Data pipeline

DSIA-5102A - ESIEE Paris

nicolas.vo@esiee.fr, raphael.courivaud@esiee.fr

What are data pipelines solving ?

Orchestration
Scheduling
ETL
DAG

Extract - Transform - Load

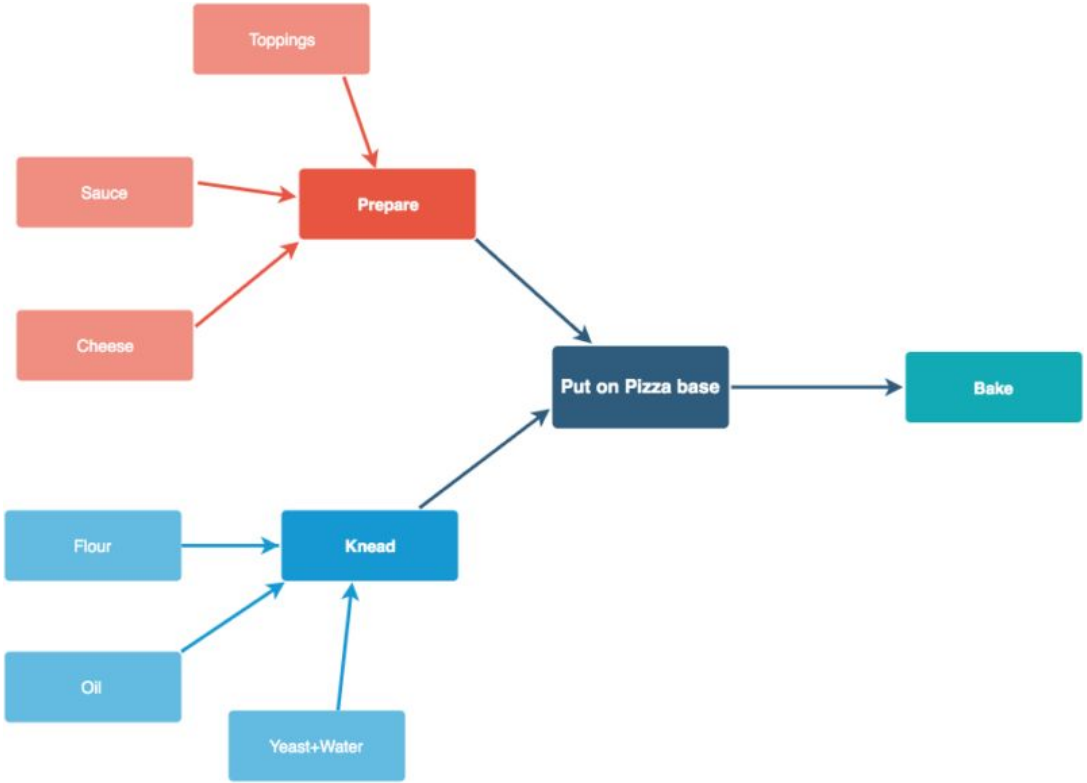
Directed acyclic graph

Workflow as code

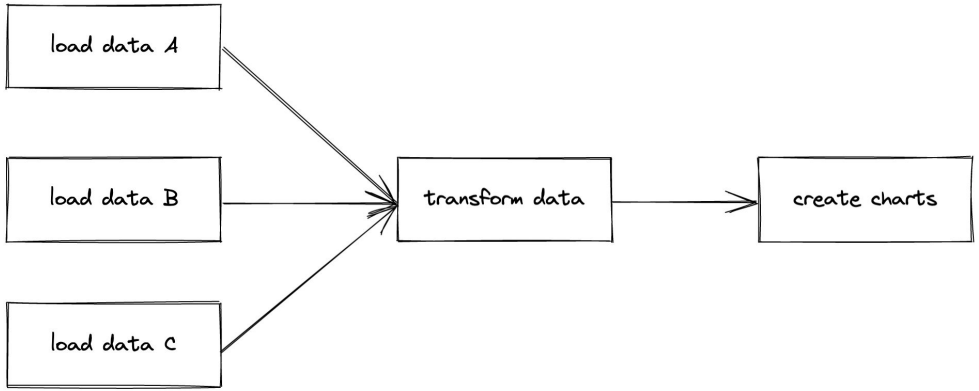
Data on-demand

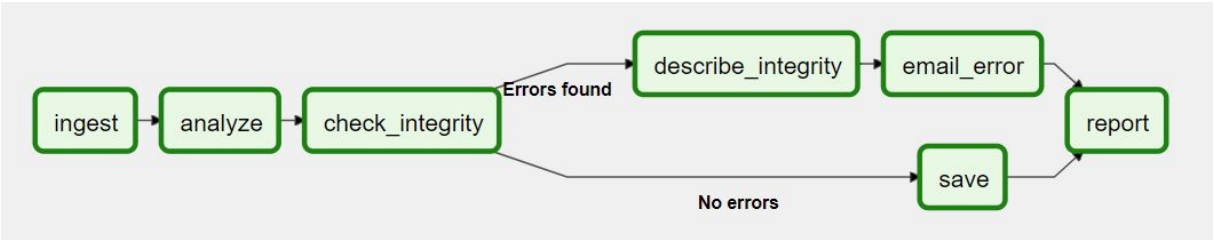
Data pipeline

Directed acyclic graph (DAG)



Directed acyclic graph (DAG)





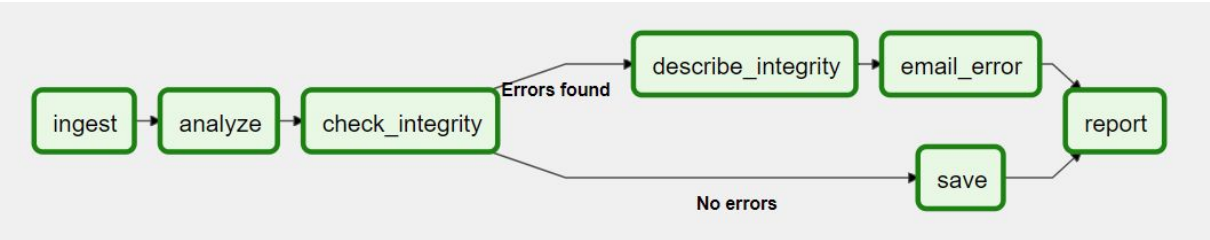
Directed acyclic graph (DAG)

General concepts

- Declare task dependencies
- Order task executions
- Isolate what tasks do
- **Sequencing, coordination, scheduling**

Airflow concepts

- Status: running, queued, scheduled, retry (n times), failed
- Fail downstream tasks
- Schedule for specific time
- Depend on completion of other DAG



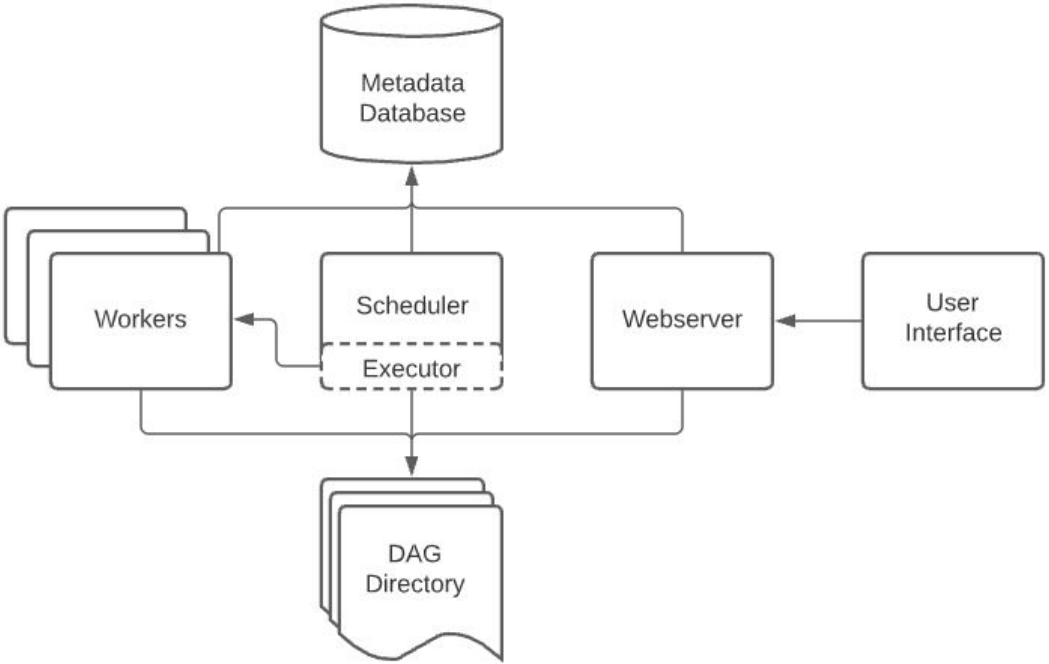
Airflow

- Created in 2014 by Airbnb
- Open-sourced to Apache in 2016
- Local, Docker Compose, Kubernetes
- Workflow as code
- DAGs not limited to executing Python

Airflow

Architecture

- Scheduler: tells executor which task is next
- Executor: sends task to worker for execution
- Worker: runs the actual task



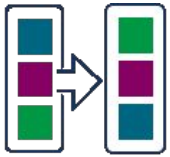
Other data pipeline frameworks

- Luigi (Spotify)
- Prefect
- Dagster
- Apache NiFi

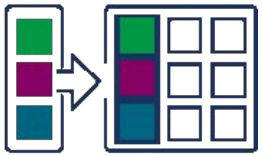
ETL & ELT

ETL and ELT

Extract

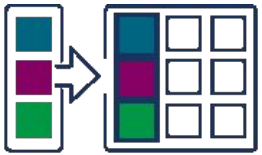


Transform



Load

Extract



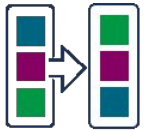
Load

Transform

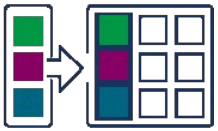
Extract Transform Load (ETL)

- Transform between departure and arrival
- Know ahead of what exactly needs to be transformed
- Clean output at destination
- Hard to know ahead of time
- Sensitive to changes
- Lot of maintenance

Extract



Transform

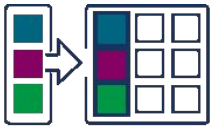


Load

Extract Load Transform (ELT)

- Transform after arrival
- Transfer source data to target destination faster
- Raw data for evolving needs
- Longer transformation

Extract



Load

Transform

Airflow executors

Airflow

Executors

Local executor

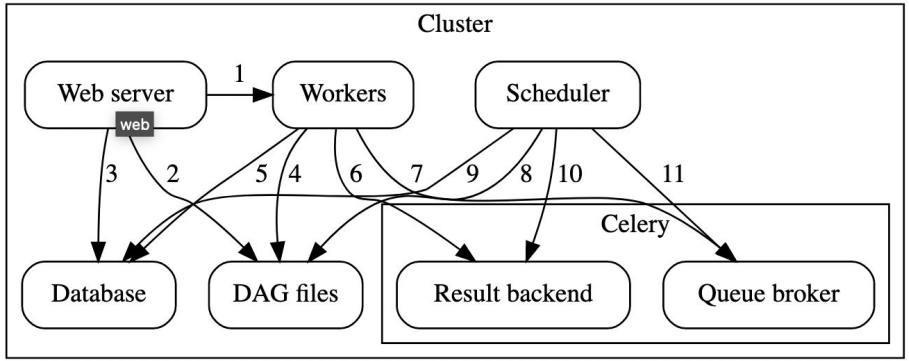
- Execute on same machine as scheduler (single-node)
- Easy to set up
- Cheap
- Parallelism possible
- Not scalable
- Single point of failure

Airflow

Executors

Celery executor

- Execute on worker machines
- Horizontal scaling
- Worker machines always up
- Static machine configurations
- Expensive
- Moderately scalable



Airflow

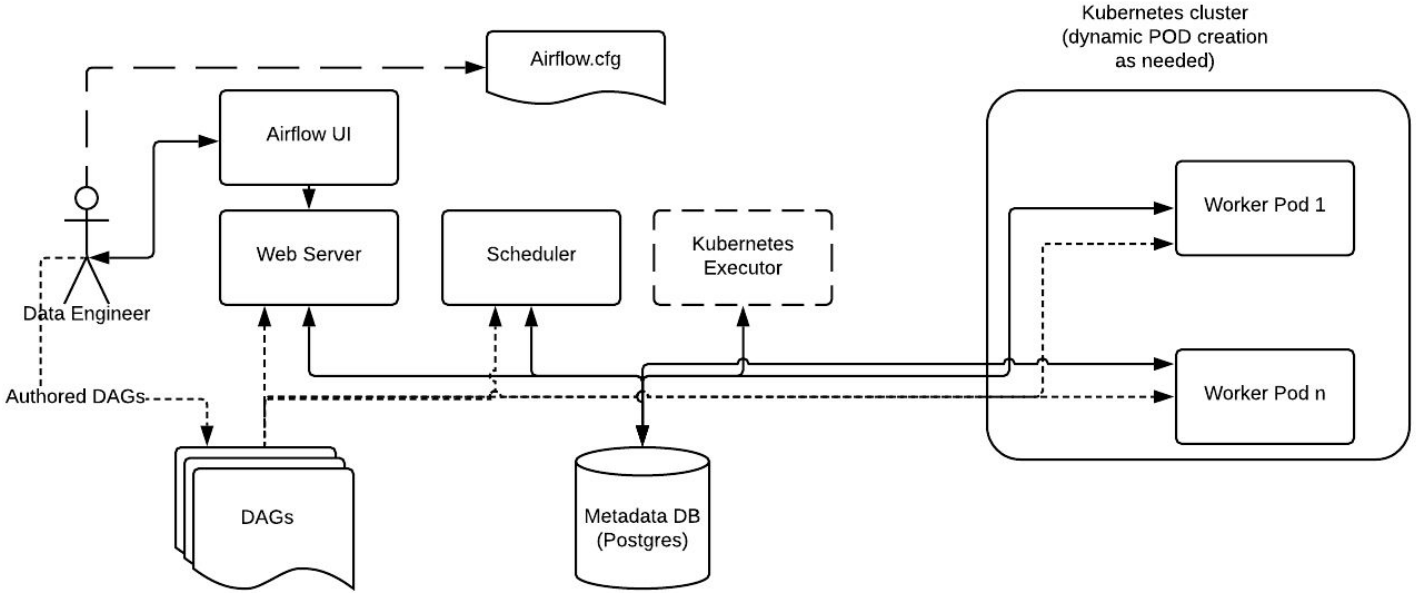
Executors

Kubernetes executor

- Execute tasks on pods
- Scale pods up and down based on load
- Specify resources by pod
- Container images per DAG
- Hard to set up

Airflow

Kubernetes executor



Link

Data pipeline
ETL & ELT
Data on-demand

Demo

Version control

DSIA-5102A - ESIEE Paris

nicolas.vo@esiee.fr, raphael.courivaud@esiee.fr

Version control

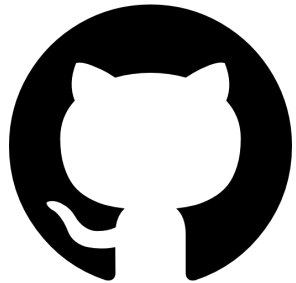
Before

- script_version_nico_2.py
- script_raph_v4.py
- script_final_v8.py

Version control

Now

- Git > local
- GitHub, GitLab > remote
- Collaborative development
- Track changes
- Git is hard but worth it
- Git is hard so use wrapper tools



Version control

Basic concepts

- Repository
- Stage changes (add, rm, restore)
- Commit changes
- Push changes
- Pull
- Clone
- Branch
- Pull request

Version control

Advanced concepts

- Revert
- Reset
- Rebase
- Cherry-pick
- Checkout
- Fork
- Status
- Log
- Diff

Demo