

Cloudera Runtime 7.0.3

Cloudera Search FAQ

Date published: 2019-11-19

Date modified:

CLOUdera

<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

Cloudera Search Frequently Asked Questions.....	4
General.....	4
What is Cloudera Search?.....	4
What is the difference between Lucene and Solr?.....	4
What is Apache Tika?.....	4
How does Cloudera Search relate to web search?.....	4
How does Cloudera Search relate to enterprise search?.....	4
How does Cloudera Search relate to custom search applications?.....	4
How can I use the Search platform? Is it exclusively available through Hue?.....	4
Does Search support querying and aggregating results across Solr collections? If data is separated into different collections, does Search support querying across those collections and aggregating results?.....	5
Which Solr Server should I send my queries to?.....	5
How does Search handle querying different document types? For example, how does Search evaluate a query against both text and images?.....	5
Do Search security features use Kerberos?.....	5
Can I restrict access to collections?.....	5
Do I need to configure Ranger restrictions for each access mode, such as for the admin console and for the command line?.....	5
Does Search support indexing data stored in JSON files and objects?.....	5
How can I set up Cloudera Search so that results include links back to the source that contains the result?.....	6
Why do I get an error “no field name specified in query and no default specified via 'df' param” when I query a Schemaless collection?.....	6
Performance and Fail Over.....	6
How large of an index does Cloudera Search support per search server?.....	6
What is the response time latency I can expect?.....	6
How does Cloudera Search performance compare to Apache Solr?.....	7
What hardware or configuration changes can I make to improve Search performance?.....	7
Where should I deploy Solr Servers?.....	7
What happens if a host running a Solr Server process fails?.....	7
How is performance affected if the Solr Server is running on a node with no local data?.....	7
How can I redistribute shards across a cluster?.....	7
How do I manage resources for Cloudera Search and other components on the same cluster?.....	7
Schema Management.....	7
If my schema changes, will I need to re-index all of my data and files?.....	7
Can I extract fields based on regular expressions or rules?.....	8
Can I use nested schemas?.....	8
What is Apache Avro and how can I use an Avro schema for more flexible schema evolution?.....	8
Supportability.....	8
Does Cloudera Search support multiple languages?.....	8
Which file formats does Cloudera Search support for indexing? Does it support searching images?.....	8

Cloudera Search Frequently Asked Questions

This section includes answers to questions commonly asked about Search for CDH. Questions are divided into the following categories:

General

The following are general questions about Cloudera Search and the answers to those questions.

What is Cloudera Search?

Cloudera Search is Apache Solr integrated with CDH, including Apache Lucene, Apache SolrCloud, Apache Tika, and Apache Hadoop MapReduce and HDFS. Cloudera Search also includes valuable integrations that make searching more scalable, easy to use, and optimized for both near-real-time and batch-oriented indexing. These integrations include Cloudera Morphlines, a customizable transformation chain that simplifies loading any type of data into Cloudera Search.

What is the difference between Lucene and Solr?

Lucene is a low-level search library that is accessed by a Java API. Solr is a search server that runs in a servlet container and provides structure and convenience around the underlying Lucene library.

What is Apache Tika?

The Apache Tika toolkit detects and extracts metadata and structured text content from various documents using existing parser libraries.

How does Cloudera Search relate to web search?

Traditional web search engines crawl web pages on the Internet for content to index. Cloudera Search indexes files and data that are stored in HDFS and HBase. To make web data available through Cloudera Search, it needs to be downloaded and stored in [Cloudera Enterprise](#).

How does Cloudera Search relate to enterprise search?

Enterprise search connects with different backends (such as RDBMS and filesystems) and indexes data in all those systems. Cloudera Search is intended as a full-text search capability for data in CDH. Cloudera Search is a tool added to the Cloudera data processing platform and does not aim to be a stand-alone search solution, but rather a user-friendly interface to explore data in Hadoop and HBase.

How does Cloudera Search relate to custom search applications?

Custom and specialized search applications are an excellent complement to the Cloudera data-processing platform. Cloudera Search is not designed to be a custom application for niche vertical markets. However, Cloudera Search does include a simple search GUI through a plug-in application for Hue. The Hue plug-in application is based on the Solr API and allows for easy exploration, along with all of the other Hadoop frontend applications in Hue.

How can I use the Search platform? Is it exclusively available through Hue?

Search supports any application that uses standard Solr APIs. You can build custom applications using the Solr API to address specific Search use-cases.

You can also use the Hue Search application.

Does Search support querying and aggregating results across Solr collections? If data is separated into different collections, does Search support querying across those collections and aggregating results?

Search does not currently support automatic querying across collections, but you can create collection aliases that include mult

Which Solr Server should I send my queries to?

Any Solr Server can accept and process client connections.

How does Search handle querying different document types? For example, how does Search evaluate a query against both text and images?

It is possible to query disparate documents, but it is often most effective and powerful to query a collection of documents that contains documents of similar types.

Before you can query a dataset, you must first make it available to Cloudera Search. This dataset is organized as a collection, and Cloudera Search indexes the documents according to a specified schema. For example, a collection of emails might use a schema that includes date sent, sender, recipient, subject, size, and body. A collection of images might use a schema that includes date created, creator, subject, size, and geolocation.

Cloudera Search supports two ways of organizing the example of the emails and images:

- Separate collections - One collection for emails and one collection for images is the most flexible solution. In this case, each collection can be queried and ranked using all attributes because each collection's index applies to all documents in each collection. It is possible to get very specific results and rankings for detailed queries, but only the items in queried collection are returned in the results.
- Unified collection - A single collection for both emails and images is the most inclusive, wide-ranging solution. You can query all documents at once, but you can only search and filter common attributes such as size and date. Querying emails for geolocation values or querying images for sender values does not make sense.

Do Search security features use Kerberos?

Yes, Cloudera Search includes support for Kerberos authentication. Search continues to use simple authentication with the anonymous user as the default configuration, but Search now supports changing the authentication scheme to Kerberos. All required packages are installed during the installation or upgrade process. Additional configuration is required before Kerberos is available in your environment.

Can I restrict access to collections?

Yes, Cloudera Search supports Apache Ranger for authorization. For more information, see [Using Ranger to Provide Authorization in CDP](#).

Do I need to configure Ranger restrictions for each access mode, such as for the admin console and for the command line?

Ranger restrictions are consistently applied regardless of the way users attempt to complete actions. For example, restricting access to data in a collection consistently restricts that access, whether queries come from the command line, from a browser, or through the admin console.

Does Search support indexing data stored in JSON files and objects?

Yes, you can use the `readJson` and `extractJsonPaths` morphline commands that are included with the CDK to access JSON data and files. For more information, see [cdk-morphlines-json](#).

How can I set up Cloudera Search so that results include links back to the source that contains the result?

You can use stored results fields to create links back to source documents. For information on data types, including the option to set results fields as stored, see the Solr Wiki page on [SchemaXml](#).

For example, with MapReduceIndexerTool you can take advantage of fields such as file_path. See [MapReduceIndexerTool](#) for more information. The output from the MapReduceIndexerTool includes file path information that can be used to construct links to source documents.

If you use the Hue UI, you can link to data in HDFS by inserting links of the form:

```
<a href="/filebrowser/download/{{file_path}}?disposition=inline">Download</a>
```

Why do I get an error “no field name specified in query and no default specified via 'df' param” when I query a Schemaless collection?

Schemaless collections initially have no default or df setting. As a result, simple searches that might succeed on non-Schemaless collections may fail on Schemaless collections.

When a user submits a search, it must be clear which field Cloudera Search should query. A default field, or df, is often specified in solrconfig.xml, and when this is the case, users can submit queries that do not specify fields. In such situations, Solr uses the df value.

When a new collection is created in Schemaless mode, there are initially no fields defined, so no field can be chosen as the df field. As a result, when query request handlers do not specify a df, errors can result. This issue can be addressed in several ways:

- Queries can specify any valid field name on which to search. In such a case, no df is required.
- Queries can specify a default field using the df parameter. In such a case, the df is specified in the query.
- You can uncomment the df section of the generated schemaless solrconfig.xml file and set the df parameter to the desired field. In such a case, all subsequent queries can use the df field in solrconfig.xml if no field or df value is specified.

Performance and Fail Over

The following are questions about performance and fail over in Cloudera Search and the answers to those questions.

How large of an index does Cloudera Search support per search server?

This question includes too many variables to provide a single answer. Typically, a server can host from 10 to 300 million documents, with the underlying index as large as hundreds of gigabytes. To determine a reasonable maximum document quantity and index size for servers in your deployment, prototype with realistic data and queries.

What is the response time latency I can expect?

Many factors affect how quickly responses are returned. Some factors that contribute to latency include whether the system is also completing indexing, the type of fields you are searching, whether the search results require aggregation, and whether there are sufficient resources for your search services.

With appropriately-sized hardware, if the query results are found in memory, they may be returned within milliseconds. Conversely, complex queries requiring results aggregation over huge indexes may take a few seconds.

The time between when Search begins to work on indexing new data and when that data can be queried can be as short as a few seconds, depending on your configuration.

This high performance is supported by custom caching optimizations that Cloudera has added to the Solr/HDFS integration. These optimizations allow for rapid read and writes of files in HDFS, performing at or above the speeds of stand-alone Solr reading and writing from local disk.

How does Cloudera Search performance compare to Apache Solr?

Assuming the same number of documents, size, hardware, and so on, with similar memory cache available, query performance is comparable. This is due to caching implemented by Cloudera Search. After the cache is warmed, the code paths for both Apache Solr and Cloudera Search are the same. You might see some performance degradation with cache misses when Cloudera Search must read from disk (HDFS), but that is mitigated somewhat by the HDFS block cache.

What hardware or configuration changes can I make to improve Search performance?

Search performance can be constrained by CPU limits. If you're seeing bottlenecks, consider allocating more CPU to Search.

Where should I deploy Solr Servers?

Cloudera recommends running them on DataNodes for locality. You can run a Solr Server on each DataNode, or a subset of DataNodes. Do not run Solr on any master nodes, such as NameNodes. Solr does not have a master server process. All Solr servers are the same.

What happens if a host running a Solr Server process fails?

Cloudera Search uses the HDFS client API. Read and write requests for HDFS blocks will be automatically redirected as appropriate. If the failed or decommissioned host runs a DataNode process, any HDFS blocks on that host are re-replicated according to your HDFS configuration.

How is performance affected if the Solr Server is running on a node with no local data?

Because of the caching capabilities in Solr, performance impact is negligible.

How can I redistribute shards across a cluster?

You can move shards between hosts using the process described in [Migrating Solr Replicas](#).

How do I manage resources for Cloudera Search and other components on the same cluster?

You can use cgroups to allocate resources among your cluster components.

Schema Management

The following are questions about schema management in Cloudera Search and the answers to those questions.

If my schema changes, will I need to re-index all of my data and files?

When you change the schema, Cloudera recommends re-indexing. For example, if you add a new field to the index, apply the new field to all index entries through re-indexing. Re-indexing is required in such a case because existing documents do not yet have the field. Cloudera Search includes a MapReduce batch-indexing solution for re-indexing and a GoLive feature that assures updated indexes are dynamically served.

While you should typically re-index after adding a new field, this is not necessary if the new field applies only to new documents or data. This is because, were indexing to be completed, existing documents would still have no data for the field, making the effort unnecessary.

For schema changes that only apply to queries, re-indexing is not necessary.

Can I extract fields based on regular expressions or rules?

Cloudera Search supports limited regular expressions in Search queries. For details, see [Lucene Regular Expressions](#).

On data ingestion, Cloudera Search supports easy and powerful extraction of fields based on regular expressions. For example the grok morphline command supports field extraction using regular expressions.

Cloudera Search also includes support for rule directed ETL with an extensible rule engine, in the form of the tryRules morphline command.

Can I use nested schemas?

Cloudera Search supports nesting documents in this release. To learn about schemas with nested documents and their limitations, see [Indexing Nested Documents](#).

What is Apache Avro and how can I use an Avro schema for more flexible schema evolution?

To learn more about Avro and Avro schemas, see the [Avro Overview page](#) and the [Avro Specification page](#).

To see examples of how to implement inheritance, backwards compatibility, and polymorphism with Avro, see this [InfoQ article](#).

Supportability

The following are questions about supportability in Cloudera Search and the answers to those questions.

Does Cloudera Search support multiple languages?

Cloudera Search supports approximately 30 languages, including most Western European languages, as well as Chinese, Japanese, and Korean.

Which file formats does Cloudera Search support for indexing? Does it support searching images?

Cloudera Search uses the Apache Tika library for indexing many standard document formats. In addition, Cloudera Search supports indexing and searching Avro files and a wide variety of other file types such as log files, Hadoop Sequence Files, and CSV files. You can add support for indexing custom file formats using a morphline command plug-in.