Introduction to Apache Airflow

INTRODUCTION TO APACHE AIRFLOW IN PYTHON

Mike Metzger
Data Engineer

datacamp

# What is data engineering?

Data engineering is:

- Taking any action involving data and turning it into a reliable, repeatable, and maintainable process.

What is a workflow?

A *workflow* is:

- A set of steps to accomplish a given data engineering task
  - Such as: downloading files, copying data, filtering information, writing to a database, etc

- Of varying levels of complexity

- A term with various meaning depending on context
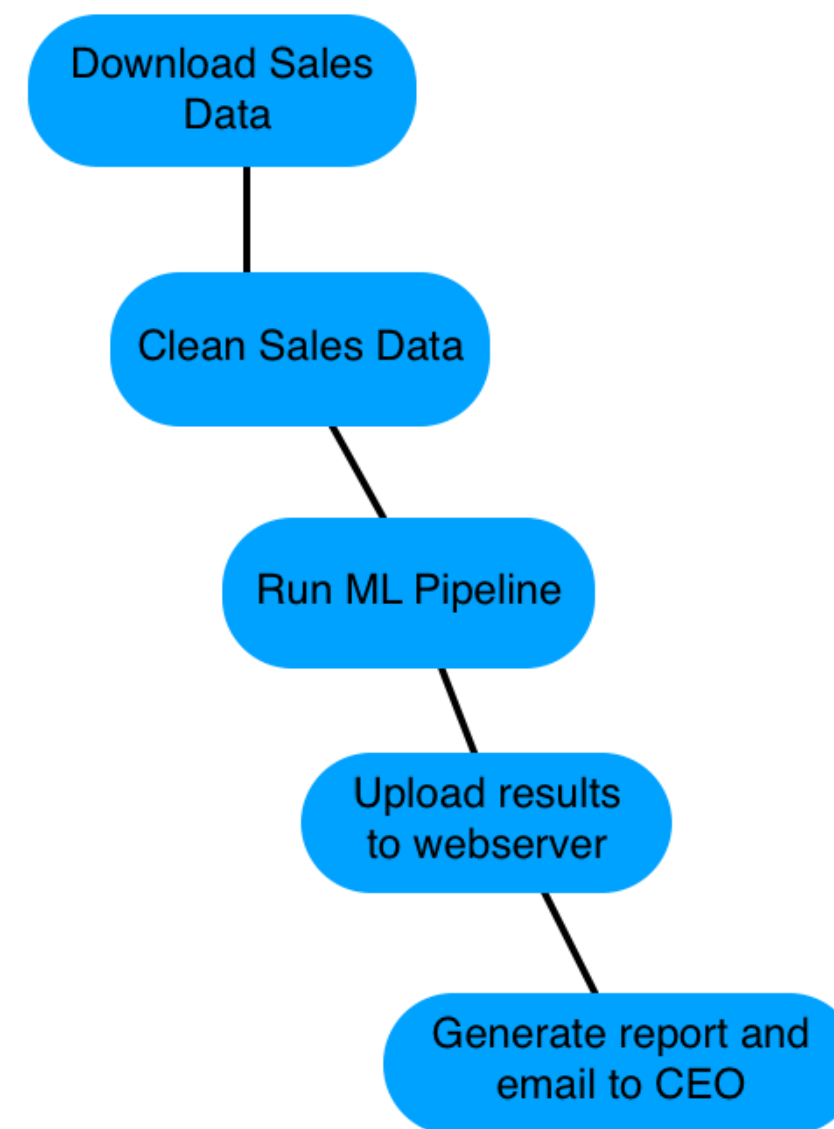
What is Airflow?

Airflow is a platform to program workflows, including:

- Creation

- Scheduling

- Monitoring

Airflow continued...

- Can implement programs from any language, but workflows are written in Python

- Implements workflows as DAGs: Directed Acyclic Graphs

- Accessed via code, command-line, or via web interface / REST API

[1] https://airflow.apache.org/docs/stable/

Other workflow tools

Other tools:

- Luigi

- SSIS

- Bash scripting

Quick introduction to DAGs

A *DAG* stands for *Directed Acyclic Graph*

- In Airflow, this represents the set of tasks that make up your workflow.

- Consists of the tasks and the dependencies between tasks.

- Created with various details about the DAG, including the name, start date, owner, etc.

- Further depth in the next lesson.
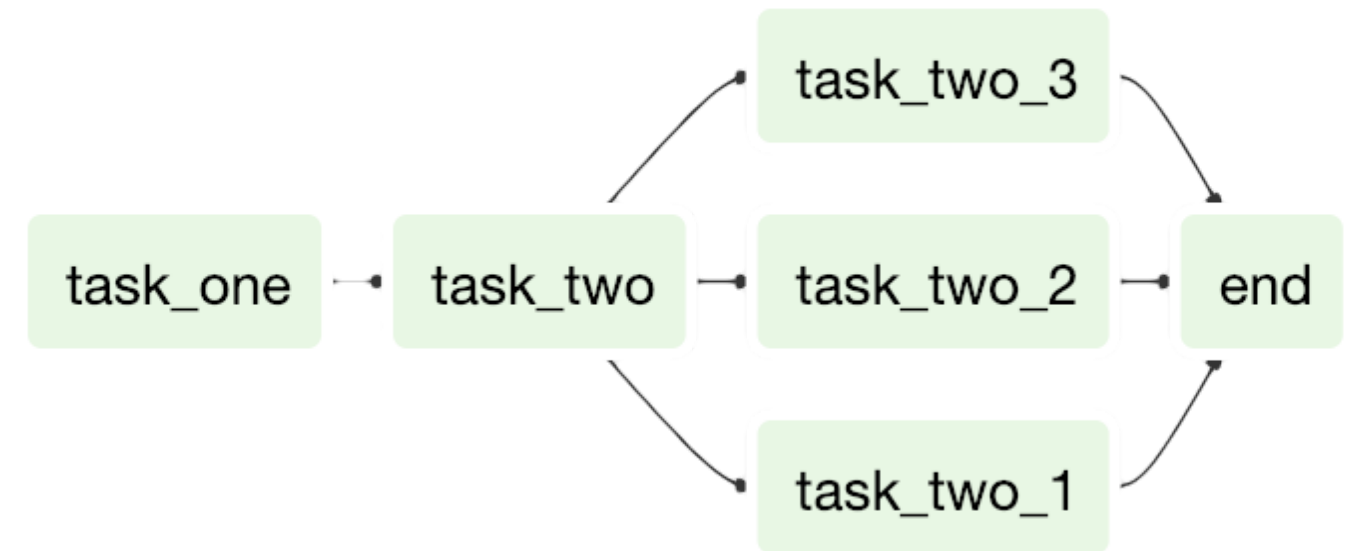
DAG code example

Simple DAG definition:

```python
etl_dag = DAG(
    dag_id='etl_pipeline',
    default_args={"start_date": "2024-01-08"}
)
```

Running a workflow in Airflow

Running a simple Airflow task

```
airflow tasks test <dag_id> <task_id> [execution_date]
```

Using a DAG named *example-etl*, a task named *download-file* on 2024-01-10:

```
airflow tasks test example-etl download-file 2024-01-10
```

Let's practice!

datacamp

Airflow DAGs

INTRODUCTION TO APACHE AIRFLOW IN PYTHON

Mike Metzger
Data Engineer

What is a DAG?

DAG, or *Directed Acyclic Graph*:

- *Directed*, there is an inherent flow representing dependencies between components.

- *Acyclic*, does not loop / cycle / repeat.

- *Graph*, the actual set of components.

- Seen in Airflow, Apache Spark, dbt
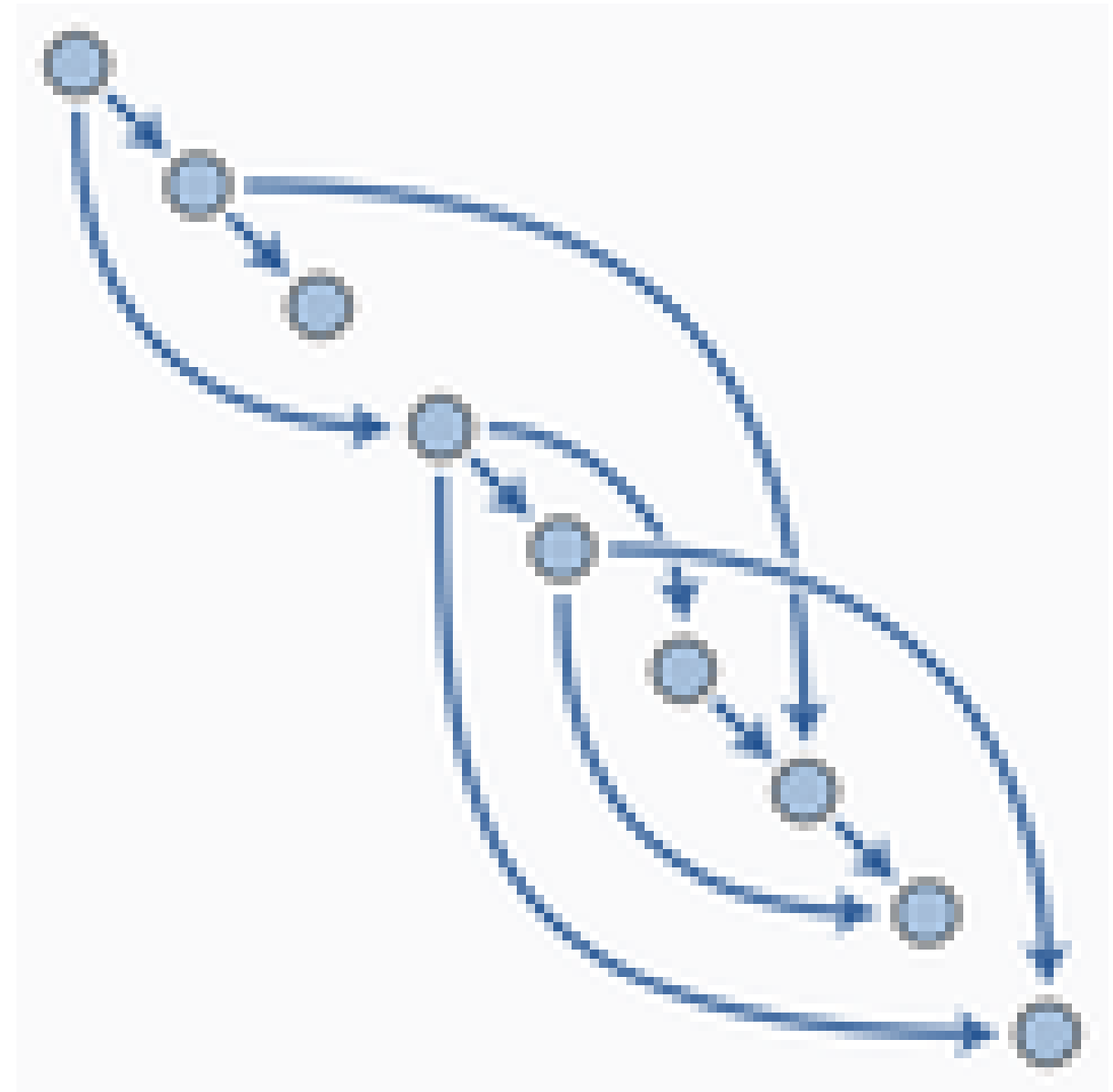
[1] https://en.m.wikipedia.org/wiki/Directed_acyclic_graph

DAG in Airflow

Within Airflow, DAGs:

- Are written in Python (but can use components written in other languages).

- Are made up of components (typically *tasks*) to be executed, such as operators, sensors, etc.

- Contain dependencies defined explicitly or implicitly.
  - ie, Copy the file to the server before trying to import it to the database service.

Define a DAG

Example DAG:

```python
from airflow import DAG

from datetime import datetime
default_arguments = {
    'owner': 'jdoe',
    'email': 'jdoe@datacamp.com',
    'start_date': datetime(2020, 1, 20)
}

with DAG('etl_workflow', default_args=default_arguments ) as etl_dag:
```

Define a DAG (before Airflow 2.x)

Example DAG:

```python
from airflow import DAG

from datetime import datetime
default_arguments = {
    'owner': 'jdoe',
    'email': 'jdoe@datacamp.com',
    'start_date': datetime(2020, 1, 20)
}


etl_dag = DAG('etl_workflow', default_args=default_arguments )
```

DAGs on the command line

Using `airflow` :

- The `airflow` command line program contains many subcommands.

- `airflow -h` for descriptions.

- Many are related to DAGs.

- `airflow dags list` to show all recognized DAGs.

Command line vs Python

Use the command line tool to:

- Start Airflow processes

- Manually run DAGs / Tasks

- Get logging information from Airflow

Use Python to:

- Create a DAG

- Edit the individual properties of a DAG

Let's practice!

INTRODUCTION TO APACHE AIRFLOW IN PYTHON

datacamp

Airflow web interface

INTRODUCTION TO APACHE AIRFLOW IN PYTHON

Mike Metzger
Data Engineer

DAGs view

DAGs view DAGs

DAGs view owner

DAGs view runs

DAGs view schedule

DAGs view last run

DAGs view next run

DAGs view recent tasks

DAGs view example_dag

DAG detail view

DAG graph view

DAG code view

Audit logs

Web UI vs command line

In most cases:

- Equally powerful depending on needs

- Web UI is easier

- Command line tool may be easier to access depending on settings

Let's practice!

INTRODUCTION TO APACHE AIRFLOW IN PYTHON

datacamp