaming ingest, supporting updates and deletes as well as inserts. You can immediately query data through BI dashboards using JDBC/ODBC end points. You can choose to author SQL queries in Cloudera's web-based SQL query editor, Hue. Executing queries with Impala, you will enjoy an end-user focused and interactive SQL/BI experience. This template is commonly used for Operational Reporting, Time Series, and other real time analytics use cases.

Cluster definition names

- Real-time Data Mart for AWS
- Real-time Data Mart for Azure
- Real-time Data Mart for Google Cloud

Cluster template name

CDP - Real-time Data Mart: Apache Impala, Hue, Apache Kudu, Apache Spark

Included services

- HDFS
- Hue
- Impala
- Kudu
- Spark 2

- Yarn

Compatible Runtime versions

> 7.1.0, 7.2.0, 7.2.1, 7.2.2, 7.2.6, 7.2.7, 7.2.8, 7.2.9, 7.2.10, 7.2.11, 7.2.12, 7.2.14, 7.2.15, 7.2.16, 7.2.17

Cluster definition names

- Real-time Data Mart - Spark3 for AWS
- Real-time Data Mart - Spark3 for Azure
- Real-time Data Mart - Spark3 for Google Cloud

Cluster template name

> Real-time Data Mart: Apache Impala, Hue, Apache Kudu, Apache Spark3

Included services

- HDFS
- Hue
- Impala
- Kudu
- Spark 3
- Yarn

Compatible Runtime versions

> 7.2.16, 7.2.17, 7.2.18

High availability

Cloudera recommends that you use high availability (HA), and track any services that are not capable of restarting or performing failover in some way.

Impala HA

The Impala nodes offer high availability. The following Impala services are not HA.

- Catalog service
- Statestore service

Kudu HA

Both Kudu Masters and TabletServers offer high availability.

Related Information

HDFS

Hue

Impala

Kudu

Spark

YARN

Operational Database clusters

The Operational Database (OpDB) template is removed from the CDP DataHub. You can access the Cloudera Operational Database (COD) instead as a superior product.

The COD is a NoSQL database powered by Apache HBase designed to support custom OLTP applications that want to leverage the power of BigData. Apache HBase is a NoSQL, scale-out database that can easily scale to petabytes and stores tables with millions of columns and billions of rows.

COD also contains Apache Phoenix which provides a way to use HBase through an SQL interface.

Cloudera recommends you to use the COD to create Operational Database clusters.

Related Information

Cloudera Operational Database

Getting started with Operational Database

Before you create an Operational Database cluster

Creating an Operational Database cluster

HDFS

HBase

Knox

Zookeeper

Phoenix

Streams Messaging clusters

Learn about the default Streams Messaging clusters, including cluster definition and template names, included services, and compatible Runtime version.

Streams Messaging provides advanced messaging and real-time processing on streaming data using Apache Kafka, centralized schema management using Schema Registry, as well as management and monitoring capabilities powered by Streams Messaging Manager, as well as cross-cluster Kafka topic replication using Streams Replication Manger and Kafka partition rebalancing with Cruise Control.

This template sets up a fault-tolerant standalone deployment of Apache Kafka and supporting Cloudera components (Schema Registry, Streams Messaging Manager, Streams Replication Manager and Cruise Control), which can be used for production Kafka workloads in the cloud or as a disaster recovery instance for on-premises. Kafka clusters.

> **Note:**
>
> Streams Messaging clusters have distinct planning considerations and how-to information. See the Cloudera DataFlow for Data Hub documentation for information about:
>
> - Planning your Streams Messaging cluster deployment
> - Creating your first Streams Messaging cluster
> - Connecting Kafka clients to CDP Public Cloud clusters

Cluster definition names

- Streams Messaging Heavy Duty for AWS
- Streams Messaging Light Duty for AWS
- Streams Messaging HA for AWS
- Streams Messaging Heavy Duty for Azure
- Streams Messaging Light Duty for Azure
- Streams Messaging HA for Azure (Technical Preview)
- Streams Messaging Heavy Duty for GCP
- Streams Messaging Light Duty for GCP
- Streams Messaging HA for GCP (Technical Preview)

Cluster template name

- CDP - Streams Messaging Heavy Duty
- CDP - Streams Messaging Light Duty
- CDP - Streams Messaging High Availability

Included services

- Kafka

- Schema Registry
- Streams Messaging Manager
- Streams Replication Manager
- Cruise Control
- Kafka Connect

Compatible Runtime version

- 7.1.0 (Preview)
- 7.2.0
- 7.2.1
- 7.2.2
- 7.2.6
- 7.2.7
- 7.2.8
- 7.2.9
- 7.2.10
- 7.2.11
- 7.2.12
- 7.2.14
- 7.2.15
- 7.2.16
- 7.2.17

Related Information

Setting up your Streams Messaging cluster

Ingesting Data into CDP Public Cloud

Kafka

Schema Registry

Streams Messaging Manager

Streams Replication Manager

Flow Management clusters

Learn about the default Flow Management clusters, including cluster definition and template names, included services, and compatible Runtime versions.

Flow Management delivers high-scale data ingestion, transformation, and management to enterprises from any-to-any environment. It addresses key enterprise use cases such as data movement, continuous data ingestion, log data ingestion, and acquisition of all types of streaming data including social, mobile, clickstream, and IoT data.

The Flow Management template includes a no-code data ingestion and management solution powered by Apache NiFi. With NiFi's intuitive graphical interface and 300+ processors, Flow Management enables easy data ingestion and movement between CDP services as well as 3rd party cloud services. NiFi Registry is automatically set up and provides a central place to manage versioned Data Flows.

Note:

Flow Management clusters have distinct planning considerations and how-to information. See the Cloudera DataFlow for Data Hub documentation for information about:

- Planning your Flow Management cluster deployment
- Creating your first Flow Management cluster
- Security considerations for Flow Management clusters
- Using Apache NiFi to ingest data into CDP Public Cloud
- Using NiFi and NiFi Registry

Cluster definition names

- Flow Management Light Duty for AWS
- Flow Management Light Duty for Azure
- Flow Management Light Duty for GCP
- Flow Management Heavy Duty for AWS
- Flow Management Heavy Duty for Azure
- Flow Management Heavy Duty for GCP

Cluster template name

- CDP - Flow Management: Light Duty
- CDP - Flow Management: Heavy Duty

Included services

- NiFi
- NiFI Registry

Compatible Runtime versions

- 7.1.0
- 7.2.0
- 7.2.1
- 7.2.2
- 7.2.6
- 7.2.7
- 7.2.8
- 7.2.9
- 7.2.10
- 7.2.11
- 7.2.12
- 7.2.14
- 7.2.15
- 7.2.16
- 7.2.17

Related Information

Setting up your Flow Management cluster

Apache NiFi documentation

Apache NiFi Registry documentation

Streaming Analytics clusters

Learn about the default Streaming Analytics clusters, including cluster definition and template names, included services, and compatible Runtime version.

Streaming Analytics offers real-time stream processing and stream analytics with low-latency and high scaling capabilities powered by Apache Flink.

Streaming Analytics templates include Apache Flink that works out of the box in stateless or heavy state environments. Beside Flink, the template includes its supporting services namely YARN, Zookeeper and HDFS. The Heavy Duty template comes preconfigured with RocksDB as state backend, while Light Duty clusters use the default Heap state backend. You can create your streaming application by choosing between Kafka, Kudu, and HBase as datastream connectors.

You can also use SQL to query real-time data with SQL Stream Builder (SSB) in the Streaming Analytics template. By supporting the SSB service in CDP Public Cloud, you can simply and easily declare expressions that filter, aggregate, route, and otherwise mutate streams of data. SSB is a job management interface that you can use to compose and run SQL on streams, as well as to create durable data APIs for the results.

> **Note:**
>
> Streaming Analytics clusters have distinct planning considerations and how-to information. See the Cloudera DataFlow for Data Hub documentation for information about:
>
> - Planning your Streaming Analytics cluster deployment
> - Creating your first Streaming Analytics cluster
> - Analyzing data using Apache Flink
> - Querying data using SQL Stream Builder

Cluster definition names

- Streaming Analytics Light Duty for AWS
- Streaming Analytics Light Duty for Azure
- Streaming Analytics Light Duty for GCP
- Streaming Analytics Heavy Duty for AWS
- Streaming Analytics Heavy Duty for Azure
- Streaming Analytics Heavy Duty for GCP

Cluster template name

- 7.2.17 - Streaming Analytics Light Duty
- 7.2.17 - Streaming Analytics Heavy Duty

Included services

- Flink
- SQL Stream Builder
- YARN
- Zookeeper
- HDFS
- Kafka

  > **Important:** In the Streaming Analytics cluster templates, Kafka service is included by default to serve as a background service only for the websocket ouput and sampling feature of SQL Stream Builder. The Kafka service in the Streaming Analytics cluster template cannot be used for production, you need to use the Streams Messaging cluster template when Kafka is needed for your deployment.

Compatible Runtime version

- 7.2.2
- 7.2.6
- 7.2.7
- 7.2.8
- 7.2.9
- 7.2.10

- 7.2.11
- 7.2.12
- 7.2.14
- 7.2.15
- 7.2.16
- 7.2.17

Related Information

Setting up your Streaming Analytics cluster
Flink
YARN
Zookeeper
HDFS

Data Discovery and Exploration clusters

Learn about the default Data Discovery and Exploration clusters, including cluster definition and template names, included services, and compatible Runtime version.

Data Discovery and Exploration

Explore and discover data sets ad-hoc. Do relevance-based analytics over unstructured data (logs, images, text, PDFs, etc). Get started with search or log analytics. Make data more accessible to everyone with Data Discovery and Exploration.

Cluster Definition Names

- Data Discovery and Exploration for AWS
- Data Discovery and Exploration for Azure

Cluster Template Name

- Data Discovery and Exploration

Included Services

- Solr
- Spark 2
- HDFS
- Hue
- YARN
- ZooKeeper

Compatible Runtime Versions

7.2.0, 7.2.1, 7.2.2, 7.2.6, 7.2.7, 7.2.8, 7.2.9, 7.2.10, 7.2.11, 7.2.12, 7.2.14, 7.2.15, 7.2.16, 7.2.17

Cluster Definition Names

- Data Discovery and Exploration - Spark3 for AWS
- Data Discovery and Exploration - Spark3 for Azure
- Data Discovery and Exploration - Spark3 for Google Cloud

Cluster Template Name

- Data Discovery and Exploration for Spark3

Included Services

- Solr
- Spark 3

- HDFS
- Hue
- YARN
- ZooKeeper

Compatible Runtime Version

7.2.18

Default cluster topology

Data Hub uses a specific cluster topology including the following host groups: master, worker, and compute.

Data Hub uses the following host groups. These host groups are defined in cluster templates and cluster definitions used by Data Hub:

Host group	Description	Number of nodes
Master	The master host group runs the components for managing the cluster resources (including Cloudera Manager), storing intermediate data (e.g. HDFS), processing tasks, as well as other master components.	1
Worker	The worker host group runs the components that are used for executing processing tasks (such as NodeManager) and handling storing data in HDFS such as DataNode).	1+
Compute	The compute host group can optionally be used for running data processing tasks (such as NodeManager).	0+

Node repair

Data Hub monitors clusters, ensuring that when host-level failures occur, they are reported right away and can be quickly resolved by performing manual repair which deletes and replaces failed nodes and reattaches the disks.

For each Data Hub cluster, CDP checks for Cloudera Manager agent heartbeat on all cluster nodes. If the Cloudera Manager agent heartbeat is lost on a node, a failure is reported for that node. This may happen for the following reasons:

- The Cloudera Manager agent process exited on a node
- An instance crashed
- An instance was terminated

Once a failure is reported, options are available for you to repair the failure manually.

Important: A manual repair operation on an unhealthy node will replace the disk mounted as root, which means all data on the root disk is lost.

To avoid the loss of important data, do not store it in the root disk. Instead, do one of the following:

- Store data that needs to persist beyond the lifetime of the cluster in S3 or ADLS Gen 2, depending on the platform in use.
- Store data that needs to survive a repair operation in /hadoopfs/fsN/ (where N is an integer), as long as it is not so large that it could crowd out components that use that location.

  For example, storing 1 GB of data in /hadoopfs/fs1/save_me would be an option to ensure that the data is available in a replacement node after a manual repair operation.

Manual repair is enabled for all clusters by default and covers all nodes, including the Cloudera Manager server node (by default, Cloudera Manager is installed on the master node). When a node fails, a notification about node failure is printed in the Event History, the affected node is marked as unhealthy, and an option to repair the cluster is available from the Actions menu. From the Hardware tab you can choose to repair a single node or specific nodes within a host group. There are two ways to repair the cluster:

- Repair the failed nodes: (1) All non-ephemeral disks are detached from the failed nodes. (2) Failed nodes are removed (3) New nodes of the same type are provisioned. (4) The disks are attached to the new volumes, preserving the data.
- Delete the failed nodes: Failed nodes are deleted with their attached volumes.

Note:

As of March 26, 2021, the manual repair option is available for all clusters and typically covers all nodes, including the master node, regardless of whether or not the cluster uses an external database.

For clusters created before March 26, 2021, manual repair covers all nodes except the master node (Cloudera Manager server node). There is an exception: if a cluster (created before March 26, 2021) uses an external database, then manual repair is enabled for the master node. Ephemeral disks cannot be reattached (they are deleted and new disks are provisioned).

Cloud storage

The options on the "Cloud Storage" page allow you to optionally specify the base storage location used for YARN and Zeppelin.

During environment creation under Data Access > Storage Location Base you configure a default S3, ADLS Gen2, or Google Cloud Storage base storage location for the environment, and all Data Hub clusters created within that environment use this location. The Cloud Storage options in the Data Hub cluster wizard allow you to additionally specify a different location for YARN application logs and Zeppelin Notebook's root directory:

- Existing Base Storage Location - By default, this is set to the Storage Location Base configured on environment level. If you do not want to make any changes, simply leave this blank. If you would like to use a different location for YARN application logs and Zeppelin Notebook's root directory, you can specify a different S3 or ADLS Gen2 location. Note that the specified S3, ADLS Gen2, or Google Cloud Storage location must exist prior to Data Hub cluster creation and that you must adjust the IAM policies created during environment's cloud storage setup to make sure that IDBroker has write access to this location.
- Path for YARN Application Logs property - This directory structure gets created automatically during cluster creation. You can customize it if you would like it be different than what is suggested by default.
- Path for Zeppelin Notebooks Root Directory property - This directory structure gets created automatically during cluster creation. You can customize it if you would like it to be different than what is suggested by default.

Note: Any S3 bucket that you designate for Data Hub cloud storage on AWS must be in the same region as the environment.

Databases for cluster components

By default, Data Hub provisions embedded databases for all cluster services that require it. These databases run on the Cloudera Manager server node (By default Cloudera Manager is installed on the master node).

Note:

One exception is Hive Metastore, which always uses the external database attached to the data lake. Therefore, no embedded database is needed for Hive Metastore.