Cloudera Observability On-Premises Overview

Date published: 2024-01-31
Date modified: 2024-08-14

CLOUDERA

Legal Notice

Contents

What is Cloudera Observability On-Premises and how is it useful

Cloudera Observability On-Premises is a Cloudera service for your air gap environment that helps you interactively understand your environment, workloads, clusters, and resources. Its wide range of metrics and health tests help you identify and troubleshoot existing and potential problems and the prescriptive guidance and recommendations help you quickly address and optimize those problems. When a workload completes, diagnostic information about the job or query and the cluster that processed them is collected by Telemetry Publisher, a role in the Cloudera Manager Management Service, and sent to Cloudera Observability On-Premises.

Cloudera Observability On-Premises enables you to interactively understand your workloads, clusters, and resources, and optimize your systems through:

- A wide range of metrics and health tests that help you identify and troubleshoot issues and potential issues.
- Prescriptive guidance and recommendations that help you quickly address and optimize those problems.
- Performance baselines and historical analysis that help you identify and address performance problems.

In addition, Cloudera Observability On-Premises also enables you to:

- Visually display your workload cluster's current and historical costs that help you plan and forecast budgets, future workload environments, and justify current user groups and resources.
- Trigger actions in real-time across jobs and queries that help you take steps to alleviate potential problems.
- Break down your workload metrics into more meaningful views for your business requirements that help you analyze specific workload criteria. For example, you can analyze how queries that access a particular database or that use a specific resource pool are performing against your SLAs. Or you can examine how all the queries are performing on your cluster that are sent by a specific user.

About the Cloudera Observability On-Premises user interface hierarchy

Provides a brief introduction to the web interface, its hierarchical components, and the frequently used interface elements of Cloudera Observability On-Premises. Cloudera recommends that you take a moment to familiarize yourself with the user interface, its components, and elements.

The Cloudera Observability On-Premises web UI hierarchically displays the health, performance, and status of your environment, workload clusters, engines, and resources, including the costs associated with your data infrastructure, from the top-down. Its dashboard components include statistics, performance, health, and prescriptive guidance visually displayed in chart widgets, cards, and tabular views.

After logging in to the Cloudera Observability On-Premises UI the landing page opens. This page is also called the Cloudera Observability On-Premises HOME page.

Tip: Clicking the Cloudera Observability On-Premises icon brings you back to this page.

About the main navigation panel

The **Main** navigation panel enables access to the following Cloudera Observability On-Premises features:

- Financial Governance, which opens the **Chargeback** page that displays the total costs and the hourly CPU and memory usage for all of your cost centers, including the unutilized resource usage costs from the Uncategorised section.

  From its Actions list you can:

  - Configure your cost centers criteria based on CPU and Memory costs and resource usage.
  - Create cost centers, which separate costs across user or pool usage and track their workload resource consumption costs.
- Analytics, which depending on the tier level within the hierarchy displays:

  - The **Environments** page that lists your environments, including:

    - The environment's platform type. At this time, only Private Cloud Base environments are supported
    - The platform's version number.
    - The type of data content.
    - The date and time that telemetry data was last collected.

    Where, a Cloudera Observability On-Premises Environment hierarchically represents the association of your environment and its resources, clusters, engines, and their workloads (Jobs and Queries).

    To display clusters that have not sent data in the last 30 days, select the Show stale clusters check box.
- Access Management, which opens the Access Management page for managing your Cloudera Observability On-Premises cluster policies and access roles when the Role Based Access property is enabled in Cloudera Manager.

  A Cloudera Observability On-Premises Cluster Policy defines the conditions for a role based access type. For example, they define:

  - Who is entitled to access jobs and queries that are created by the user.
  - Who is entitled to create and administer cost centers and view cluster costs.
  - Who is entitled to access and administer jobs and queries within either a specific cluster or across all clusters within the Cloudera Observability On-Premises environment.

  Limiting the trust boundary for jobs, queries, cluster costs, and administrative management at the cluster level, enables more control over the security and access management of your Cloudera Observability On-Premises environment.
- User login name, which enables you to securely log out.

About the environment navigation panel

The **Environments** navigation panel hierarchically displays the environment from its parent tier (environment name) to the lower tier levels (clusters, engines, jobs and queries, and, where applicable, your Hive Metastore).

The following table describes the environment categories, which are displayed in the Environment panel dependent on the selected Cloudera Observability On-Premises environment type:

Table 1: Environment categories

Category	Description
ENGINES	Lists the Hive, Impala, Spark, Oozie, and MapReduce workload engines for Private Cloud Base environments.
HIVE METASTORE	Lists the Hive and Impala engine metastores. When a metastore is selected the HMS Summary view opens displaying information about the current state and activity of all your tables in the selected Environment. To display details about each table in your system that were processed in Hive and Impala engines, regardless of whether they have been queried or not, select the Tables tab, which opens the HMS Tables view.

The following table describes the Cluster, and Engine Summary pages:

Table 2: Environment summary pages

Cluster Summary	Displays the Summary page and the Cloudera Observability On-Premises features available for the environment's cluster and subscription entitlement: • Summary tab, which displays performance trends and metrics about the processed jobs and queries as chart widgets and enables you to view historical trends for analysis when you select a predefined or custom time period from the Time-Range filter list. • Workloads tab, which enables the creation and management of your Workload views for the environment's cluster. • Auto Actions tab, which enables the creation and management of your Auto Actions events for the environment's cluster.
Engine Summary	Displays information about the workload jobs or queries run by the selected engine, such as which jobs or queries have failed or are slow, their processing time, missed SLAs (thresholds), user and pool metrics, and outlier issues.

About the Cloudera Observability On-Premises chart widgets

The Cloudera Observability On-Premises chart widgets enable you to quickly observe real-time and historical patterns, trends, and outliers of your workload data. They provide quick insights into the health and performance of your workloads, clusters, and resources. Where:

• Hovering over an element with your mouse pointer, such as over a time-line or a data point, displays more information about the element underneath.
• Clicking a link within a chart widget or a bar within bar chart widget, such as the Suboptimal bar chart widget, opens the engine's Jobs or Queries page that contains more information in a tabular view for you to investigate further.

About the Jobs and Queries pages

Depending on the engine chosen, the engine's Jobs or Queries page provides information about each of the jobs or queries that are serviced by the engine. You can filter further by selecting a filter option from one of the filter categories; Pool, User, Status, Health Check, or Duration.

For more information about a specific job's or query's health, execution details, baseline, and trends, open their respective page by clicking the link in the Job or Query column and selecting the tab of interest. To investigate further, these pages also provide prescriptive guidance and recommendations that enable you to address problems and optimize solutions.

About the Cloudera Observability On-Premises common web user interface features

Learn about the Cloudera Observability On-Premises web UI navigation features, charts, and actions.

As you explore your environment's statistics and health the following UI elements are available to help you navigate and explore your diagnostic data:

Navigation

• Navigation drawer panels (side-bars) that toggle between open and close, which enable you to view more or less real estate space and provide access from the parent tier (environment) to the lower tier levels (clusters, engines, jobs, and queries).
• Drawer panels that toggle between open and close on the right-side of the page to provide more detailed information about a component.

- Breadcrumbs that are displayed at the top of the page, which displays the name of your current location and its preceding pages. You can move between these pages by clicking on a breadcrumb location.

Charts and statistic banners

- Statistic banners, which display dynamic and interactive pie charts, rose charts, bar charts, and statistic cards about your jobs, queries, tables, and engines.
- Chart widgets, which enable you to dynamically observe real-time and historical patterns, trends, and outliers of your workload data, jobs, and queries.

Action tasks

- Filters, which enable you to refine your selection and display only the components of interest.
- Action menus, which list the actions that you can perform on an environment or component, such as on a data service or engine.
- Time-range list (time-picker), which displays the current or historical data for the selected time-period.
- Search fields and lists, which enable you to locate a specific component, such as an environment, data service, or cluster ID.

Using the Cloudera Observability On-Premises user interface

Describes a few frequently used interface elements of Cloudera Observability On-Premises that help you identify and troubleshoot your workload issues.

The following examples describe a few interface elements of Cloudera Observability On-Premises that enable you to quickly identify workload problems, health issues, resource contentions, and abnormal or degraded performance problems.

These examples assume that you have logged in to the Cloudera Observability On-Premises UI.

Identify workload problems and health issues

You can easily locate what engines are running on your clusters from the environment's Cluster Summary page and what jobs and queries are failing the health tests with the Suboptimal chart widget.

The Suboptimal chart widget, displays the distribution of jobs and the jobs and queries that failed. This chart widget enables you to visually see at a glance what issues are currently impacting your jobs or queries and how they are executing on your cluster.

Location: The Suboptimal chart widget is found on the engine's Summary page, which, depending on the environment selected, is accessed by selecting the environment for analysis in the **Environments** page and then in the Environment navigation panel, drilling down through the environment's categories and selecting an engine of interest.
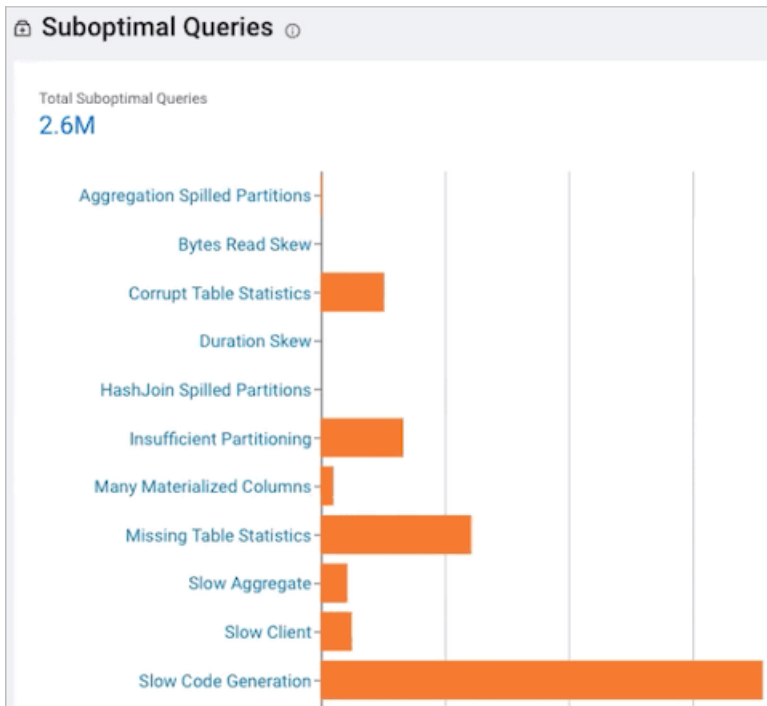
Identify and address resource contentions

Cloudera Observability On-Premises provides the following chart widgets that help you analyze and identify resource consumption and contention problems:

- The Resource Consumption By Services chart widget displays the CPU and memory consumption for each service across the time range you selected. Hover your mouse over the timeline, to display the percentage of CPU or memory consumed by each of the cluster's services.

  Location: The Resource Consumption By Services chart widget is found on the **Cluster Summary** page of the Private Cloud Base environment.
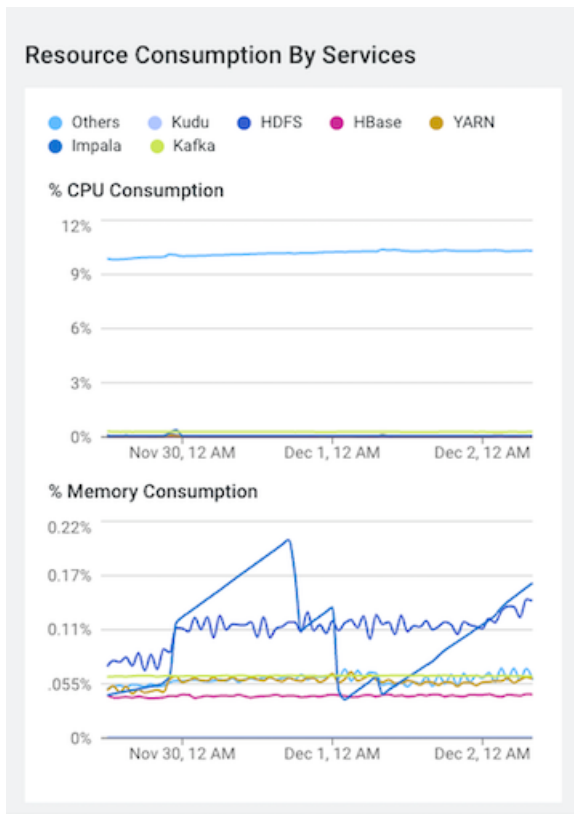
- The Resource Consumption By Nodes chart widget displays the CPU and memory consumption for each node in the cluster. Hover your mouse over the time line, to display the amount of CPU or memory, as a percentage, that is consumed by each node and its services.

  Location: The Resource Consumption By Nodes chart widget is found on the **Cluster Summary** page of the Private Cloud Base environment.
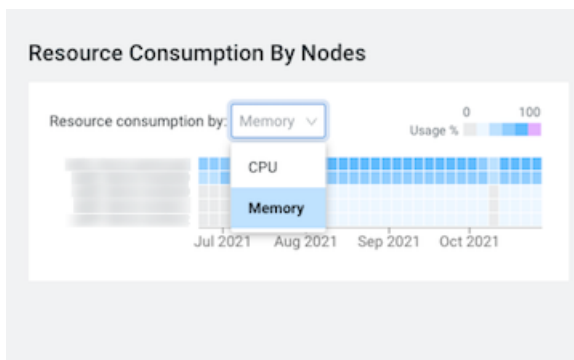

- The Memory Utilization chart widget, displays the aggregated maximum amount of memory that is used by the queries on any node performing the processing. It helps you identify inefficient queries that are consuming the most amount of memory and decide if you need to allocate more memory to continue running your query jobs.

  Location: The Memory Utilization chart widget is found on the Impala engine's Summary page, which, depending on the environment selected, is accessed by selecting the environment for analysis in the **Environments** page and then in the Environment navigation panel, drilling down through the environment's categories and selecting the Impala engine of interest.

- The Resource Consumption chart widget, displays the concurrent use of CPU and memory consumption for a workload across the timeline you selected.

  Location: The Resource Consumption chart widget is found on the Impala engine's Summary page. which, depending on the environment selected, is accessed by selecting the environment for analysis in the **Environments** page and then in the Environment navigation panel, drilling down through the environment's categories and selecting the Impala engine of interest.
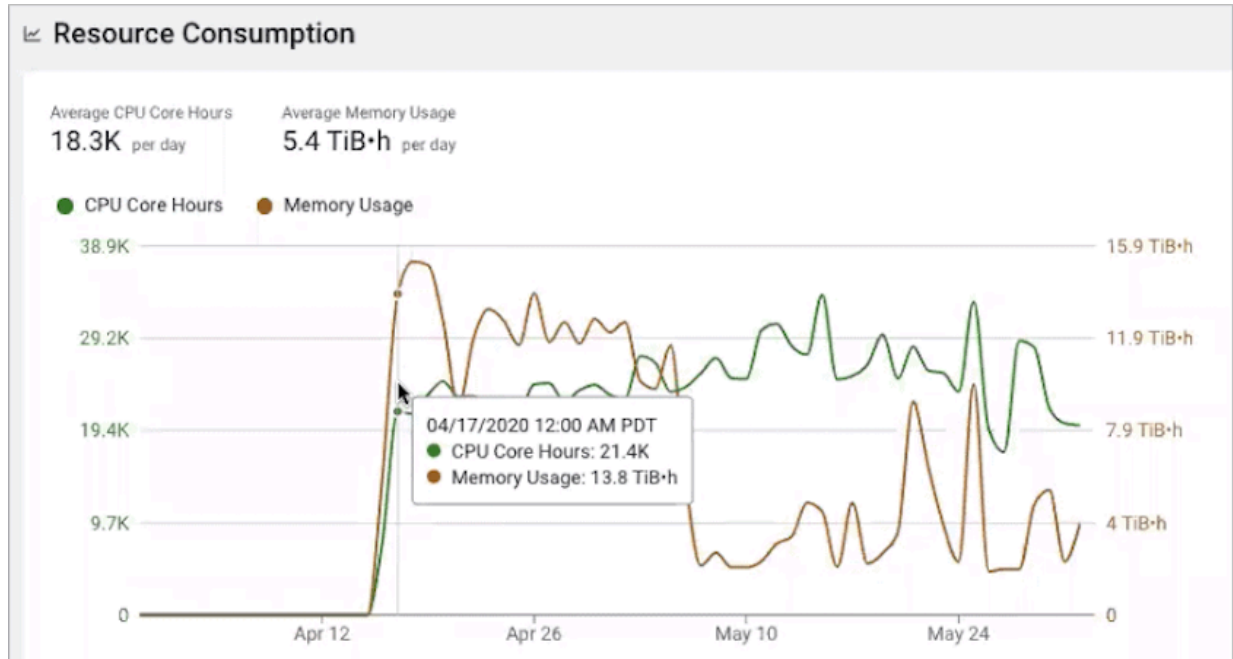
Identify and address abnormal or degraded performance problems

Cloudera Observability On-Premises enables you to identify and address abnormal or degraded performance problems by establishing baselines from health issues that also enable a performance comparison of your workloads. The Cloudera Observability On-Premises baseline metrics measure the current performance of a job against the average performance of previous runs. They use performance data from 30 of the most recent runs of a job and require a minimum of three runs. The baseline comparisons start with the fourth run of a job.

Location: The baseline for a job or a query is found on the Baseline page, which is accessed by selecting the environment, followed by the cluster and the then engine for analysis, then depending on the engine, clicking the Total Jobs or Total Queries in the Job Trend chart widget, selecting the job or query of interest, and then selecting the Baseline tab.

Note: Cloudera Observability On-Premises requires at least four job runs in order to create the job or query's baseline.

The following images show a few of the baseline and comparison metrics that are provided:

- This image shows the comparison between the baseline performance metrics and the current job run:

- To display only those metrics with performance issues, select Show only abnormal metrics:

Identify performance trends

You can identify trends as well as baselines by analyzing your engine's or cluster's performance trends from the Trends chart widget and the Trend tab. Where:

- The engine's job or query Trends time-series chart widget, displays more detailed metrics about the processed jobs and queries and enables you to view historical trends for analysis when you select a predefined or custom time period from the Time-Range filter list.

  Location: This chart widget is found on the Cluster Summary and the engine Summary pages, which are accessed by selecting the environment and then the cluster for analysis or by selecting the environment, followed by the cluster, and then the engine for analysis.

- The Trends tab, displays the job or query's instances executed during the selected time period. Depending on the engine, the Trends page displays a job's historical trend from Duration, Data Input, and Data Output histogram charts or lists the runs of the query to show how its performance changes overtime.

  Location: The Trends tab is found on the Jobs or Query's page, which is accessed by selecting the environment, followed by the cluster and then the engine for analysis, then depending on the engine, clicking the Total Jobs or Total Queries in the Job Trend chart widget, selecting the job or query of interest, and then selecting the Trends tab.
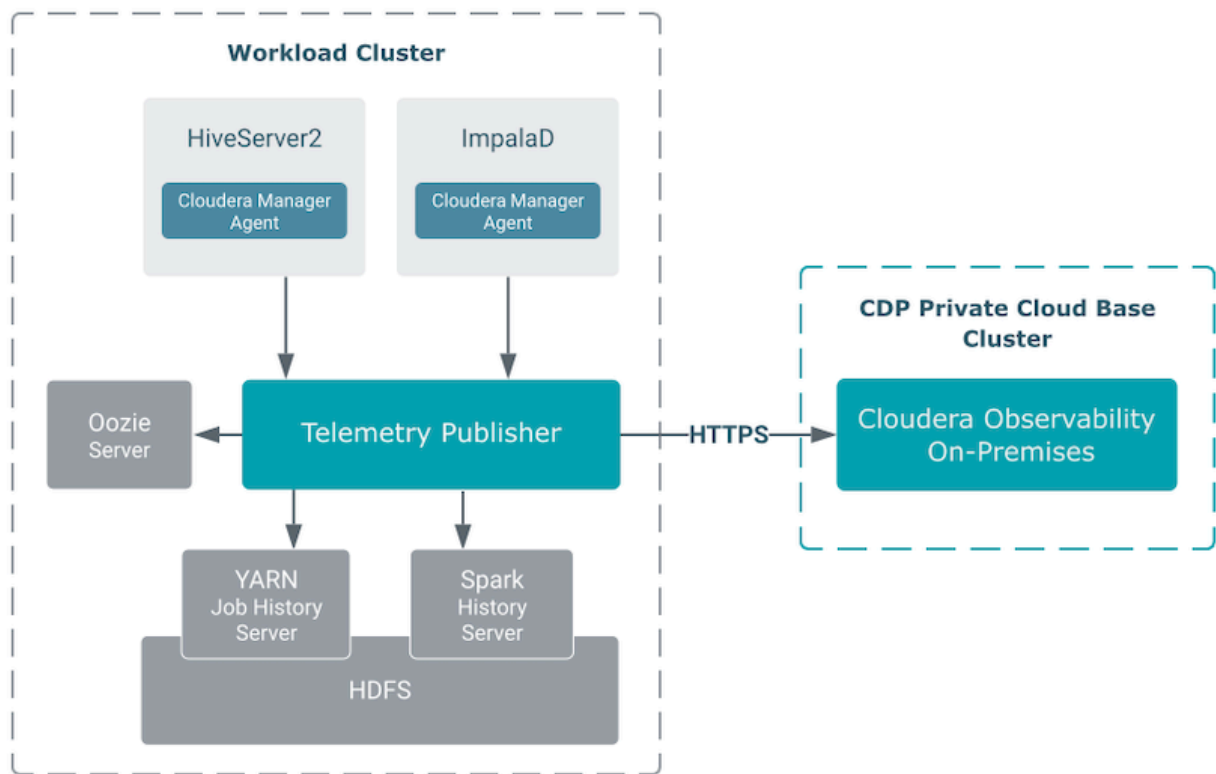
Collecting Cloudera Observability On-Premises diagnostic metrics

When you enable the Telemetry Publisher service for Cloudera Observability On-Premises, the Cloudera Management Service starts the Telemetry Publisher role. Cloudera Telemetry Publisher collects and transmits metrics, as well as configuration and log files, from Impala, Oozie, Hive, YARN, and Spark services for jobs running on your clusters to Cloudera Observability On-Premises. Telemetry Publisher collects metrics for all the clusters that use Cloudera Observability On-Premises-enabled environments.

Understanding the sources of information sent to Cloudera Observability On-Premises and how that data is redacted is described in the following topics.

Metric sources sent to Cloudera Observability On-Premises

Describes the resources from which you can configure Telemetry Publisher to collect diagnostic metrics.

Telemetry Publisher collects and transmits metrics as well as configuration and log files from Impala, Oozie, Hive, YARN, and Spark services for jobs running on your clusters to Cloudera Observability On-Premises, as shown in the above diagram. The metrics are collected as follows:

- Pull — Telemetry Publisher pulls diagnostic metrics from Oozie, YARN, and Spark periodically (by default, once per minute).
- Push — A Cloudera Manager Agent pushes diagnostic data from Hive and Impala to Telemetry Publisher within 5 seconds after a job finishes.

After the diagnostic data reaches Telemetry Publisher, it is stored temporarily in its data directory and periodically (once per minute) exported to Cloudera Observability On-Premises.

Diagnostic metrics collection details

Describes the type of data provided by the Cloudera data services and collected by Telemetry Publisher.

Telemetry Publisher collects and sends the following diagnostic metrics and data to Cloudera Observability On-Premises:

- Cloudera Manager Metrics — The Telemetry Publisher agent pulls a subset of Cloudera Manager metrics from the Cloudera Manager API endpoint in Private Cloud Base clusters and sends them to Cloudera Observability On-Premises. For more information, click the Related Information link below.
- Cloudera Manager Events — The Telemetry Publisher agent pulls the Cloudera Manager Events from the Cloudera Manager API endpoint in Private Cloud Base clusters and sends them to Cloudera Observability On-Premises. For more information, click the Related Information link below.
- Hive MetaStore (HMS) data source — Telemetry Publisher polls Hive and Impala for HMS metadata about your tables and their database and sends the details to Cloudera Observability On-Premises. This data includes the table's schema, database location, partitions, structure and relationships, columns, column names and their data types, and the table's metadata properties that include user-defined and predefined key-value pairs.

> **Important:** The Hive Metastore (HMS) must be deployed in the Workload cluster.

- Hive Queries — The Cloudera Manager agent periodically searches for query detail files that are generated by HiveServer2 after a query completes, and then sends the details from those files to Telemetry Publisher.

> **Important:** Hive query audits must be enabled.

- Impala Queries — A Cloudera Manager agent periodically looks for query profiles of recently completed queries and sends them to Telemetry Publisher.
- MapReduce Jobs — Telemetry Publisher polls the YARN Job History Server for recently completed MapReduce jobs. For each of these jobs, Telemetry Publisher collects the configuration and jhist file, which is the job history file that contains job and task counters, from HDFS. Telemetry Publisher can be configured to collect MapReduce task logs from HDFS and send them to Cloudera Observability On-Premises. By default, this log collection is turned off.
- Oozie Workflows — Telemetry Publisher polls Oozie servers for recently completed Oozie workflows and sends the details to Cloudera Observability On-Premises.
- Spark Applications — Telemetry Publisher polls the Spark History Server for recently completed Spark applications. For each of these applications, Telemetry Publisher collects the event log from HDFS. You can configure Telemetry Publisher to collect the executor logs of Spark applications from HDFS and send them to Cloudera Observability On-Premises. By default, this data collection is turned off.

> **Important:**
>
> To collect Spark application data from Apache Spark 3.x versions, Telemetry Publisher requires Cloudera Manager version 7.11.0 or above.
>
> For CDH 5.x clusters, Telemetry Publisher requires the CDS 2.2 Powered by Apache Spark release 2 or later, which is packaged with Apache Spark version 1.6.

Related Information

Cloudera Manager Metrics

Cloudera Manager Event Schema Reference

Redaction capabilities for diagnostic data

Describes the resources that you can configure for redaction. Cloudera recommends enabling redaction even if you are not sending diagnostic data to Telemetry Publisher.

The diagnostic data collected by Telemetry Publisher may contain sensitive information in job configuration or log files. The following lists the data and resources that you can configure for redacting sensitive data before it is sent to Telemetry Publisher:

- Log and query redaction — You can redact information in logs and queries collected by Telemetry Publisher based on filters created with regular expressions.
- MapReduce job properties redaction — You can redact job configuration properties before they are stored in HDFS. Since Telemetry Publisher reads the job configuration files from HDFS, it only fetches redacted configuration information.
- Spark event and executor log redaction — The spark.redaction.regex configuration property is used to redact sensitive data from event and executor logs in your YARN service. When this configuration property is enabled, Telemetry Publisher sends only redaction data to Cloudera Observability On-Premises. By default, this configuration property is enabled, but it can be overridden by using the advanced configuration snippet in Cloudera Manager or in the Spark application itself.

Related Information

Redacting data