




Enterprise Data Ops Airflow - Migration to Unified Airflow Platform

 Presented by Nam Tonthat

 2023-08-30

Agenda

1.  Progress
2.  Challenges
3.  Remaining
4.  Maintenance
5.  Demo

Progress

- Migration of 80% of our DAGs (~4 remaining) from `docker-airflow` to `airflow-fargate-corporate`

Repository Goals

- Identical working envs between local and `buildkite`
- Fast (<5 mins) CI/CD
- Friendly developer experience

★ Key Features

Dynamic DAGs

🎯 Motivation

Reduce manual effort maintaining multiple DAGs for different envs

🏗️ Implementation

- A `base_config.yml` in `Jinja2` template to have a consistently render each DAG
- Different `env.yml` files nested within their folder
- Use `common_airflow.get_dag_configs` function; adds additional metadata on rendering.

★ Key Features

📁 Folder Structure

```
├── workday_rest
│   ├── base_config.yml
│   ├── dag.py
│   └── envs
│       ├── blue.yml
│       ├── pink.yml
│       └── prod.yml
├── powerbi_api
│   ├── base_config.yml
│   ├── dag.py
│   ├── envs
│   │   ├── dev.yml
│   │   └── prod.yml
│   └── tasks
│       ├── dev.yml
│       └── prod.yml
```

★ Key Features

🌀 Under the hood features:

- Addition of `pre-commit` to ensure each commit is properly linted
- Catching `import` errors
- Emojis 🧑

🏛️ Helper Classes

- `SeekAWSBatchOperator` renamed to `SeekBatchOperator`
 - By default, `aws_conn_id` is set to `corporate`
 - Support for using the `corporate-hr`
- New helper classes `SeekCorporateDAG` (adds correct tags and `corporate_` prefix) and `SeekCorporateExternalTaskSensor`.

Challenges

- Migration over to using the `dataops-unified-airflow-{env}` role
- `AwsAccount` (`@dataclass`) has been created to manage the connections.
- Dealing with the `dags/corporate` `PYTHONPATH` issue

Remaining

- High impact DAGs
- Testing of PagerDuty integration

Planned Migration

- All moderate impact DAGs that have dependency on `crm_base_semantic` and `salesforce_v2` will be turned off in production.
- Mid September switch over (post month end reporting).

Maintenance

What does it mean for you?

- If the DAG is already switched on in `unified-airflow` in production. Switch over completely to `airflow-fargate-corporate`.
- Otherwise if the DAG exists in `unified-airflow` but not switched on, you'll have to edit both the `airflow-fargate-corporate` and `docker-airflow` repos to keep them in sync. (Only `dag.py` related changes)

Demo

 **Build a DAG within five minutes**