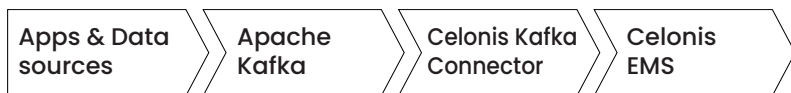


Kafka Connector

About

The new **Celonis Kafka Connector** allows you to ingest any data from Apache Kafka® topics into the Celonis Execution Management System (EMS). The connector is working with Kafka Connect and is driven by configuration, so you don't need to write a single line of code.



What is Kafka Connect?

Kafka Connect is a free, open-source component of Apache Kafka® to continuously import and export data as event streams to integrate with your existing systems. The Kafka Connect cluster is highly scalable and fault-tolerant to ensure continuous operations without any data loss.

What it means for you

Plugin your Kafka infrastructure

Any streaming data into EMS

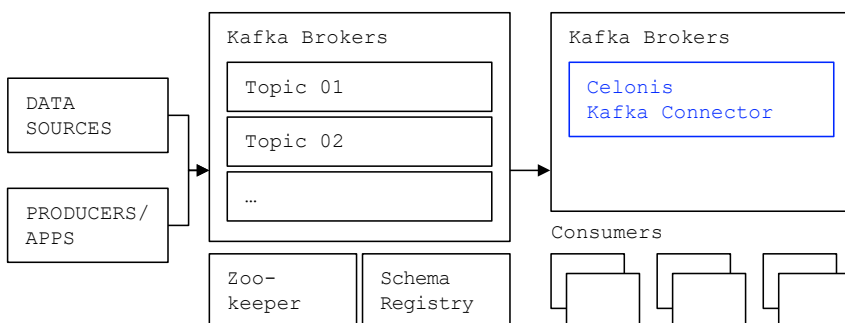
Support execution with timely data

The Celonis Kafka Connector enables Celonis customers to integrate streams of real-time, event-driven data into the Celonis Execution Management System (EMS), to support decision-making and execution, in the moment.

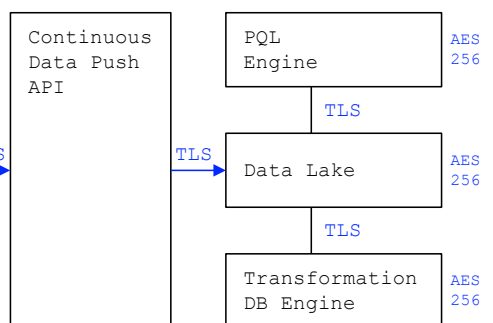
Architecture Overview



Customer Kafka Infrastructure



Celonis EMS



How it works



Connect to EMS

To establish the connection to the EMS you need to obtain the relevant endpoint for your realm and the API Authorization key from your EMS account.

EMS Tables

If the table does not exist in the EMS, it will be created with the first file being processed on the EMS side. It will use the Parquet file schema to create the table. If the connector is pushing to an existing table, you need to make sure that the schema is matching. In this version there is no validation done by the connector.

Supported data formats

Converters are necessary to have a Kafka Connect deployment support a particular data format when writing to or reading from Kafka. Connectors use converters to change the format of data from bytes to a Connect internal data format and vice versa.

Topics to EMS tables

You can spin up multiple instances of the Connector (sink). Each instance can read from one or more Kafka topics, and publish to a given target table in EMS. To send data to more than one target tables, you will need one connector per target.

Commit policy

There are 3 criteria to trigger to push data:
01 | Parquet file size
02 | Number of records in file
03 | Time since last write
Once a topic-partition record is written, the connector checks if any of the criteria is met and uploads the file.

Transform data

The Connector can be configured with Single Message Transformations to make lightweight modifications to individual messages as they flow through connect. SMTs allow users to change the data shape for example by adding or removing fields, or moving them between the key and the value.

EMS Continuous Data API

The EMS Continuous Data API expects parquet files. The connector accumulates files locally for each topic-partition, to reduce network use. Each file then holds records from the stream. Based on the policies the files are pushed to the API. An EMS job will then update the PQL.

Error handling

Kafka Connect includes error handling options and the ability to route messages to a dead letter queue in the case of a message cannot be processed. The Connector also support error policies while inserting data and logs errors accordingly.

Evolve schemas

When Kafka topics can evolve their schema the Connector rolls the accumulated file over and creates a new file with the new schema which will be uploaded to EMS. The Connector will not evolve the schema within EMS.

Quickstart

01 | Download the Celonis Sink Connector

Download the connector and unzip the files

02 | Add to connector to your Kafka Connect cluster

Add the files to the Kafka Connect cluster's plugins location

03 | Create a running instance

Use the Connect API to instantiate a connector with the appropriate configuration

Pre-requisites

Kafka >v. 2.3 | Running Kafka Connect cluster

Resources

celonis.com/lenses
celonis.com/kafka-connector

Contacts

<https://www.celonis.com/contact/>