

Cloudera Stream Processing Community Edition 2.0.0

Community Edition Overview

Date published: 2022-07-29

Date modified: 2023-01-31

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.

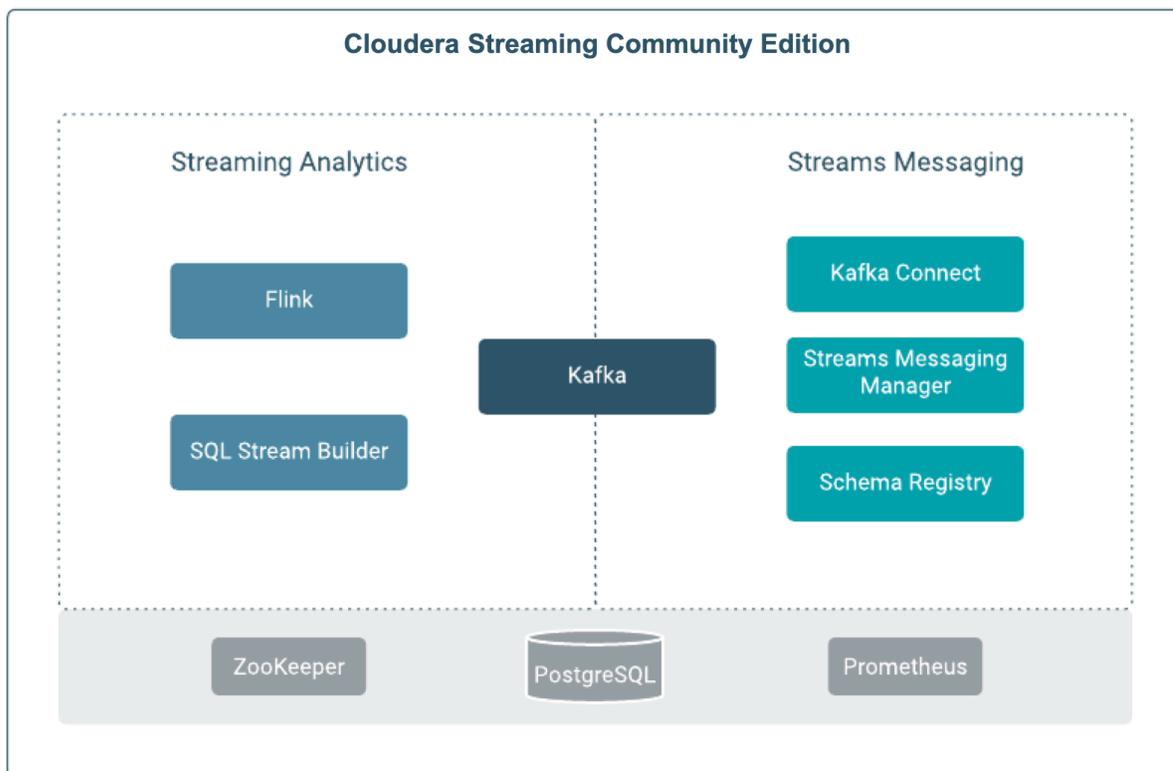
Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES 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

Introduction to Cloudera Streaming Community Edition.....	4
Cloudera Streaming as a Dockerized Application.....	5

Introduction to Cloudera Streaming Community Edition

Cloudera Streaming Community Edition is a standalone deployment of Streams Messaging and Cloudera Streaming Analytics. You can use this dockerized version to quickly set up, and try out real-time streams processors in your local environment using Apache Kafka, Schema Registry, Streams Messaging Manager, Kafka Connect, Apache Flink, and Cloudera SQL Stream Builder.



Apache Kafka

Apache Kafka supports millions of messages per second with low latency and high throughput, scaling elastically and transparently without downtime. It addresses a wide range of streaming data initiatives, enabling enterprises to keep up with customer demand, provide better services, and proactively manage risk.

Schema Registry

Schema Registry lets you manage, share, and support the evolution of all producer and customer schemas in a shared schema repository that allows applications to flexibly interact with each other across the Kafka landscape. It safely mitigates interruptions that occur due to schema mismatches.

Streams Messaging Manager

Streams Messaging Manager provides a single pane of glass view with end-to-end visibility into how data moves across Kafka clusters—among producers, brokers, topics, and consumers—allowing you to track data lineage and governance from edge to cloud. It also simplifies troubleshooting of Kafka environments with intelligent filtering and sorting.

Apache Flink

Apache Flink is a distributed processing engine and a scalable data analytics framework. You can use Flink to process data streams on a large scale to deliver real-time analytical insights.

Cloudera SQL Stream Builder

Cloudera SQL Stream Builder is a comprehensive interactive user interface for creating stateful stream processing jobs using SQL powered by Apache Flink. By using SQL, you can simply and easily declare expressions that filter, aggregate, route, and otherwise mutate streams of data. Cloudera SQL Stream Builder 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.

For more information about Cloudera Streaming as a product, and how Cloudera Streaming is implemented in Cloudera, see the [Cloudera Streaming product page](#).

Cloudera Streaming as a Dockerized Application

Cloudera Streaming Community Edition consists of preconfigured Docker images of the components. The Streams Messaging and Streaming Analytics components share Zookeeper and PostgreSQL. The components can be reached using their dedicated ports. Storage for Cloudera Streaming Community Edition is handled by docker volumes, while PostgreSQL is integrated for database management and storing metadata.

The containers use the following docker volumes to provide persistent local storage between restarts. If the volumes do not exist in your local environment, they are created when running the docker-compose up command.

kf-volume

Used by the Kafka container to store the topics.

When used with Streaming Analytics:

- The Kafka container is by default preconfigured in Cloudera SQL Stream Builder as the **Local Kafka** data provider.

kfc-volume

Used for Kafka Connect.

prom-volume

Used for Streams Messaging Manager Metrics.

sr-volume

Used for Schema Registry.

flink-volume

Persistent in the Flink TaskManager and JobManager containers. It is used for storing savepoints of the jobs. When using the Filesystem connector, it is also recommended to use this volume for file management.

ssb-volume

Used by the Cloudera SQL Stream Builder engine for persistent storage under the Streaming SQL Engine container.

pg-volume

Used by the PostgreSQL database.

When used with Streaming Analytics:

- It stores the internal tables required for Cloudera SQL Stream Builder to work, as well as the created Materialized Views.
- PostgreSQL is used by Cloudera SQL Stream Builder components internally. It is also used as the underlying database for the Materialized View Engine. The PostgreSQL database for the Materialized View tables (*EVENTADOR_SNAPPER* database) can be accessed by using the user *EVENTADOR_SNAPPER*. The default password for the database is *CLOUDERA*.

zk-volume

Used by ZooKeeper.

It is possible to delete the docker volumes for a fresh start by shutting down all of the containers with `docker-compose down --volumes` command, or individually removing them with `docker volume rm <VOLUME NAME>` command. The containers use a docker network (named `ssb-net`) to communicate.