

# Adding Connectors or Software

Confluent provides two images for Kafka Connect:

- The Kafka Connect Base image contains Kafka Connect and all of its dependencies. When started, it will run the Connect framework in distributed mode.
- The Kafka Connect image extends the Kafka Connect Base image and includes several of the connectors supported by Confluent: JDBC, Elasticsearch, HDFS, S3, and JMS.

There are currently two ways to add new connectors to these images.

- Use the `cp-kafka-connect` or `cp-kafka-connect-base` image as-is and [add the connector JARs via volumes](#)
- Build a new Docker image that has the new connectors installed. See the following examples.

## Create a Docker Image containing Confluent Hub Connectors

This example shows how to use the [Confluent Hub client](#) to create a Docker image that extends from one of Confluent's Kafka Connect images but which contains a custom set of connectors. This may be useful if you'd like to use a connector that isn't contained in the `cp-kafka-connect` image, or if you'd like to keep the custom image lightweight and not include any connectors that you don't plan to use.

1. Add connectors from [Confluent Hub](#).
2. Choose an image to extend.

Functionally, the `cp-kafka-connect` and the `cp-kafka-connect-base` images are identical. The only difference is that the `cp-kafka-connect` image already contains several of Confluent's connectors, whereas the `cp-kafka-connect-base` image comes with none by default. The `cp-kafka-connect-base` image is shown in this example.

3. Choose the connectors from Confluent Hub that you'd like to include in your custom image. Note that the remaining steps result in a custom image containing a MongoDB connector, a Microsoft Azure IoT Hub connector, and a Google BigQuery connector.
4. Write a Dockerfile.

```
FROM confluentinc/cp-kafka-connect-base:5.3.1

RUN confluent-hub install --no-prompt hpgrahsl/kafka-connect-mongodb:1.1.0 \
    && confluent-hub install --no-prompt microsoft/kafka-connect-iothub:0.6 \
    && confluent-hub install --no-prompt wepay/kafka-connect-bigquery:1.1.0
```

5. Build the Dockerfile.

```
docker build . -t my-custom-image:1.0.0
```

The output from that command should resemble:

```
Step 1/2 : FROM confluentinc/cp-kafka-connect-base
--> e0d92da57dc3
...
Running in a "--no-prompt" mode
Implicit acceptance of the license below:
Apache 2.0
https://github.com/wepay/kafka-connect-bigquery/blob/master/LICENSE.md
Implicit confirmation of the question: You are about to install 'kafka-connect-bigquery' from WePay, as published on Confluent Hub.
Downloading component BigQuery Sink Connector 1.1.0, provided by WePay from Confluent Hub and installing into /usr/share/confluent-hub-components
Adding installation directory to plugin path in the following files:
/etc/kafka/connect-distributed.properties
/etc/kafka/connect-standalone.properties
/etc/schema-registry/connect-avro-distributed.properties
/etc/schema-registry/connect-avro-standalone.properties

Completed
Removing intermediate container 48d4506b8a83
--> 496befc3d3f7
Successfully built 496befc3d3f7
Successfully tagged my-custom-image:1.0.0
```

This results in an image named `my-custom-image` that contains the MongoDB, Azure IoT Hub, and BigQuery connectors, and which is capable of running any/all of them via the Kafka Connect framework.

If you are using a `docker-compose.yml` file and the [Confluent Hub client](#) to build your Kafka environment, use the following properties to enable a connector.

```
connect:
  image: confluentinc/kafka-connect-datagen:latest
  build:
    context: .
    dockerfile: Dockerfile-confluenthub
```

---

## Create a Docker Image containing Local Connectors

This example shows how to create a Docker image that extends the `cp-kafka-connect-base` image to contain one or more local connectors. This is useful if you want to use your connectors instead of pulling connectors from Confluent Hub.

1. Package your local connector in a zip file.
2. Set up the Dockerfile as shown in the example below.

```
FROM confluentinc/cp-kafka-connect-base:5.3.1
COPY target/components/packages/my-connector-5.3.1.zip /tmp/my-connector-5.3.1.zip
RUN confluent-hub install --no-prompt /tmp/my-connector-5.3.1.zip
```

3. Build the Dockerfile.

```
docker build . -t my-custom-image:1.0.0
```

---

## Add Additional Software

This example shows how to add new software to an image. For example, you might want to extend the Kafka Connect client to include the MySQL JDBC driver. If this approach is used to add new connectors to an image, the connector JARs must be on the `plugin.path` or the `CLASSPATH` for the Connect framework.

1. Write the Dockerfile.

```
FROM confluentinc/cp-kafka-connect
ENV MYSQL_DRIVER_VERSION 5.1.39
RUN curl -k -SL "https://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-java-${MYSQL_DRIVER_VERSION}.tar.gz" \
  | tar -xzf - -C /usr/share/java/kafka/ --strip-components=1 mysql-connector-java-5.1.39/mysql-connector-java-${MYSQL_DRIVER_VERSION}-bin.jar
```

2. Build the image.

```
docker build -t foo/mysql-connect:latest .
```

#### Note

This approach can also be used to create images with your own Kafka Connect Plugins.

---

© Copyright 2019, Confluent, Inc. [Privacy Policy](#) | [Terms & Conditions](#). Apache, Apache Kafka, Kafka and the Kafka logo are trademarks of the [Apache Software Foundation](#). All other trademarks, servicemarks, and copyrights are the property of their respective owners.

[Please report any inaccuracies on this page or suggest an edit.](#)

2 Votes



Last updated on Oct 17, 2019.