Apache Airflow

What is Airflow?

Apache Airflow is a platform for programmatically authoring, scheduling, and monitoring workflows.
It is completely open-source and is especially useful in architecting complex data pipelines.
It's written in Python, so you're able to interface with any third party python API or database to extract, transform, or load your data into its final destination.
It was created to solve the issues that come with long-running cron tasks that execute hefty scripts.

A little bit of History:

In 2015, Airbnb experienced a problem.
They were growing like crazy and had a massive amount of data that was only getting larger.
To achieve the vision of becoming a fully data-driven organization, they had to grow their workforce of data engineers, data scientists, and analysts- all of whom had to regularly work to automate processes by writing scheduled batch jobs.
To satisfy the need for a robust scheduling tool, Data Engineer Maxime Beauchemin created and open-sourced Airflow with the idea that it would allow them to quickly author, iterate on, and monitor their batch data pipelines.

Here are a few cool things we can do with Airflow:

- Aggregate daily sales team updates from Sales-force to send a daily report to executives at the company.
- Use Airflow to organize and kick off machine learning jobs running on external Spark clusters.
- Load website/application analytic data into a data warehouse on an hourly basis.

Core concepts:

DAG

DAG stands for "Directed Acyclic Graph". Each DAG represents a collection of all the tasks you want to run and is organized to show relationships between tasks directly in the Airflow UI. They are defined this way for the following reasons:

1. Directed: If multiple tasks exist, each must have at least one defined upstream or downstream task.
2. Acyclic: Tasks are not allowed to create data that goes on to self-reference. This is to avoid creating infinite loops.
3. Graph: All tasks are laid out in a clear structure with processes occurring at clear points with set relationships to other tasks.

TASKS

Tasks represent each node of a defined DAG. They are visual representations of the work being done at each step of the workflow, with the actual work that they represent being defined by Operators.

OPERATORS

Operators in Airflow determine the actual work that gets done. They define a single task, or one node of a DAG. DAGs make sure that operators get scheduled and run in a certain order, while operators define the work that must be done at each step of the process.

HOOKS

Hooks are Airflow's way of interfacing with third-party systems. They allow you to connect to external APIs and databases like Hive, S3, GCS, MySQL, Postgres, etc. They act as building blocks for larger operators. Secure information such as authentication credentials are kept out of hooks- that information is stored via Airflow connections in the encrypted metadata db that lives under your Airflow instance.

PLUGINS

Airflow plugins represent a combination of Hooks and Operators that allows you to accomplish a certain task, like transfer data from sale-forces to Redshift.

CONNECTIONS

Connections are where Airflow stores information that allows you to connect to external systems, such as authentication credentials or API tokens. This is managed directly from the UI and the actual information is encrypted and stored in as metadata in Airflow's underlying Postgres or MySQL

Samir Benzada.