

Data Engineering with Dagster — Week 3

Production Pipelines

Production Pipelines

Avoiding On-call

Avoiding On-call

At this point we know how to build a pipeline and how to run Dagster as a service. If we swapped out the connection details for our pipeline to talk to an actual S3 bucket and Redis cluster, would we be good to go?

Data pipelines have a reputation for being brittle. We discussed this at the start of the class but this is partly because of the challenges around data engineering and tools and frameworks in data not always fostering engineering best practices.

When it comes to Dagster we should not view our pipelines as one-offs. Instead each exists within our data platform which we treat like any other application. Therefore we have to think about more than just getting our pipeline to execute successfully. We want to think about how it is being triggered, how we can make sure our pipeline is configured correctly for the data it relies on and how to be resilient in the case of failures.

This week we will focus on Dagster abstraction to make our pipelines more robust and something we can deploy regularly.