

Installing

To get started:

1. [Download the latest Conduit release](#)
2. .
3. Unzip:

If you're on Mac, it will look something like this:

tar zxvf conduit_0.9.1_Darwin_x86_64.tar.gz Depending on your operating system you may need to make the conduit binary executable.

chmod a+x conduit

Starting Conduit

Now that we have Conduit installed let's start it up to see what happens.

```
./conduit ... Conduit v0.9.1 darwin/arm64 "" 2024 -  
02-21T14:41:26+00:00 INF All 0 tables opened in 0s component = badger.DB 2024 -02-21T14:41:26+00:00 INF Discard  
stats nextEmptySlot: 0
```

component

```
badger.DB 2024 -02-21T14:41:26+00:00 INF Set nextTxnTs to 0
```

component

```
badger.DB 2024 -02-21T14:41:26+00:00 INF builtin plugins initialized component = builtin.Registry count = 6 2024 -02-  
21T14:41:26+00:00 WRN could not read plugin directory error = "open /app/conduit/connectors: no such file or directory"
```

component

```
standalone.Registry 2024 -02-21T14:41:26+00:00 INF standalone plugins initialized component = standalone.Registry count  
= 0
```

plugin_path

```
/app/conduit/connectors 2024 -02-21T14:41:26+00:00 INF processors initialized component = processor.Service count = 0  
2024 -02-21T14:41:26+00:00 INF connectors initialized component = connector.Service count = 0 2024 -02-  
21T14:41:26+00:00 INF pipelines initialized component = pipeline.Service count = 0 2024 -02-21T14:41:26+00:00 INF  
pipeline configs provisioned component = provisioning.Service created = [ ]
```

deleted

```
[ ]
```

pipelines_path

```
./pipelines 2024 -02-21T14:41:26+00:00 INF grpc server started address = [ :: ] :8084 2024 -02-21T14:41:26+00:00 INF http  
server started address = [ :: ] :8080 2024 -02-21T14:41:26+00:00 INF 2024 -02-21T14:41:26+00:00 INF click here to  
navigate to Conduit UI: http://localhost:8080/ui 2024 -02-21T14:41:26+00:00 INF click here to navigate to explore the HTTP  
API: http://localhost:8080/openapi 2024 -02-21T14:41:26+00:00 INF A few things to point out in this start up message.
```

1. We see Conduit warning that there's noconnectors
2. directory. This is fine because we haven't created one yet.

```
2024 -02-21T14:41:26+00:00 WRN could not read plugin directory error = "open /app/conduit/connectors: no such file or  
directory"
```

component

standalone.Registry 1. We see Conduit telling us the location of the Conduit UI and the HTTP API

2024 -02-21T14:41:26+00:00 INF click here to navigate to Conduit UI: <http://localhost:8080/ui> 2024 -02-21T14:41:26+00:00 INF click here to navigate to explore the HTTP API: <http://localhost:8080/openapi> Now that we have Conduit up and running you can now navigate to <http://localhost:8080> to check the admin UI:

Building a pipeline

While you can provision pipelines via Conduit's UI, the recommended way to do so is using [pipeline configuration file](#).

For this example we'll create a pipeline that will move data from one file to another.

First we'll create the pipelines directory in the same directory as our conduit binary.

mkdir pipelines Next we'll create the pipeline configuration file file-to-file.yaml in the pipelines directory.

version :

2.2 pipelines : -

id : file - to - file status : running description :

Example pipeline reading from file "example.in" and writing into file "example.out". Note that the output file will contain the whole OpenCDC record, the field "payload.after" will contain the base64 encoded line written in "example.in". connectors : -

id : example.in type : source plugin : builtin : file settings : path : ./example.in -

id : example.out type : destination plugin : builtin : file settings : path : ./example.out Start conduit:

./conduit Conduit should start and we should see references to our new pipeline in the output.

```
... Conduit v0.9.1 darwin/arm64 "" 2024 -02-21T16:50:35+00:00 INF All 0 tables opened in 0s component = badger.DB 2024 -02-21T16:50:35+00:00 INF Discard stats nextEmptySlot: 0
```

component

badger.DB 2024 -02-21T16:50:35+00:00 INF Set nextTxnTs to 0

component

badger.DB 2024 -02-21T16:50:35+00:00 INF Deleting empty file: conduit.db/000001.vlog component = badger.DB 2024 -02-21T16:50:35+00:00 INF builtin plugins initialized component = builtin.Registry count = 6 2024 -02-21T16:50:35+00:00 WRN could not read plugin directory error = "open /Users/simonl/work/conduit-test/connectors: no such file or directory"

component

standalone.Registry 2024 -02-21T16:50:35+00:00 INF standalone plugins initialized component = standalone.Registry count = 0

plugin_path

/Users/simonl/work/conduit-test/connectors 2024 -02-21T16:50:35+00:00 INF processors initialized component = processor.Service count = 0 2024 -02-21T16:50:35+00:00 INF connectors initialized component = connector.Service count = 0 2024 -02-21T16:50:35+00:00 INF pipelines initialized component = pipeline.Service count = 0 2024 -02-21T16:50:35+00:00 INF pipeline started component = pipeline.Service pipeline_id = file-to-file 2024 -02-21T16:50:35+00:00 INF pipeline configs provisioned component = provisioning.Service created = ["file-to-file"]

deleted

[]

pipelines_path

./pipelines 2024 -02-21T16:50:35+00:00 INF seeking .. . component = plugin connector_id = file-to-file:example.in plugin_name = builtin:file plugin_type = source position = 0 2024 -02-21T16:50:35+00:00 INF destination connector plugin successfully started component = connector.Destination connector_id = file-to-file-dlq 2024 -02-21T16:50:35+00:00 INF destination connector plugin successfully started component = connector.Destination connector_id = file-to-file:example.out 2024 -02-21T16:50:35+00:00 INF source connector plugin successfully started component = connector.Source connector_id = file-to-file:example.in 2024 -02-21T16:50:35+00:00 INF grpc server started address = [::] :8084 2024 -02-21T16:50:35+00:00 INF http server started address = [::] :8080 2024 -02-21T16:50:35+00:00 INF 2024 -02-21T16:50:35+00:00 INF click here to navigate to Conduit UI: <http://localhost:8080/ui> 2024 -02-21T16:50:35+00:00 INF click here to navigate to explore the HTTP API: <http://localhost:8080/openapi> To test out the pipeline we'll write a few lines into the fileexample.in .

echo

"line 1"

example.in echo

"line 2"

example.in echo

"line 3"

example.in If we look atexample.out we'll see three lines that contain[OpenCDC](#) records.

cat example.out | jq { "position" :

"Nw==" , "operation" :

"create" , "metadata" :

{ "conduit.source.connector.id" :

"file-to-file:example.in" , "file.path" :

"./example.in" , "opencdc.readAt" :

"1708552274797733000" , "opencdc.version" :

"v1" } , "key" :

"MQ==" , "payload" :

{ "before" :

null , "after" :

"bGluZSAX" } } { "position" :

"MTQ=" , "operation" :

"create" , "metadata" :

{ "conduit.source.connector.id" :

"file-to-file:example.in" , "file.path" :

"./example.in" , "opencdc.readAt" :

"1708552285104750000" , "opencdc.version" :

"v1" } , "key" :

"Mg==" , "payload" :

{ "before" :

null , "after" :

"bGluZSAy" } } { "position" :

"MjE=" , "operation" :

"create" , "metadata" :

{ "conduit.source.connector.id" :

"file-to-file:example.in" , "file.path" :

"./example.in" , "opencdc.readAt" :

"1708552350421094000" , "opencdc.version" :

"v1" } , "key" :

"Mw==" , "payload" :

{ "before" :

null , "after" :

"bGluZSAz" } } We decode the .payload.after field to get the data that was inserted into example.in .

cat example.out | jq ".payload.after | @base64d" "line 1" "line 2" "line 3" Congratulations! You've pushed data through your first Conduit pipeline.

What's next?

Looking for more examples? Check out the examples in our [repo](#) .

Now that you've got the basics of running Conduit and creating a pipeline covered. Here are a few places to dive in deeper:

- [Connectors](#)
- [Pipelines](#)
- [Processors](#)
- [Conduit Architecture](#) [Edit this page](#) [Next](#) [What is Conduit?](#)