

Cloudera Runtime 7.3.2

## Troubleshooting Apache Atlas

Date published: 2020-07-28

Date modified: 2026-03-31

# CLOUDERA

<https://docs.cloudera.com/>

# Legal Notice

© Cloudera Inc. 2026. 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

<b>Atlas index repair configuration.....</b>	<b>4</b>
<b>Activating Concurrent Message Ingestion.....</b>	<b>5</b>
<b>Using Zookeeper in SSL mode.....</b>	<b>5</b>
<b>Real-time update of entity details.....</b>	<b>7</b>

## Atlas index repair configuration

You can use reindexing to troubleshoot Apache Atlas basic search inconsistency.

### Rebuilding the whole Atlas index

In your Cloudera Manager instance running the Atlas service, add the following in Atlas Server Advanced Configuration Snippet (Safety Valve) for conf/atlas-application.properties.

```
atlas.rebuild.index=true
```

```
atlas.patch.numWorkers=3
```

```
atlas.patch.batchSize=300
```

Later, restart the Atlas Service.



#### Attention:

- You must revert back this configuration once the reindexing is completed, else the reindexing takes place on every restart.
- The reindexing process will be done during Atlas restart, so Atlas will not be reachable till reindexing process is completed.
- The time taken for reindexing depends upon the amount of data.

### Rebuilding the index for particular GUID

Incorrect search results related to a particular GUID can be repaired by limiting the reindex to that element.

```
atlas-index-repair/repair_index.py [-g <***GUID***>]
```



#### Note:

Atlas will use REST APIs to fetch the entity, which will need the correct authentication mechanism to be specified based on the installation.

For an Atlas installation with username and password use the following:

```
atlas-index-repair/repair_index.py [-g <***GUID***>] [-u <***USER***>] [-p <***PASSWORD***>] *
guid: [optional]
```

Example:

```
atlas-index-repair/repair_index.py -u admin -p admin123 -g 13d77457-2a45-4e92-ad53-a172c7cb70a5
```

For Atlas installations using Kerberos as authentication mode, use the following:

```
kinit -kt /etc/security/keytabs/atlas.service.keytab atlas/fqdn@DOMAIN
```

Example:

```
kinit -kt /etc/security/keytabs/atlas.service.keytab atlas/fqdn@EXAMPLE.com
```

```
atlas-index-repair/repair_index.py -g 13d77457-2a45-4e92-ad53-a172c7cb70a5
```



**Note:** In case of many affected entities, it is recommended to rebuild the whole index instead.

## Activating Concurrent Message Ingestion

Atlas hook message processing can take a long time with a high number of messages, greatly increasing wait times. This can be improved by turning on Concurrent Message Ingestion.

Atlas Hook message processing has linear complexity for consuming the messages. This can lead to the following:

- Enforcing authorization policies takes longer.
- Metadata showing up in Atlas takes an unpredictable amount of time.

Concurrent Message Ingestion can offset these issues by enabling the following after determining dependencies within incoming messages:

- Dependent messages are processed serially.
- Messages without dependencies are processed concurrently.

### Enabling Concurrent Message Ingestion

1. Go to Cloudera Manager Clusters Atlas Configuration .
2. Add the following property `atlas.notifications.concurrent=true`.

## Using Zookeeper in SSL mode

Using Apache Zookeeper in SSL mode requires all clients to be in SSL mode.

### About this task

Missing to activate the SSL mode for any of the Zookeeper clients can lead to the Apache Atlas user interface being inaccessible with an HTTP 503 error.

### Procedure

1. Go to Cloudera Manager Clusters HBase Configuration .
2. Search for HBase ZooKeeper Secure Client Enabled.

The screenshot shows the Cloudera Manager interface for Cluster 1. The left sidebar contains navigation options like Clusters, Hosts, Diagnostics, Audits, Charts, Replication, Administration, Data Services, Parcels, Running Commands, Support, and a user profile for 'admin'. The main content area is titled 'Cluster 1' and shows the configuration for 'HBASE-1'. The search bar contains 'HBase ZooKeeper Secure Client Enabled'. The configuration table shows a single entry for 'HBase ZooKeeper Secure Client Enabled' with a checkbox for 'HBASE-1 (Service-Wide)' that is currently unchecked. The table also lists properties: 'hbase.zookeeper.property.client.secure' and 'zookeeper.secure\_client\_enabled'. The left sidebar has a 'Filters' section with categories like SCOPE, CATEGORY, and STATUS.

3. Select the checkbox.
4. Save your changes.
5. Go Cloudera Manager Clusters Solr Configuration
6. Search for Enable TLS/SSL for Solr.

The screenshot shows the Cloudera Manager interface for Cluster 1. The left sidebar is the same as in the previous screenshot. The main content area is titled 'Cluster 1' and shows the configuration for 'SOLR-1'. The search bar contains 'Enable TLS/SSL for Solr'. The configuration table shows a single entry for 'Enable TLS/SSL for Solr' with a checkbox for 'SOLR-1 (Service-Wide)' that is checked. The table also lists the property 'solr\_use\_ssl'. The left sidebar has a 'Filters' section with categories like SCOPE, CATEGORY, and STATUS.

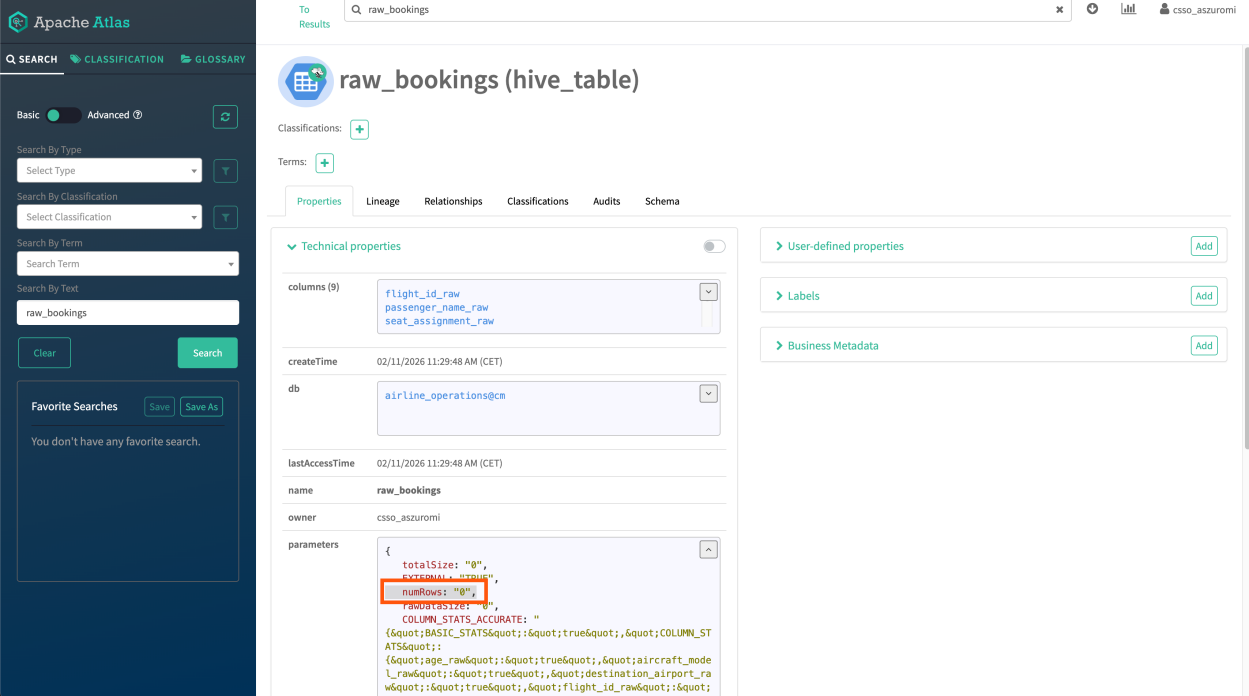
7. Select the checkbox.
8. Save your changes.

## Real-time update of entity details

Atlas cannot update row numbers for entities unless a Data Definition Language operation is made on the entity.

Atlas can only monitor events caused by the DDL commands. This means that Atlas is not updated when Data Manipulation Language (DML) operations (SELECT/INSERT/UPDATE/DELETE) cause changes in Hive. Details not updated include also the operational properties, for example, **lastAccessTime**, **last\_modified\_by**, **last\_modified\_time**. The displayed details in Atlas (for example, number of rows) can be triggered by Data Definition Language (DDL) operations (CREATE/ALTER/DROP).

**Figure 1: Hive table without any rows in Atlas**



The screenshot displays the Apache Atlas web interface for a Hive table named 'raw\_bookings'. The interface is divided into several sections:

- Search Panel (Left):** Includes search filters for Type, Classification, Term, and Text. The search term 'raw\_bookings' is entered.
- Table Header:** Shows the table name 'raw\_bookings (hive\_table)' and its classification.
- Properties Tab:** The 'Technical properties' section is expanded, showing:
  - columns (9): flight\_id\_raw, passenger\_name\_raw, seat\_assignment\_raw
  - createTime: 02/11/2026 11:29:48 AM (CET)
  - db: airline\_operations@cm
  - lastAccessTime: 02/11/2026 11:29:48 AM (CET)
  - name: raw\_bookings
  - owner: csso\_aszuromi
  - parameters: A JSON object containing 'totalSize: "0"', 'numRows: "0"', and 'numPartitions: "0"'. The 'numRows: "0"' value is highlighted with a red box.
- Right Panel:** Contains expandable sections for 'User-defined properties', 'Labels', and 'Business Metadata', each with an 'Add' button.

You can check the number of rows in Hue with the **Table Browser**.

Table Browser

Databases > airline\_operations > raw\_bookings

Overview Sample (20) Details

**PROPERTIES**

Table

External and stored in location

Created by csso\_aszuromi on 11/02/2026 11:29 +01:00

**STATS**

Files 1 **Rows 20** Total size 1.5 KB

Data last updated on 11/02/2026 11:29 +01:00

**SCHEMA**

Column (9)	Type	Description	Sample
flight_id_raw	string		BUD-JFK-20250828
passenger_name_raw	string		John Smith
seat_assignment_raw	string		14A
aircraft_model_raw	string		Boeing 787-9
origin_airport_raw	string		BUD
destination_airport_raw	string		JFK
age_raw	int		45
ticket_price_usd_raw	decimal...		1250.75
passport_number_raw	string		583472910

You can also check the number of rows by using the COUNT query.

Hive

5.74s airline\_operations

```
SELECT COUNT(*) FROM airline_operations.raw_bookings;
```

INFO : INPUT\_FILES\_Map\_1: 1  
INFO : RAW\_INPUT\_SPLITS\_Map\_1: 1  
INFO : Completed executing command(queryId=hive\_20260217131857\_4fb655da-66ab-44ac-9242-e287551b5c39); Time taken: 4.798 seconds  
INFO : OK

Query History Saved Queries Results (1)

COLUMNS (1) Q	_c0
_c0 bigint	20

Tables

- airline\_operations.raw\_bookings
  - flight\_id\_raw string
  - passenger\_name\_raw string
  - seat\_assignment\_raw string
  - aircraft\_model\_raw string
  - origin\_airport\_raw string
  - destination\_airport\_raw string
  - age\_raw int
  - ticket\_price\_usd\_raw decimal(10,2)
  - passport\_number\_raw string



**Note:** Running an ALTER operation on the specific table, for example adding new column, triggers the recalculation of rows in Atlas' parameters field.