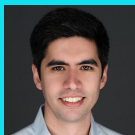




Building Blocks, Not Factories



Collin McNulty

How Dag factories come to be

ASTRONOMER

So what's the problem?

ASTRONOMER

Goals for Abstractions

1. Get credit for a job well done
2. Move on to the next project
3. New: Evolve to handle new curveballs

Make reusable Task Groups instead

Example Task Group

```
@task_group
def source(table: str, columns):
    primary_data = SQLExecuteQueryOperator(task_id="primary_data",
                                             sql=query.format(table, columns),
                                             conn_id="finance_postgres_db")

    verify_data(primary_data)

@task
def verify_data(primary_data):
    ...
```

Comparing Dags

```
input_data_table: "financial.predictions_fy26"  
columns: ["revenue_monthly", "product_line", ...]
```

```
classifier:  
    type: "random_forest"  
    max_depth: 4
```

```
output_data_table: "estimates.fy27_inventory"
```

```
schedule: "@daily"  
name: "next_year_inventory_estimate"
```

```
from airflow.sdk import dag
```

```
from shared_task_groups import source, classify,  
store_result
```

```
@dag(schedule="@daily")
```

```
def next_year_inventory_estimate():  
    inputs = source("financial.predictions_fy26",  
                    ["revenue_monthly", "product_line", ...])  
    model_info = classify(inputs,  
                           type="random_forest",  
                           max_depth=4  
    )  
    store_result(model_info, "estimates.fy27_inventory")
```

Real Reusable Task Group

```
@task_group
def close_expired_maydays(
    cause: Cause, time_to_live: int = 60, never_autoclose: bool = False
):
    top_of_hour = check_top_of_hour()
    active_maydays = get_active_maydays(cause.name)
    close_inactive_task = close_closable_maydays.partial(
        mayday_ttl=time_to_live,
        never_autoclose=never_autoclose,
        top_of_hour=top_of_hour,
    ).expand(mayday=active_maydays)
    active_maydays >> close_inactive_task
    close_inactive_task >> EmptyOperator(
        task_id="always_succeed", trigger_rule="all_done"
    )
    Empty Operator(task_id="reflect_upstream_state")
```

Few, simple
parameters

Abstracted
complexity

The 2026 Apache Airflow® Survey is Live!

- Takes just 7 minutes
- Get a free AI Orchestration Fundamentals Certification Exam or DAG Authoring Certification (normally \$150 each)

