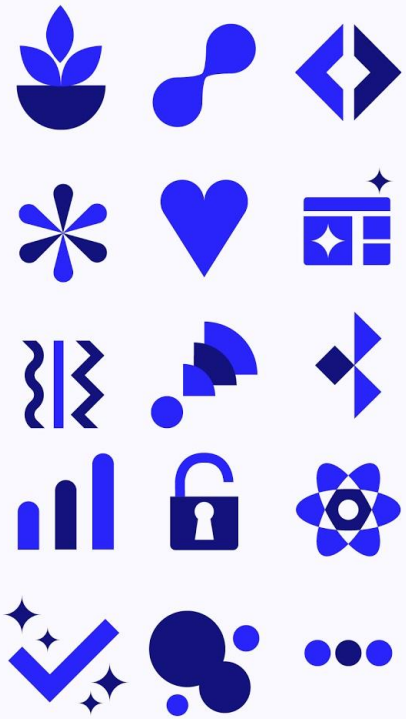
Polidea

Unique tech



Polidea

What's coming in Apache Airflow 2.0

Apache Airflow

Airflow is a platform to programmatically author, schedule and monitor workflows.

Dynamic/Elegant

Extensible

Scalable

What’s on today ?

What is the presentation about ?

- The team @ Polidea
- What the Airflow ?
- Where Apache Airflow is now?
- What’s coming in Apache Airflow 2.0.

Team @ Polidea

Hi!

Jarek Potiuk

Principal Software Engineer @Polidea

Apache Airflow PMC member

Certified GCP Architect

ex-Googler, ex-CTO, ex-choir member

@higrys

Apache Airflow Development team@ Polidea

Jarek Potiuk

Kamil Breguła

Tomasz Urbaszek

Karolina Rosół

Tobiasz Kędzierski

Michał Słowikowski

Past:

Dariusz Aniszewski

Szymon Przedwojski

Antoni Smoliński

Apache Airflow Website team @ Polidea

Kamil Breguła

Zuzanna Rykowska

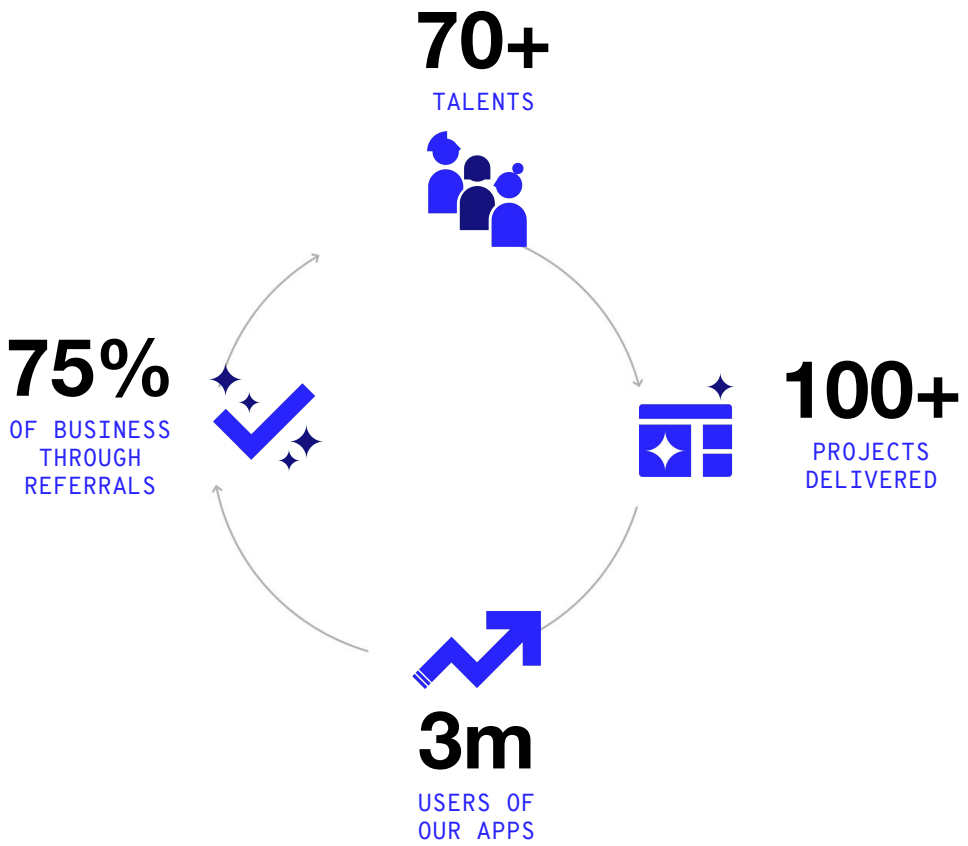
Kamil Gabryjelski

Tomasz Urbaszek

Marta Strzałkowska

Magdalena Węgrzyńska

Team @Polidea

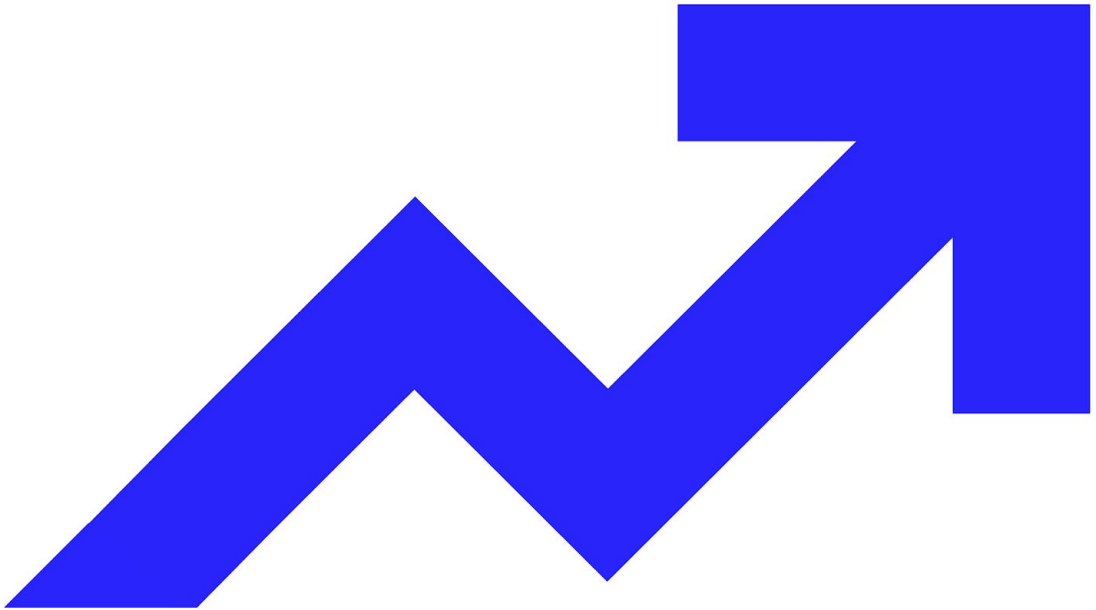


Polidea & Apache Airflow

Timeline

December 2019

6 (9) people



August 2018

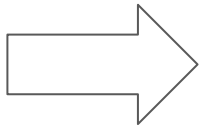
2 people

Our tasks

- 130+ operators
- 18+ GCP services
- Oozie-To-Airflow
- New Apache Airflow Website

What we delivered extra

- Documentation improvements
- Breeze - improved dev environment
- Py2 -> Py3
- Pylint compatibility
- Pre-commit framework introduction
- CI environment reimplemented
- Operator scaffolding
- Convert tests to pytests



2 Apache Airflow Committers

Apache Airflow PMC member

Open-source friendly company

Polidea

MCE 2018
REDEFININGTECH

Mobile Warsaw

Apache Airflow

Why Apache Airflow and not one of these?

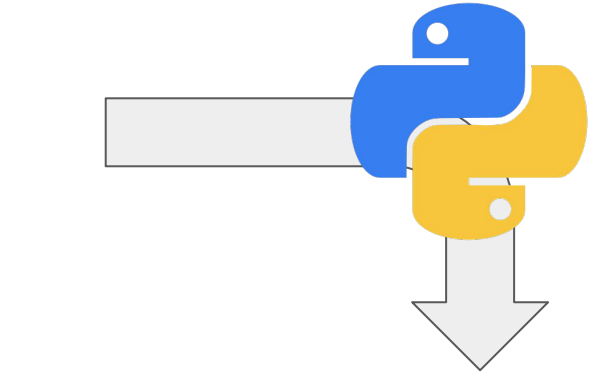
And many, many, many more ....

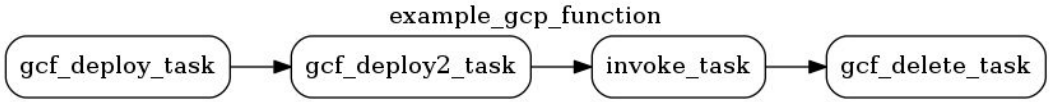
Airflow is an Orchestrator

- Tells others what/when to do
- Synchronizes work between others
- Monitors what's going on
- Intervenes if needed
- Mostly does not do much

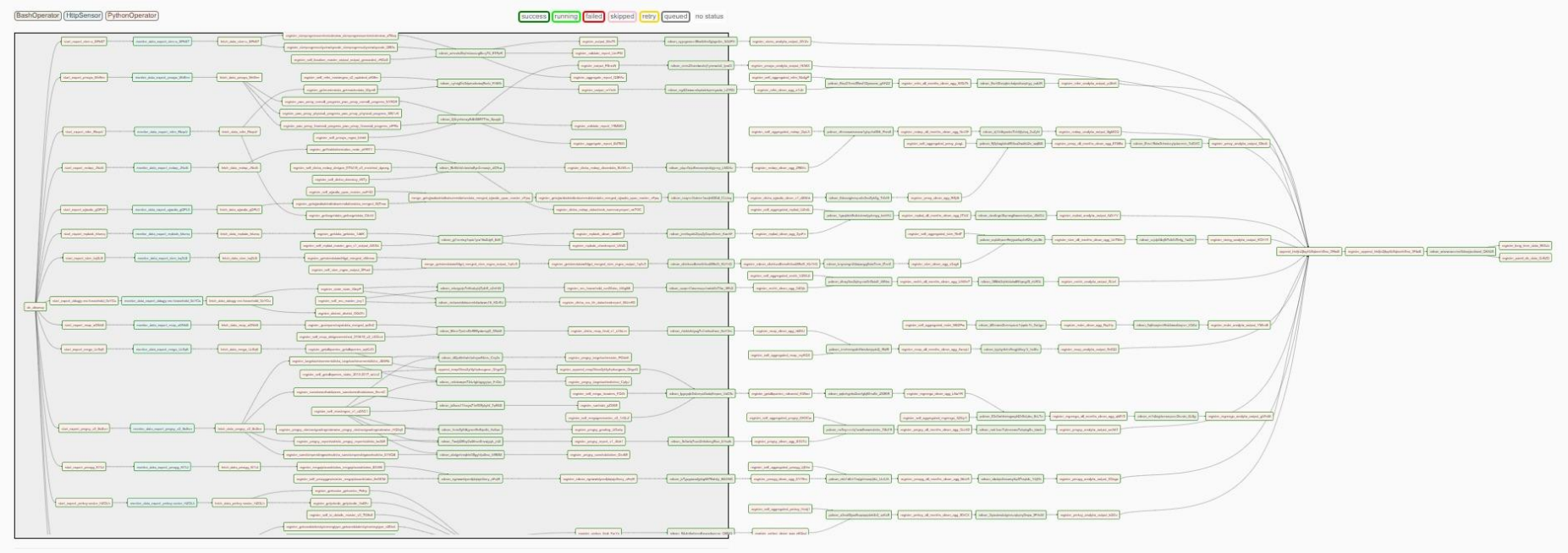
Airflow is Python

```python
with models.DAG(
    'example_gcp_function',
    default_args=default_args,
    schedule_interval=None  # Override to match your needs
) as dag:
    deploy_task = GcfFunctionDeployOperator(
        task_id="gcf_deploy_task",
        project_id=GCP_PROJECT_ID,
        location=GCP_LOCATION,
        body=body,
        validate_body=GCP_VALIDATE_BODY
    )
    deploy2_task = GcfFunctionDeployOperator(
        task_id="gcf_deploy2_task",
        location=GCP_LOCATION,
        body=body,
        validate_body=GCP_VALIDATE_BODY
    )
    invoke_task = GcfFunctionInvokeOperator(
        task_id="invoke_task",
        project_id=GCP_PROJECT_ID,
        location=GCP_LOCATION,
        input_data={},
        function_id=GCF_SHORT_FUNCTION_NAME
    )
    delete_task = GcfFunctionDeleteOperator(
        task_id="gcf_delete_task",
        name=FUNCTION_NAME
    )
    deploy_task >> deploy2_task >> invoke_task >> delete_task
```

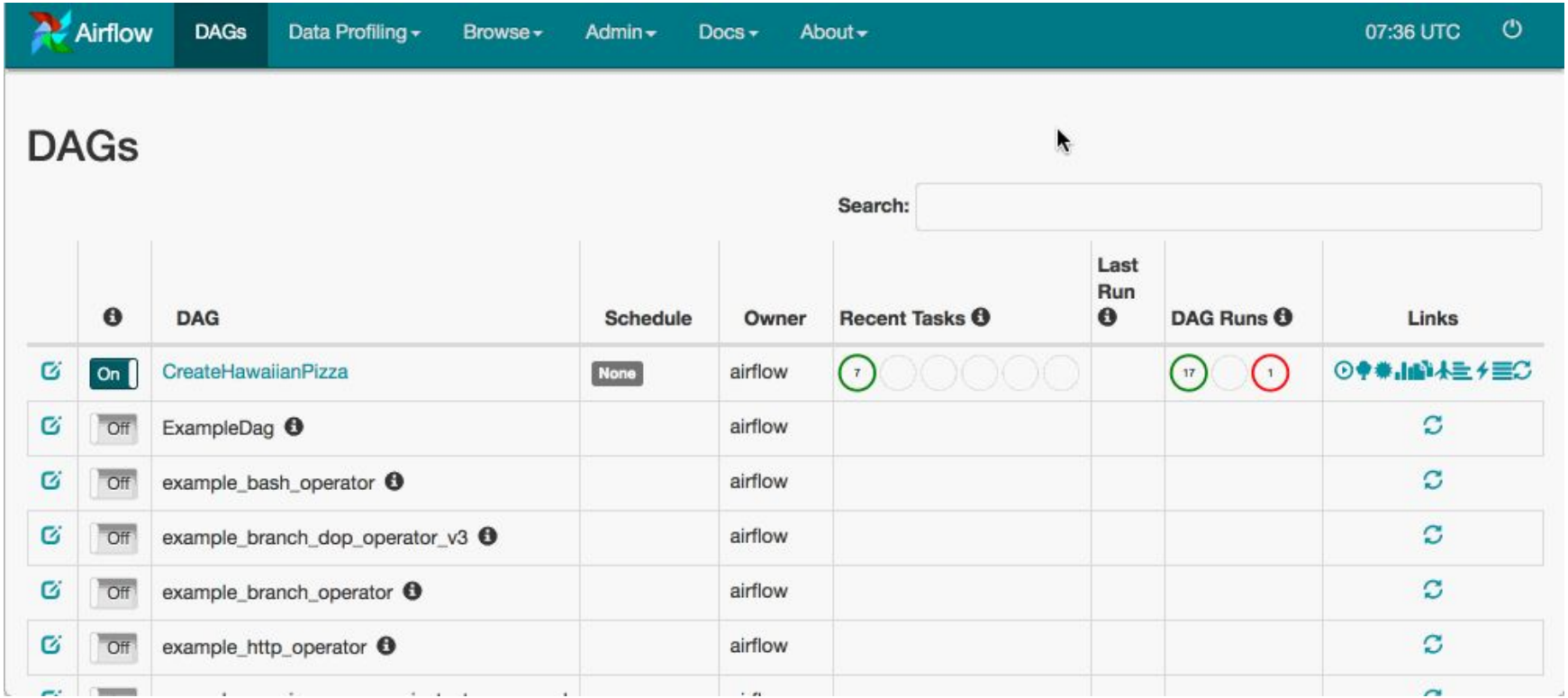




Arbitrary complex workflows as a program



Airflow has usable UI



Airflow CLI

```
usage: airflow [-h]
               {connections,dags,db,flower,kerberos,pools,roles,rotate_fernet_key,s
               ...

positional arguments:
  {connections,dags,db,flower,kerberos,pools,roles,rotate_fernet_key,scheduler,serv
                        sub-command help
    connections         List/Add/Delete connections
    dags                List and manage DAGs
    db                  Database operations
    flower              Start a Celery Flower
    kerberos            Start a kerberos ticket renewer
    pools               CRUD operations on pools
    roles               Create/List roles
    rotate_fernet_key   Rotate all encrypted connection credentials and
                        variables; see
                        https://airflow.readthedocs.io/en/stable/howto/secure-
                        connections.html#rotating-encryption-keys.
    scheduler           Start a scheduler instance
    serve_logs          Serve logs generate by worker
    sync_perm           Update permissions for existing roles and DAGs.
    tasks               List and manage tasks
    users               List/Create/Delete/Update users
    variables           CRUD operations on variables
    version             Show the version
    webserver           Start a Airflow webserver instance
    worker              Start a Celery worker node

optional arguments:
  -h, --help            show this help message and exit
```

What Airflow shines at ?

- Regular batch ETL jobs (think CRON)
- Processing fixed intervals of data
- Managing complex dependencies
- Backfilling data
- Interfacing to hundreds of different systems
- Platform for others to generate DAG files

Apache Airflow 1.10

state of the pinwheel

Current versions

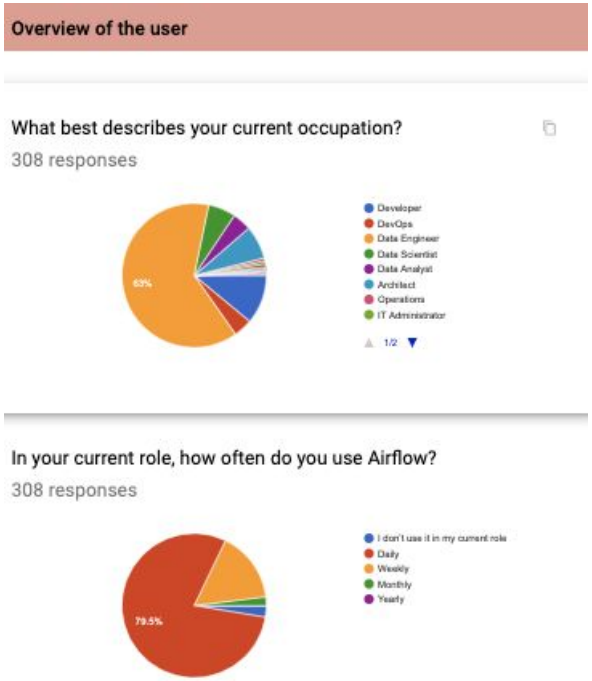
- 1.10.2, 1.10.3, 1.10.4, 1.10.5, 1.10.6 ....
- 1.10.7 in the making
- Deployed in thousands of companies
- On the rise of usage
- 2.0 - in master

How to stay relevant ?

- Cloud Native is coming
- APIs are backbone of modern software
- User Interface matters
- Performance and reliability matter
- Many services, many changes
- Community over code

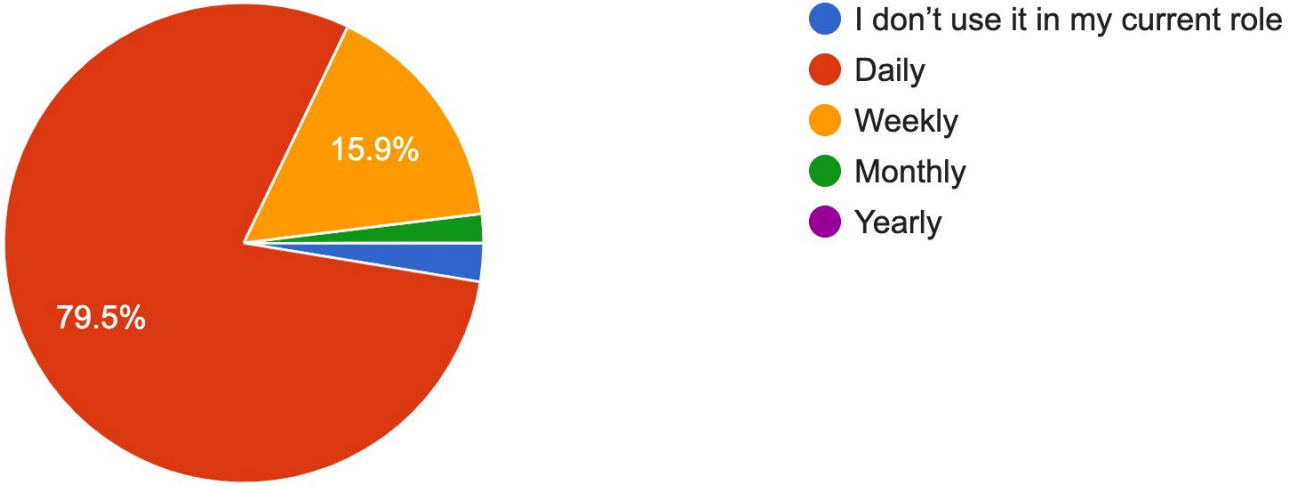
End of 2019 survey: 300 responses(!)

- Started by Tomasz Urbaszek
- Run for the last 2 weeks
- Fresh off-the press
- Some surprises found
- Going in the right direction



In your current role, how often do you use Airflow?

308 responses



What do you use Airflow for?

Data processing (ETL)	97%
Artificial Intelligence and Machine Learning Pipelines	29%
Automating DevOps operations	21%

What can be improved ?

Scheduler performance	61%
Web UI	58%
Logging, monitoring and alerting	47%
Examples, howtos, onboarding documentation	46%
Technical documentation	44%
Reliability	36%
REST API	31%
Authentication and authorization	29%

What would be the most interesting feature for you ?

Production-ready docker image	56%
Declarative way of writing DAGs	50%
Horizontal autoscaling	40%
Examples, howtos, onboarding documentation	46%
Asynchronous Operators	31%
Stateless web server	26%
Knative Executor	16%
I already have all I need	4%

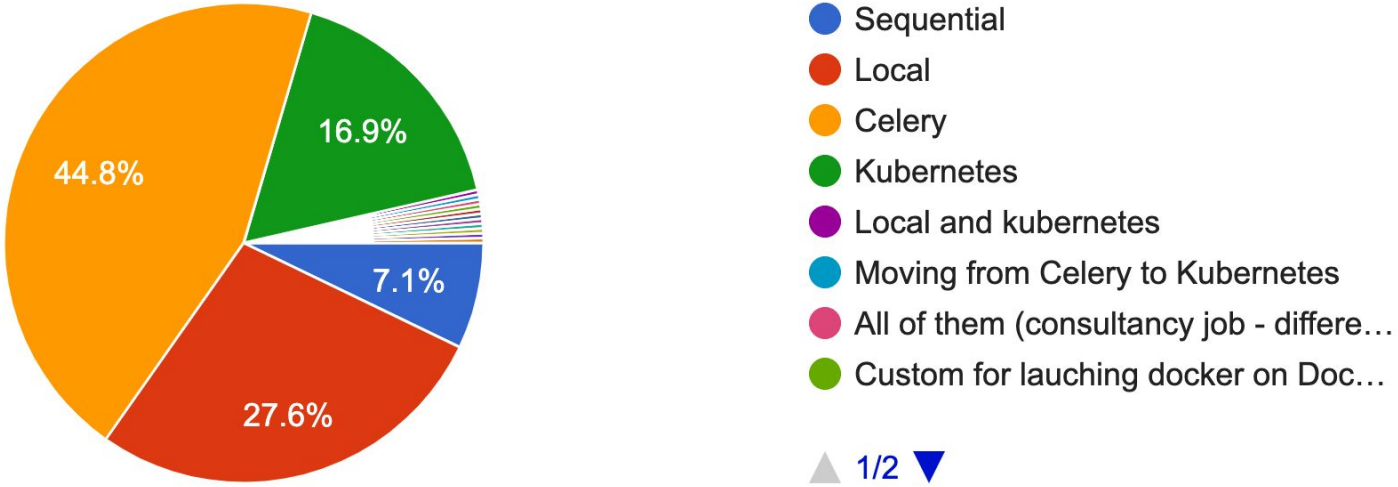
Apache Airflow 2.0

Polidea

Cloud Native is coming

In Airflow, what executor do you use?

308 responses



Do you use Kubernetes-based deployments for Airflow?

No - we do not plan to use Kubernetes near term	29%
Yes - setup on our own via Helm Chart or similar	21%
Not yet - but we use Kubernetes in our organization and we could move	20%
Yes - via managed service in the cloud (Composer/Astronomer etc.)	15%
Not yet - but we plan to deploy Kubernetes in our organization soon	14%
Other	2%

Either use or can use Kubernetes in foreseeable future	69%
Do not have plans to use Kubernetes	29%

Cloud Native is coming: Scalability

- ~~Knative Executor~~
- SIG-Knative => SIG Scalability
- Native Airflow Executor (WIP)
- Pub/Sub communication
- Horizontally auto-scalable

Cloud Native is coming: Deployability

- Native worker deployable at different providers
- “As a service” and “on-premises” friendly
- Generic Pub/Sub architecture for communication
- No DB communication between components
- Production-optimised docker image

Cloud Native is coming: Monitoring

- Integrate with standard monitoring tools
- More metrics exported using stats
- Integration with Prometheus on Kubernetes
- Horizontal Scalability approach based on metrics

APIs are backbone of modern software

APIs are taking over the world

- Modern API
- HTTP-based API used by CLI, webserver
- Pub/Sub API for communication Scheduler <> Workers
- Generic APIs - not tied to Kubernetes/other deployment options
- Better Authentication/Authorization
- Opens up multi-tenancy capabilities

User Interface matters

Which interface(s) of Airflow do you use as part of your current role?

Original Airflow Graphical User Interface	97%
CLI	40%
API (experimental)	20%
Custom Own Created UI	8%

UIs are getting better

- Make UI refresh like it’s 2020
- Modern design (possibly)
- Use APIs for communication not DB/file access
- Better authentication and authorisation
- Stateless web-server
- Better responsiveness

Performance and reliability

matter

Performance and reliability is important

- Automated performance testing (CI - targeted)
- Monitoring performance characteristics
- Improve Webserver/Scheduler Performance
- Internal instrumentation and optimisations

Many services, fast changes

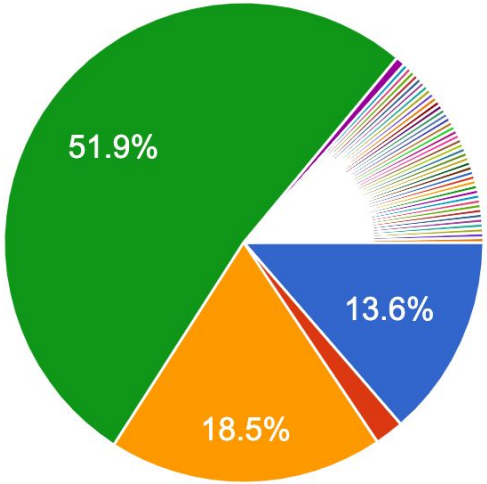
Fast evolving services

- Currently operators bound to releases of Airflow
- Migration to 2.0 will take time
- Introducing new approach
  - move operators to new path/namespaces
  - change import paths
  - backporting to 1.10
  - backportable to 1.10 (!)
  - future: per-provider packaging

Community over code

When onboarding new members to airflow, what is the biggest problem?

308 responses



- Documentation is not clear enough
- Small number of blogs regarding Air…
- Small number of tutorials on differe…
- No guide on best practises on devel…
- OK
- Execution Date
- Understanding the scheduler
- There is alot of things not covered b…

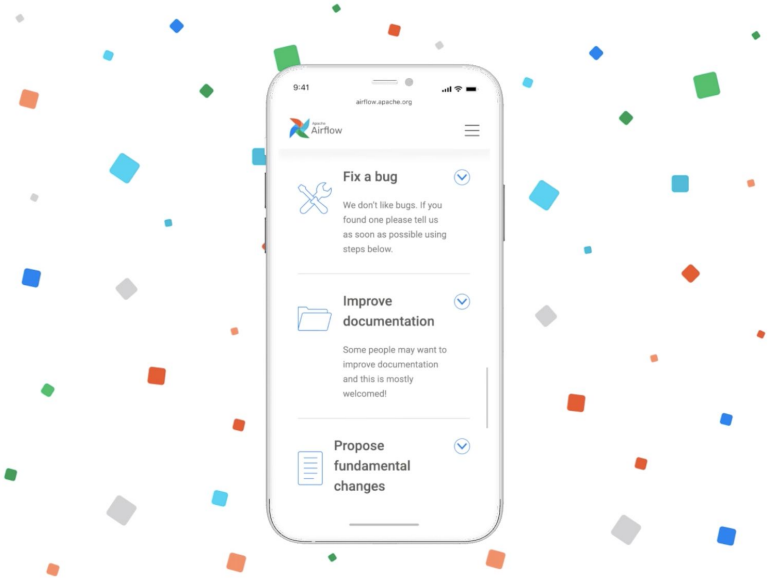
1/6

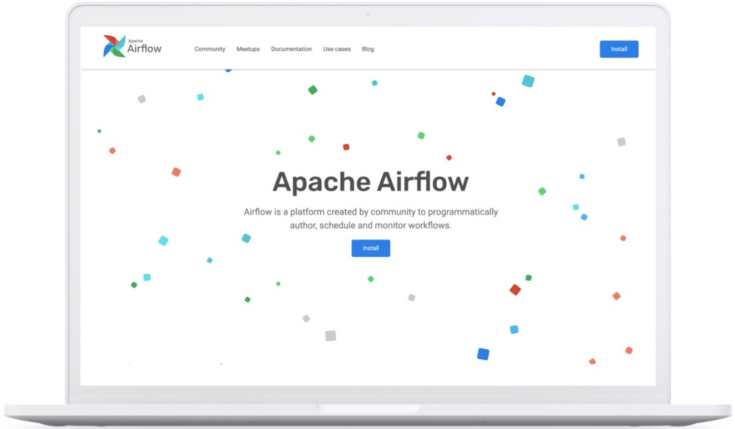
Community over code: Documentation

- Google Season of Docs - great programme!
- Onboarding, best practices, architecture, deployment options
- Better, clearer structure
- Both user and developer documentation improved
- Worked with technical writers from India and Russia

Community over code: New website: airflow.apache.org

Work sponsored by Google Cloud





Community over code: Development environment

- It’s a Breeze to develop Apache Airflow
- Get your environment up in 10 minutes
- Integration with IDE
- Well documented
- Team-work enabler
- Allows to run and debug DAGs
- Fully debuggable: DebugExecutor - cooperation with Databand.ai

Workshop for first time
contributors to Apache Airflow

It's a Breeze to contribute to Apache Airflow

Polidea

AIRFLOW BREEZE

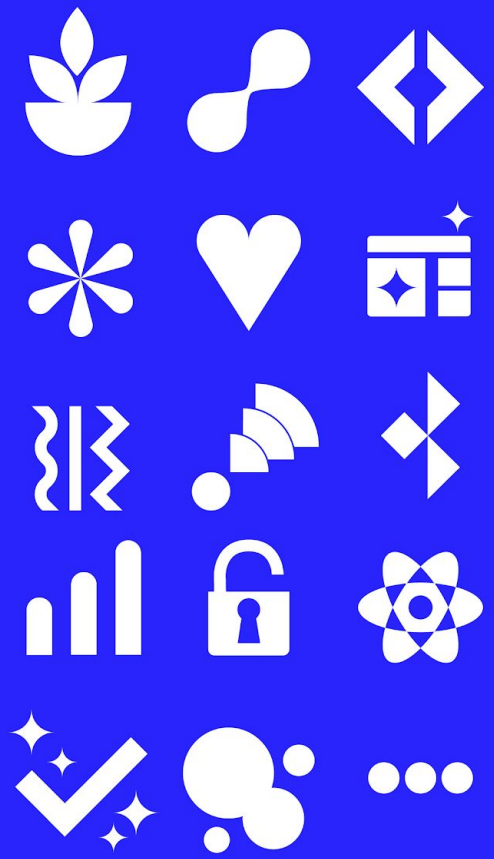
Apache Airflow

First Warsaw Apache Airflow Workshop

Friday, December 13, 2019
5:30 PM to 9:30 PM

Polidea Sp. z o.o.
Przeskok 2 IV p. · Warsaw

https://t.co/TmWdWwfeml

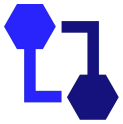


Thanks!

Polidea

hello@polidea.com

Thanks!



Prototypes

Widely Used

Personal Growth

Individuality

Trust

React Native

Testing

Team

Mobile AR

Research

Seamless UX

Task

Flutter

UX Design

Beter, Fester

Management

Quality

iOS

Maintain

Security

Front End

Ideation

Shipping

Decision

Open Source

Save money

CV

Contact

Collaboration

API Design

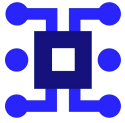
Building blocks

Backend

VR

Android

Learn more



Firmware

Coffee