🏢 Enterprise Data Ops Airflow
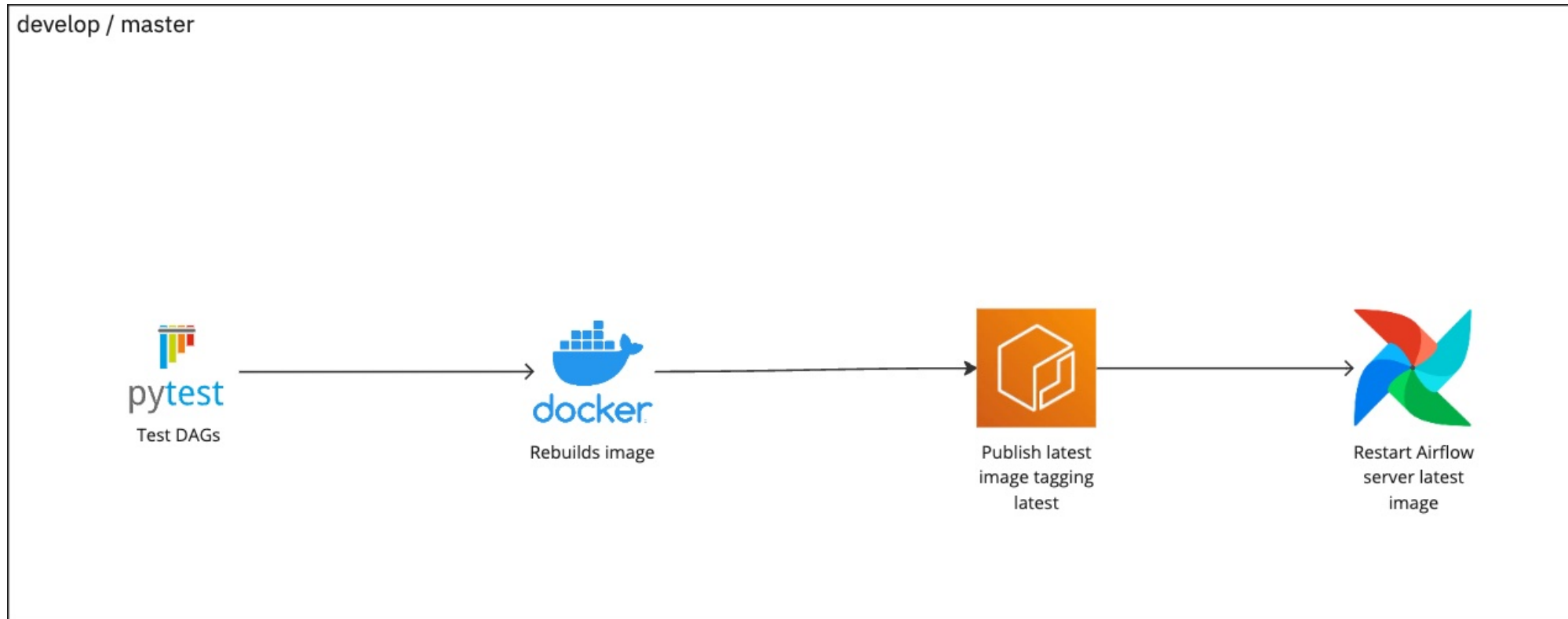
👱 Presented by Nam Tonthat

📅 2023-06-21

📖 Agenda

- 🚀 CI/CD
  - Present
  - Planned
- ❓ Questions

🎁 Present CI/CD

- Builds a new image for any changes
- Publishes this image `~500 MB` each time on updated changes
- Takes 10 minutes for each build

develop / master

pytest — Test DAGs → docker — Rebuilds image → Publish latest image tagging latest → Restart Airflow server latest image
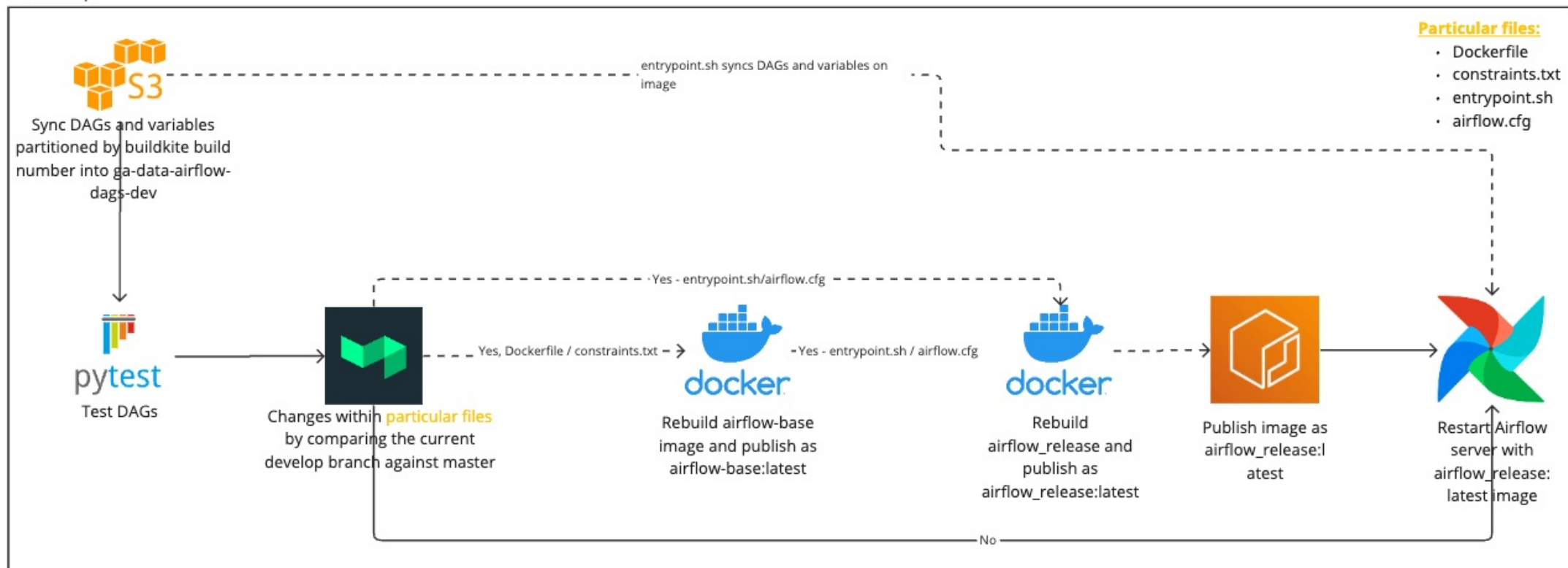
🎯 Planned CI/CD

- Goals:
  - Reduce the time to build by 50%

- Achieved by:

  i. Sync `DAGs` / `variables` into `ga-data-airflow-dags-{env}` S3 bucket.

  ii. Splitting the Dockerfile into a multi stage build
  - `airflow_base` : for requirements
  - `airflow_release` : for `entrypoint.sh` / `airflow.cfg` changes

  iii. Update `entrypoint.sh` using `aws s3 sync` to sync DAGs back

- *Other changes*:
  - Using SEEK's open source software plugins (`seek-oss/docker-ecr-publish`)
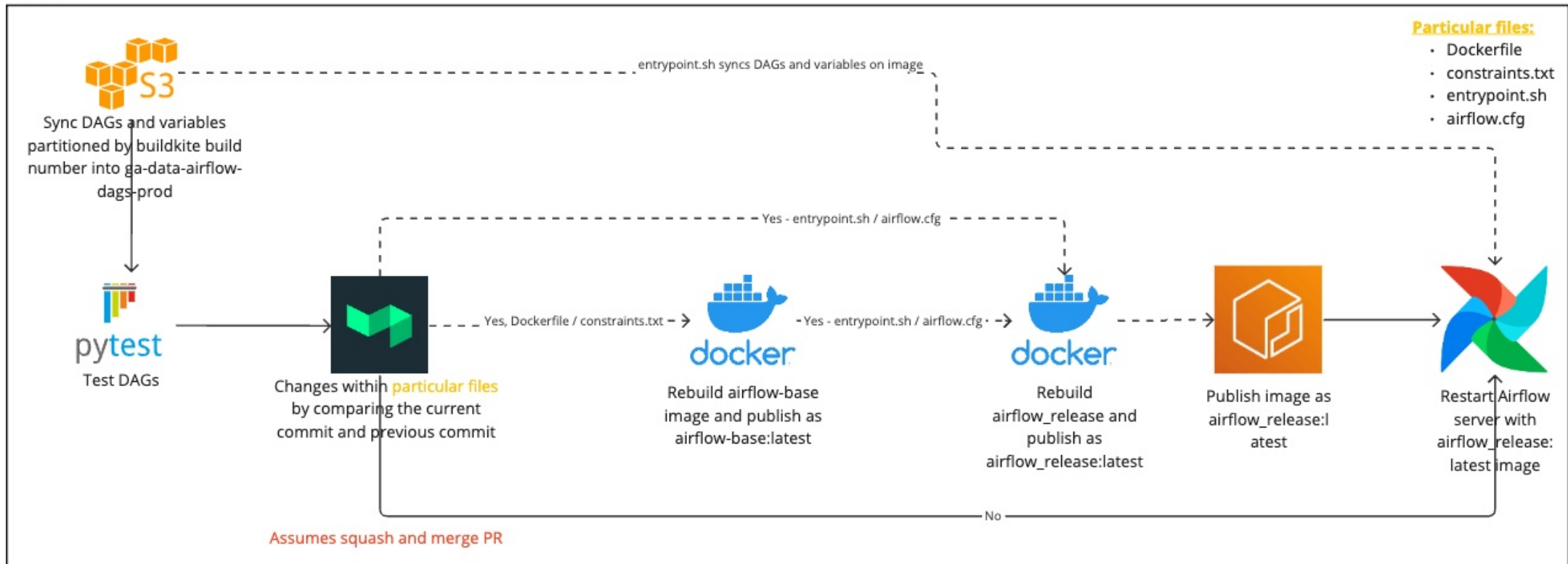
🎯 Planned CI/CD - develop

develop

Particular files:
- Dockerfile
- constraints.txt
- entrypoint.sh
- airflow.cfg

S3

Sync DAGs and variables partitioned by buildkite build number into ga-data-airflow-dags-dev

entrypoint.sh syncs DAGs and variables on image

pytest

Test DAGs

Changes within particular files by comparing the current develop branch against master

Yes, Dockerfile / constraints.txt →

Yes - entrypoint.sh/airflow.cfg

docker

Rebuild airflow-base image and publish as airflow-base:latest

Yes - entrypoint.sh / airflow.cfg

docker

Rebuild airflow_release and publish as airflow_release:latest

Publish image as airflow_release:latest

Restart Airflow server with airflow_release: latest image

No

🎯 **Planned CI/CD - `master`**

Difference between `develop` and `master` is that each PR is a **squash and merge.**
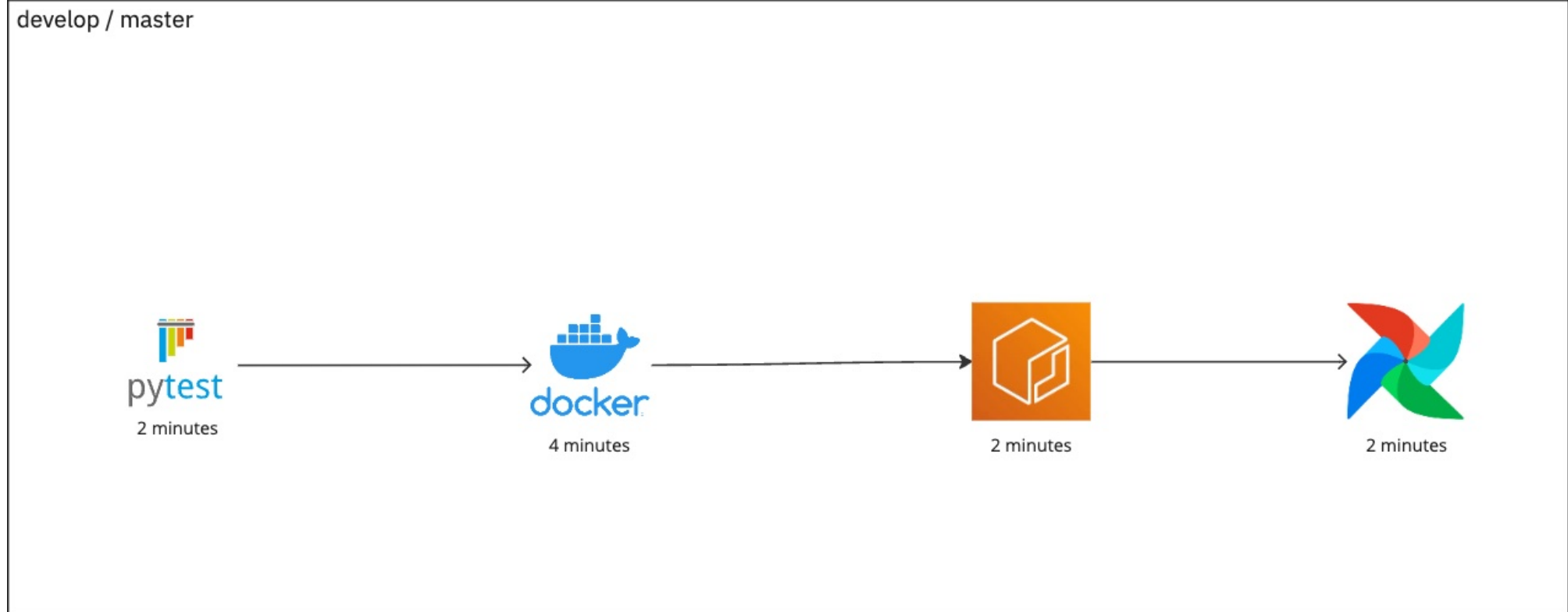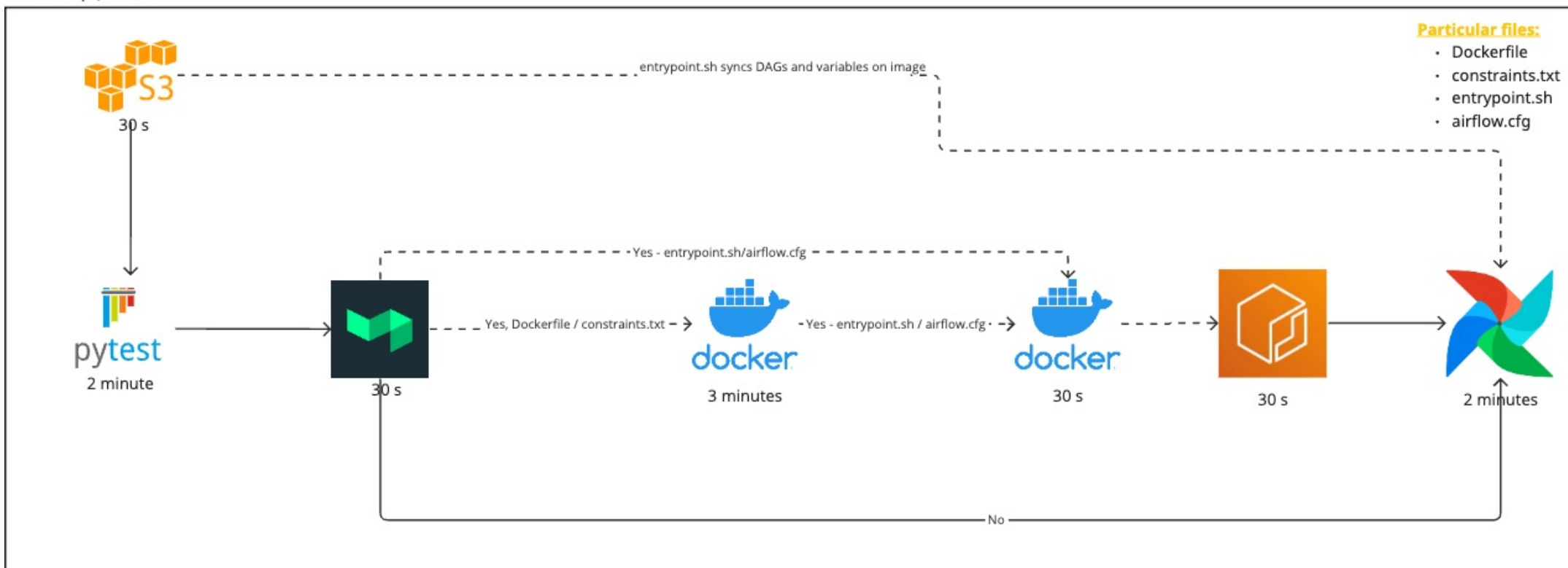
Timing differences - Present CI/CD

- Build time: 10 minutes

Timing differences - Planned CI/CD

- Build times:
  - 5 minutes for DAG / variable changes (most of the time ~95%)
  - 10 minutes for rebuilding image

? Questions