



Airflow Multi-Team Project Structure

Multi-team Airflow setup with domain-based DAG folders. Clear ownership boundaries, shared libraries, and team autonomy.

#airflow #python #data #enterprise #multi-team #domains

PNG

PDF

Copy

Prompt

Project Directory

airflow-platform/

- >

📁

 dags/
- >

📁

finance/ Finance team DA...
- 📄

 __init__.py
- 📄

 revenue_daily.py
- 📄

 invoicing.py
- 📄

 README.md Team docs
- >

📁

marketing/ Marketing team ...
- 📄

 __init__.py
- 📄

 campaign_sync.py
- 📄

 attribution.py
- >

📁

data_platform/ Platform team D...
- 📄

 __init__.py
- 📄

 dbt_run.py
- 📄

 data_quality.py
- >

📁

common/ Shared DAG util...
- 📄

 __init__.py
- 📄

 alerts.py
- 📄

 dag_factory.py
- >

📁

plugins/
- 📄

 __init__.py
- >

📁

operators/
- 📄

 __init__.py
- 📄

 dbt_operator.py
- 📄

 slack_operator.py
- >

📁

hooks/
- 📄

 __init__.py
- 📄

 warehouse_hook.py
- >

📁

include/
- >

📁

sql/
- 📁

finance/
- 📁

marketing/
- >

📁

schemas/
- 📄

 events.json
- >

📁

tests/
- 📄

 conftest.py
- >

📁

finance/
- 📄

 test_revenue_daily.py
- >

📁

marketing/
- 📄

 test_campaign_sync.py
- 📄

 docker-compose.yml
- 📄

 pyproject.toml
- 📄

 CODEOWNERS GitHub ownership
- 📄

 .gitignore
- 📄

 README.md

Why This Structure?

Domain-based folders (`finance/` , `marketing/`) give teams clear ownership. Each team manages their own DAGs while sharing common utilities and operators. CODEOWNERS enforces review rules. Tests mirror the DAG structure for easy navigation.

Key Directories

- dags/{team}/** - Team-owned DAGs with dedicated README
- dags/common/** - Shared utilities like dag_factory and alerts
- plugins/operators/** - Organization-wide custom operators
- CODEOWNERS** - Enforce PR reviews by domain owners

Getting Started

- Clone repo and `pip install -e ".[dev]"`
- Create new team folder: `dags/{team_name}/`
- Add team to CODEOWNERS file
- Use `dag_factory` for consistent DAG creation
- Add tests in `tests/{team_name}/`

DAG Factory

```
# dags/common/dag_factory.py
from airflow import DAG
from datetime import datetime, timedelta

DEFAULT_ARGS = {
    "owner": "data-platform",
    "retries": 2,
    "retry_delay": timedelta(minutes=5),
}

def create_dag(dag_id: str, schedule: str, **kwargs):
    return DAG(
        dag_id=dag_id,
        default_args=DEFAULT_ARGS,
        start_date=datetime(2024, 1, 1),
        schedule=schedule,
        catchup=False,
    )
```

Code Ownership

Use CODEOWNERS to require team approval for their DAGs:
`/dags/finance/ @finance-team` . This prevents accidental changes and ensures domain expertise in reviews.

When To Use This

- Multiple teams sharing one Airflow instance
- 50+ DAGs across different domains
- Need clear ownership and review boundaries
- Enterprise deployments with governance requirements

Best Practices

- Each team folder should have its own README
- Use DAG tags matching team names for UI filtering
- Prefix DAG IDs with team name: `finance_revenue_daily`
- Shared operators go in plugins/, team-specific in dags/{team}/
- Run team-specific tests in CI based on changed files