# Cloudera Observability Release Notes

**Date published: 2026-05-15**
**Date modified:**

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

# Legal Notice

# Contents

# Release Summary

Cloudera Observability on premises is a single pane of glass observability solution, continually discovering and collecting performance telemetry across data, applications, and infrastructure components running in your on premises deployment. It enables you to interactively explore and understand your existing environments, workloads, clusters, and resources running in your on premises environment.

With advanced intelligent analytics and correlations, it provides insights and easy to follow recommendations that reduce time to resolution of complex issues, help manage and optimize costs, and improve performance. Cloudera Observability on premises also supports better financial governance by tracking and reporting on the costs associated with your business' cost centers.

Cloudera Observability on premises helps Administrators and Developers to:

- Watch and protect against budget overruns with its financial governance capabilities, allowing you to define cost centers and chargeback reports.
- Keep workloads healthy with active system monitoring, so you not only know what's going on right now, you'll be comparing to previous trends and historical analysis, to predict issues before they happen, receive alerts to take actions, and get automatic mitigations when possible.
- Improve performance with automations that help things run as best as they can, helping you optimize resource utilization and improve performance. With recommendations, you'll get insights into how to tune, and with custom automatic actions, Cloudera can be configured to auto-tune, your way.
- Maintain end to end health by identifying and eliminating bottlenecks that impact performance, while also ensuring your entire system, from infrastructure to platform, and workload, is healthy and optimized.
- Get actionable insights through self-service analytics, putting easy to use visualizations into everyone's hands.

Cloudera Observability on premises collects and visualizes a wide range of metrics and health tests, enabling you to do the following:

- Gain insights on current and completed workload jobs and queries, resource consumption, and system performance from a wide range of metrics.
- Identify bottlenecks, performance, and resource health issues from a wide range of health tests.
- Address performance issues with performance tuning and prescriptive guidance and recommendations.
- Gain visibility into the workload resource costs of your environment's infrastructure with the Financial Governance Chargeback feature.
- Define workload thresholds and consumption rules, create actions and alerts, and securely control user access, with the Workload Views and Access Management features.

You can integrate Cloudera Data Warehouse and Cloudera Data Engineering with Cloudera Observability. This integration supports limited features to collect query and job-level telemetry. These integrations allow you to use performance dashboards to optimize Apache Hive, Apache Impala, and Apache Spark workloads while ensuring SLA compliance. Additionally, Cloudera Observability includes Spark resource recommendations that provide prescriptive configuration values to improve application efficiency.

This release also introduces the Financial Governance Budgeting feature, which helps you manage infrastructure expenses and control spending. You can now use criteria and trigger logic when you create auto actions to automate workload management.

### Related Information

Triggering actions across jobs and queries

Analyzing your environment costs with Cloudera Observability cost centers

Classifying workloads for analysis with Workload Views

Assigning access roles in Cloudera Observability on premises

Hive, MapReduce, Oozie, and Spark Health Checks

Impala Health Checks

# What's new in Cloudera Observability on premises 3.7.1

Review the new features and functionality improvements in this release of Cloudera Observability on premises.

### Monitoring Cloudera Data Warehouse with Cloudera Observability on premises

Integrate Cloudera Data Warehouse with Cloudera Observability on premises to collect query and job-level telemetry. Use the dashboards to optimize Hive and Impala queries, monitor performance, and ensure SLA compliance. For more information, see the Monitoring Cloudera Data Warehouse with Cloudera Observability documentation.

### Monitoring Cloudera Data Engineering with Cloudera Observability on premises

Integrate Cloudera Data Engineering with Cloudera Observability on premises to collect Spark job telemetry and monitor resource usage. Use job-level metrics and performance dashboards to optimize Spark workloads and troubleshoot issues. You can also access historical analysis directly from the Cloudera Data Engineering job runs page. For more information, see the Monitoring Cloudera Data Engineering with Cloudera Observability documentation.

### Spark resource recommendations

This feature offers prescriptive configuration values for optimizing Spark application resource allocation, thereby improving efficiency with actionable suggestions for optimal Spark settings. For more information, see the Spark resource recommendations documentation.

### Hive statement types

Cloudera Observability on premises now supports a list of SQL statement types for workloads that use Apache Hive. For more information, see the Hive statement types documentation.

### Financial Governance Budgeting feature

Financial Governance Budgeting allows organizations to control spending by allocating and managing budgets for workload and infrastructure expenses. For more information, see the Financial Governance Budgeting feature documentation

### Criteria and Trigger logic

When you create an auto action, you can now set a criteria and trigger logic conditioning. For more information, see the Creating an Auto Action event documentation.

# What's new in Platform Support

You must be aware of the platform support for the Cloudera Observability on premises 3.7.1 release.

### Cloudera Data Services on premises

- 1.5.5 Service Pack 3

### Cloudera Base on premises

- 7.3.1
- 7.3.2

### Cloudera Manager

- 7.13.1
- 7.13.2

# Known Issues

This section lists known issues and workarounds in Cloudera Observability on premises 3.7.1.

**SSL handshake failures**

In Cloudera Data Services on premises environments, telemetry agent pods, such as database producer in Cloudera Data Warehouse or dbus-wxml-client in Cloudera Data Engineering can encounter an SSLHandshakeException when communicating with the DBus API. The pod logs display a PKIX path building failed error.

**Cloudera Data Warehouse**

1. Download the fix_cdw_ssl.py script
2. Export the kubeconfig env variable. Run the command `export KUBECONFIG=/path/to/kube.yaml`
3. Within the Kubernetes cluster, run the command `python3 fix_cdw_ssl.py --namespace [***NAMESPACE***]`. For example, `python3 fix_cdw_ssl.py --namespace warehouse-xodjvg-env`

The script downloads the DBus API SSL certificate and adds it to the databus-producer truststore.

**Cloudera Data Engineering**

1. Download the fix_cde_ssl.py script
2. Export the kubeconfig env variable. Run the command `export KUBECONFIG=/path/to/kube.yaml`
3. Within the Kubernetes cluster, run the command `python3 fix_cde_ssl.py --namespace [***NAMESPACE***]`. For example, `python3 fix_cde_ssl.py --namespace dex-base-pc74hb7x`

The script downloads the DBus API SSL certificate and adds it to the databus-producer truststore.

**Impala does not support super user configuration for the observability user for Apache Ranger-enabled cluster**

The observability user requires full privileges on the Observability cluster. Required services such as Kafka, HDFS, HBase, and Hive support super user setup by specifying the ranger.plugin.[service].super.users property to observability. However, this super user setup is not supported for Impala.

Manually add a new user named observability in Apache Ranger and assign full privileges.

- For information, see *Adding a user* in Cloudera Base on premises documentation.
- For information on granting user access using Apache Ranger, see *Impala Authorization* in Cloudera Data Warehouse Runtime for Cloudera on premises documentation.

**Exporting of Impala queries fail for Telemetry Publisher with Cloudera Manager 7.11.3**

Telemetry Publisher for Impala queries does not work with Cloudera Manager 7.11.3

Upgrade Cloudera Manager from 7.11.3 to 7.11.3 cumulative hotfix 6 (CHF6) version to successfully export Impala queries.

**Auto Action trigger for Impala Engine**

Impala Auto Action triggers do not work for the Kerberos-enabled Private Cloud base cluster running on Cloudera Manager 7.9.5 and 7.11.3.

Upgrade Cloudera Manager to 7.11.3 cumulative hotfix 9 (CHF9) version.

**Telemetry publisher test altus connection fails for Cloudera Manager 7.11.3 hotfix (CHF6, 7, and 8) versions**

Test connection fails with the following error:

```
Exception in thread "main" java.lang.NoSuchMethodError: 'com.goo
gle.common.collect.ImmutableSet com.google.common.collect.Immuta
bleSet.copyOf(java.util.Collection)'
 at com.cloudera.cdp.http.HttpCodesRetryChecker.<init>(HttpCode
sRetryChecker.java:57)
 at com.cloudera.cdp.client.CdpClientConfigurationBuilder.<init>(
CdpClientConfigurationBuilder.java:53)
 at com.cloudera.cdp.client.CdpClientConfigurationBuilder.default
Builder(CdpClientConfigurationBuilder.java:400)
 at com.cloudera.cdx.client.TestDatabusConnection.main(TestDatab
usConnection.java:55)
```

This issue only affects the test connection method.

Upgrade Cloudera Manager to 7.11.3 cumulative hotfix 9 (CHF9) version, and then start Telemetry Publisher.

**Limitations of Cloudera Observability Spark processing**

Spark event log processing is skipped under the following conditions:

- The event log size exceeds 20 GB.
- The number of Spark jobs in the Spark application exceeds 500.
- The number of SQL executions in the Spark application exceeds 500.
- When spark.history.fs.eventLog.rolling.maxFilesToRetain is set, Spark keeps only recent event log files as configured inspark.history.fs.eventLog.rolling.maxFilesToRetain and older event log files are compacted (in a file with .compact extension). Spark event log processing of such Spark applications are skipped by Cloudera Observability.

## Related Information

Adding a user

Impala Authorization