



GitHub's Airflow Journey

Lessons, Mistakes, and Insights

Oleksandr Slynko

3.0

Airflow at GitHub

1000

active DAGs

70

teams

50k+

tasks per day

GitHub is running Airflow for 9 years

Since 9 years ago

GitHub

- Got acquired by Microsoft
- Introduced GitHub Actions
- Acquired npm, Dependabot and CodeQL
- Introduced GitHub Copilot

Airflow

- Added CHANGELOG
- Added Kubernetes executor
- Graduated from Apache incubator
- Extracted DAG processor from scheduler
- Added RBAC and audit

Since 9 years ago

GitHub

- Got acquired by Microsoft
- Introduced GitHub Actions
- Acquired npm, Dependabot and CodeQL
- Introduced GitHub Copilot

- Got 18k pull requests

Airflow

- Added CHANGELOG
- Added Kubernetes executor
- Graduated from Apache incubator
- Extracted DAG processor from scheduler
- Added RBAC and audit

- Got 37k pull requests



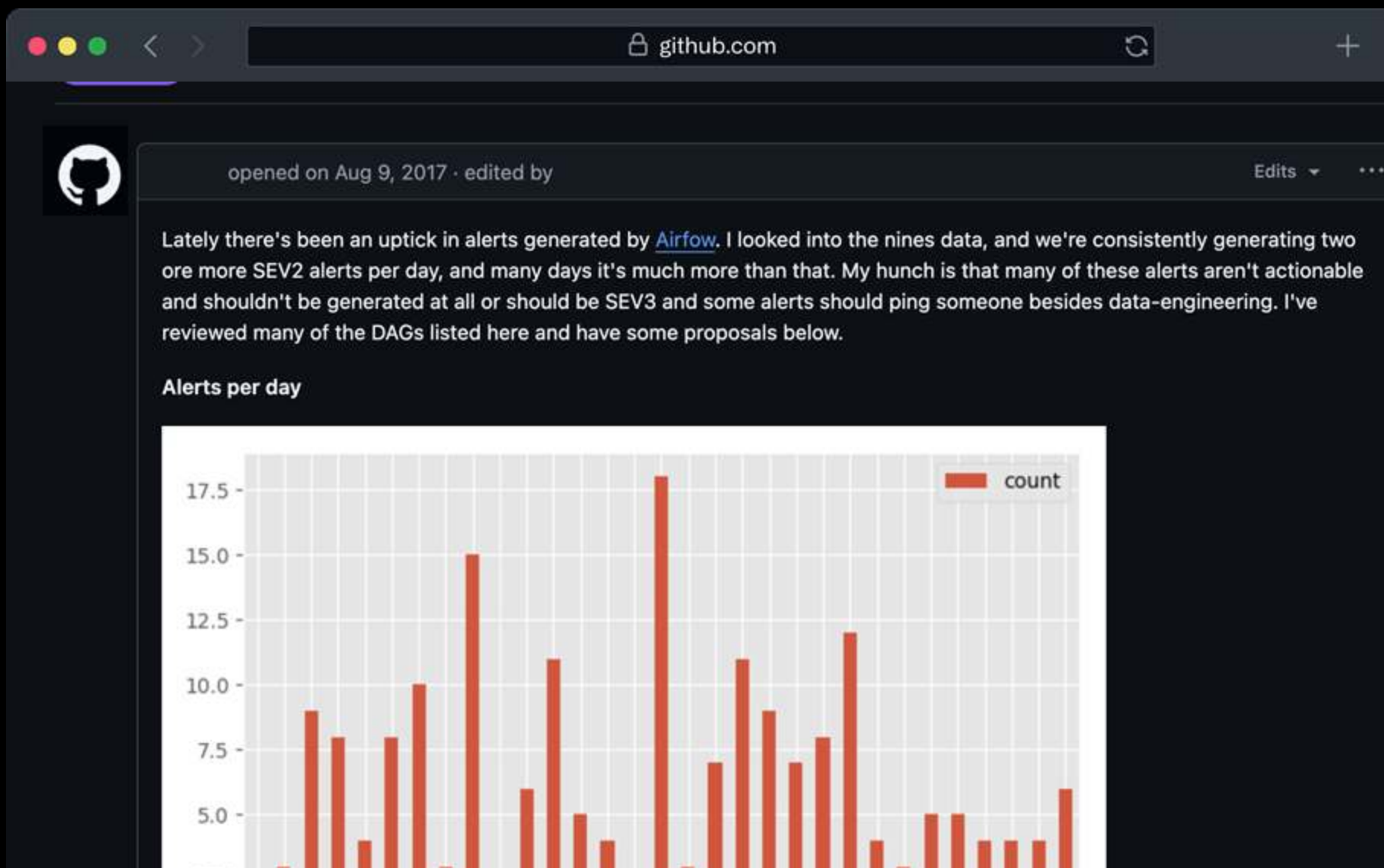
11 lessons about growing Airflow usage

How it started

- Needed to get some data for billing in June 2016
- Added more and more DAGs
- Everything is critical and important

What if we monitor all DAGs for failures?

Getting too many alerts



**What if we monitor ~~all~~ some
DAGs for failures?**

Lesson 0

DAG monitoring is not enough

- You need to monitor underlying infrastructure
- Possibly monitor that DAGs finish at expected time

**We need more data for
analysis**

**We need more data for
analysis**

Let's add more people

Lesson 1

ETL is too much for a single team

Self-serve for ETL



Make sure code has owners

- Use CODEOWNERS file
- Data owners are not always codeowners
- Someone who will debug and fix the issue when DAG fails
- Someone who can approve code
- Test it in CI and Airflow continuously

Follow Airflow best practices



Provide example DAG

- No one wants to learn the code, it is easier to copy
- Keep it up-to-date and running in your cluster

Provide example DAG

- No one wants to learn the code, it is easier to copy
- Keep it up-to-date and running in your cluster
- Fix deprecations

Provide example DAG

- No one wants to learn the code, it is easier to copy
- Keep it up-to-date and running in your cluster
- Fix deprecations
- Do not create custom DAG class

Lesson 4

Do linting and code format

- ruff
- SQLFluff
- Unit tests for required params in DAGs



Lesson 5

Make backfills simple for users

- DAG to do backfills of other DAGs with all the steps and configs

Lesson 6

Check operators and connections

- Know your connection specifics and issues and test them



20%

**Increase in number
of PRs to Airflow
DAGs**

Year comparison, excluding library
code



100%

**Increase in number
of issues with
Airflow DAGs**

10%

increase for issues for our team

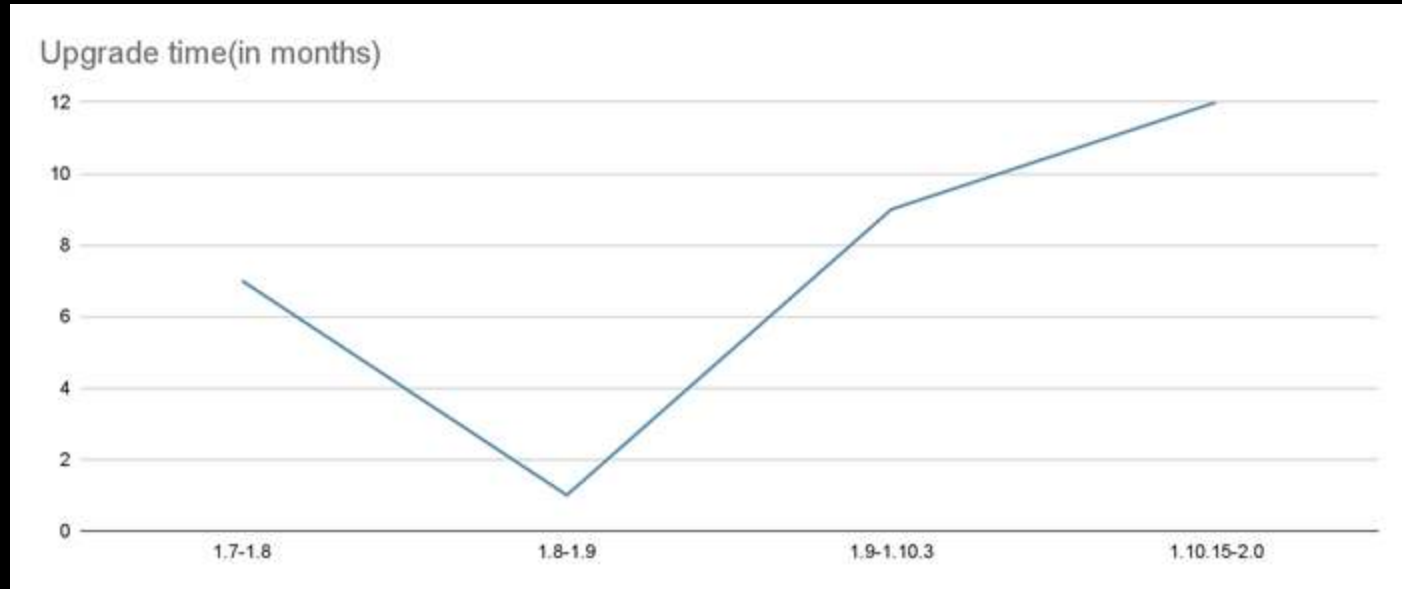
30%

more DAGs

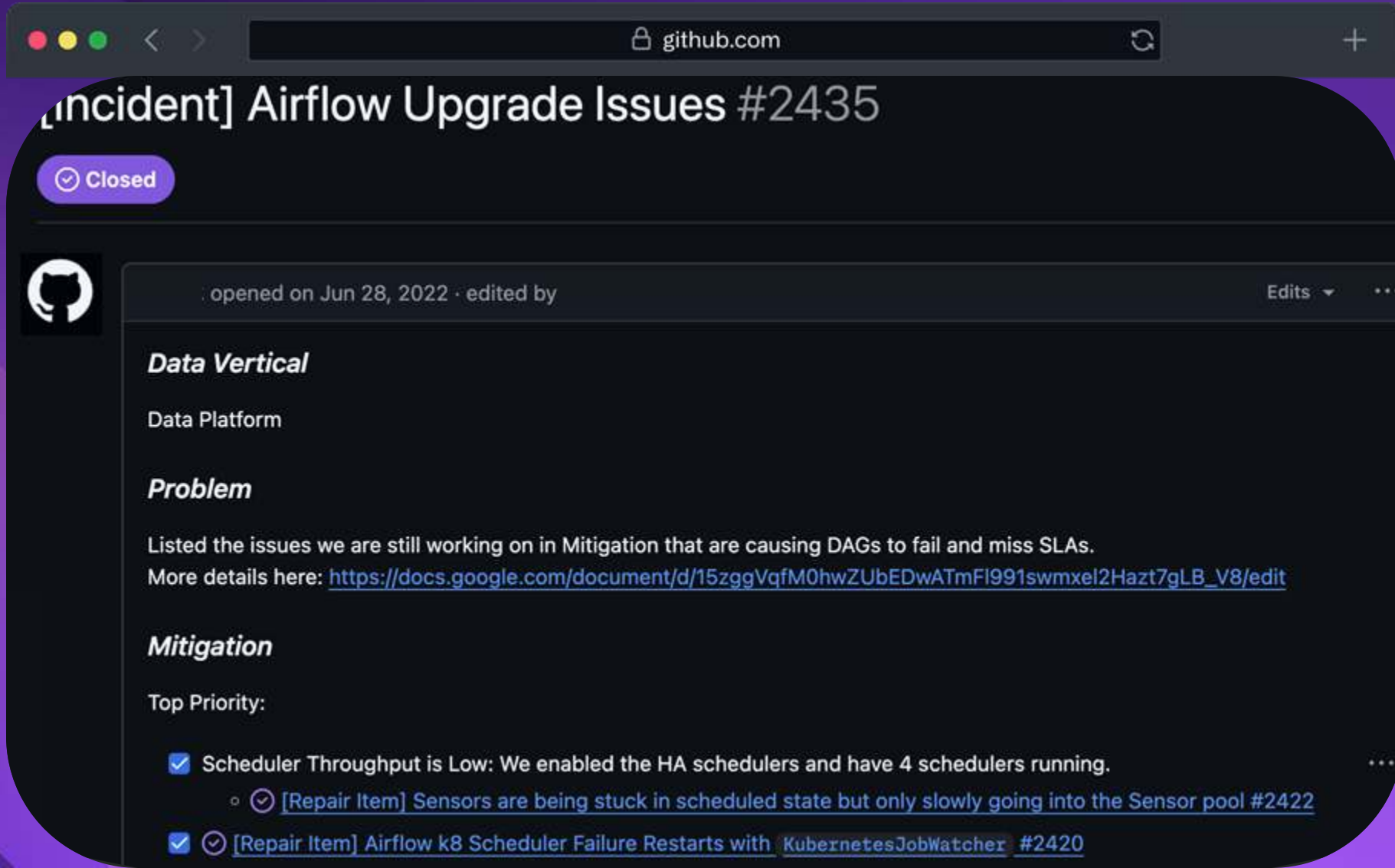
Running a platform



Upgrades



Year long upgrade



The screenshot shows a GitHub issue page for '[Incident] Airflow Upgrade Issues #2435'. The issue is marked as 'Closed' with a green checkmark icon. It was opened on Jun 28, 2022, and edited by the user 'Data Vertical'. The issue is categorized under 'Data Platform'. The 'Problem' section states that the issue is related to DAGs failing and missing SLAs, with a link to a Google document for more details. The 'Mitigation' section lists two items: 'Scheduler Throughput is Low' and 'Airflow k8 Scheduler Failure Restarts with KubernetesJobWatcher'. Both items are marked as resolved with green checkmarks.

github.com

[Incident] Airflow Upgrade Issues #2435

Closed

opened on Jun 28, 2022 · edited by

Data Vertical

Data Platform

Problem

Listed the issues we are still working on in Mitigation that are causing DAGs to fail and miss SLAs.
More details here: https://docs.google.com/document/d/15zggVqfM0hwZUbEDwATmFI991swmxel2Hagt7gLB_V8/edit

Mitigation

Top Priority:

- ✓ Scheduler Throughput is Low: We enabled the HA schedulers and have 4 schedulers running.
 - ✓ [Repair Item] Sensors are being stuck in scheduled state but only slowly going into the Sensor pool #2422
- ✓ [Repair Item] Airflow k8 Scheduler Failure Restarts with [KubernetesJobWatcher](#) #2420

Lesson 7

Use open source knowledge

- Do not copy just code
- Contribute back PRs and issues



Lesson 8

Simplify setup and development

- Codespaces/devcontainers - same setup inside container

Ephemeral dev environments

- One staging is not enough
- It is hard to test upgrades on single environment



alex-slynko 1:11 PM

.airflow canary <https://github.com/github/airflow-sources/pull/42018>

x

#42018 Use different image for postgres

Overview [🐱](<https://github.githubassets.com/images/icons/emoji/octocat.png> "

🐱")

This PR replaces postgres image in hope it will fix canary.

Validations ✅

Post-PR Checklist

- If any files **outside** of the `dags` directory were modified, run `.airflow deploy` in `#data-engineering-ops`.

🔄 [github/airflow-sources](#) Sep 18th Added by GitHub



Hubot APP 1:11 PM

Canary deployment started for <https://github.com/github/airflow-sources/pull/42018> by alex-slynko.



FlowBee APP 1:11 PM

@alex-slynko is rolling out an airflow canary for `replace-postgres` using image tag `downstream-0dd30cd`. This may take several minutes.

@alex-slynko's deploy of `replace-postgres` to `canary` is complete. The canary is available at <https://airflow-canary.githubapp.com/alex-slynko>

Lesson 10

Simplify CI

- It should be easy to change Airflow versions
- The test setup should be simple

Lesson 11

Upgrade packages

- Use Dependabot

Time to major upgrade

4

months

Reduced issues by

20%

PR number increased by

30%

Things to do



Test operators and
connections in DAG



Make sure you have dev
environment to test



Fix deprecations

Things to do

Contribute to Airflow



Test operators and
connections in DAG



Make sure you have dev
environment to test



Fix deprecations



Thank you



@alex-slynko