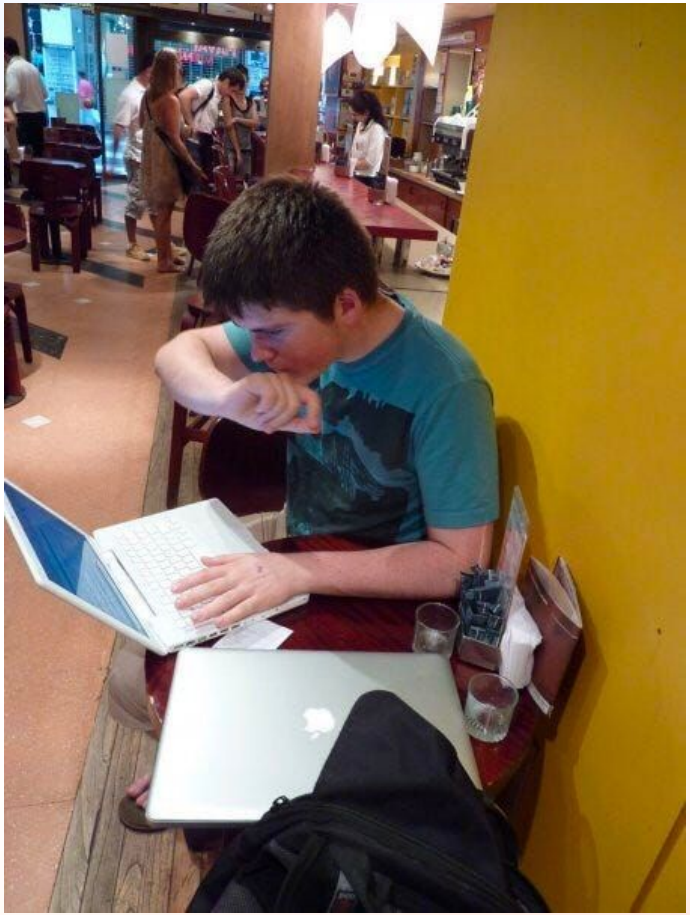

# Do you trust Airflow with your money? We do*!

3.0

Sabrina Liu
Senior Software Engineer, Stripe

# In 2010, Stripe was a hole in the wall



John Collison, cofounder of Stripe, in 2010

Global scale

# The backbone for global commerce

Stripe makes moving money as easy and programmable as moving data. Our teams are based in offices around the world and we process hundreds of billions of dollars each year for ambitious businesses of all sizes.

**500M+**

API requests per day, peaking at 13,000 requests a second.

**99.999%**

historical uptime for Stripe services.

**90%**

of U.S. adults have bought from businesses using Stripe.

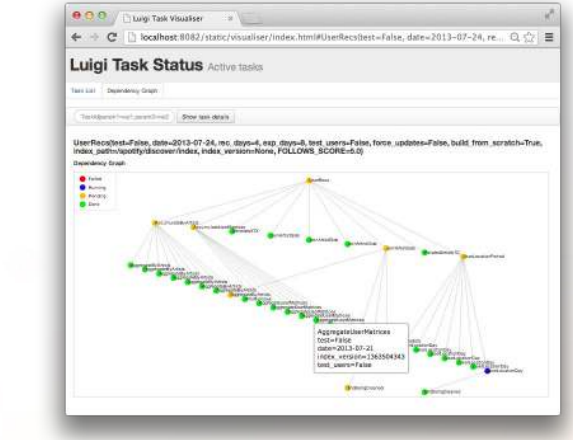
**135+**

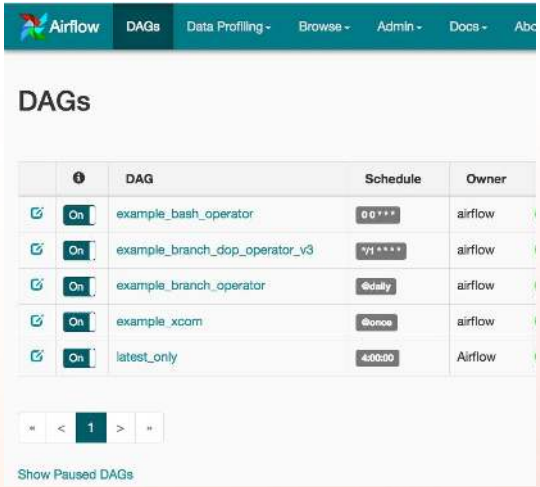
currencies and payment methods supported.

## Mid-2010s

## 2017 - present

What are data pipelines??




stripe

# Airflow at Stripe in 2025

**15,000**
unique Task classes

**180,000**
unique Tasks instantiated

**10 million**
daily task instance executions

stripe

stripe

# 2 Airflow clusters

# Why Airflow 1, 2, or 3 out of the box don’t solve our problems

We have an **UberDAG** (not you, Dara)

I want to reuse code across the **monorepo** like a good engineer

**Just run** my workload

Let my task live for **3 days**

# What we've built

### Airflow but easier

Low-code + no code orchestration on top of Airflow

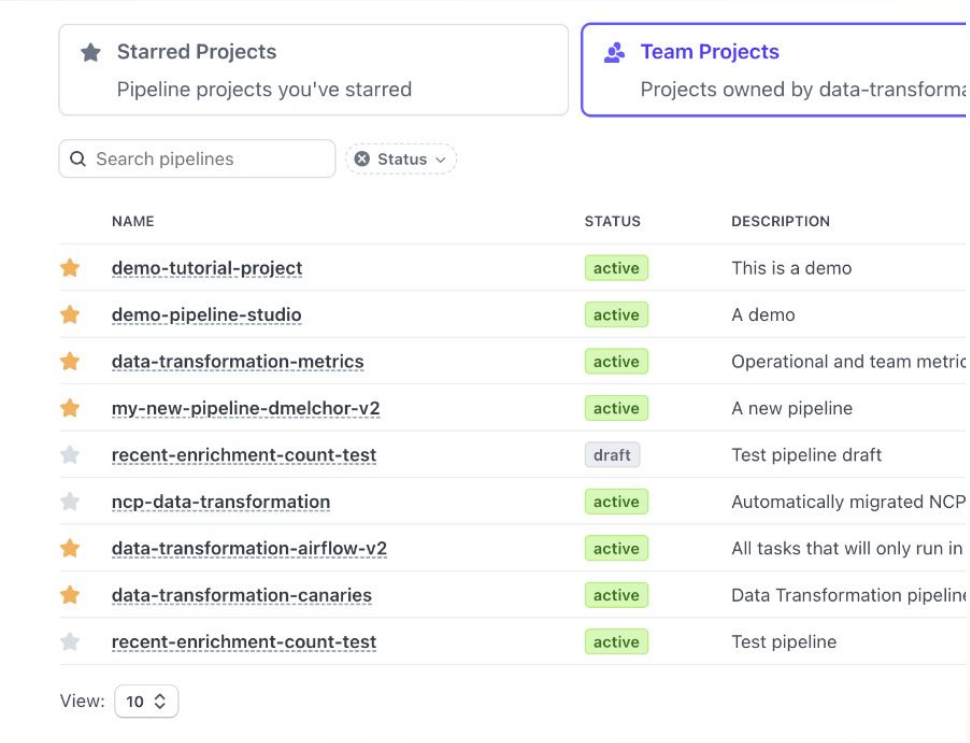
### Local testing

User Scoped Mode (USM)

### Multitenancy

Project-level isolation and reliability on a single Airflow cluster

# Airflow but easier



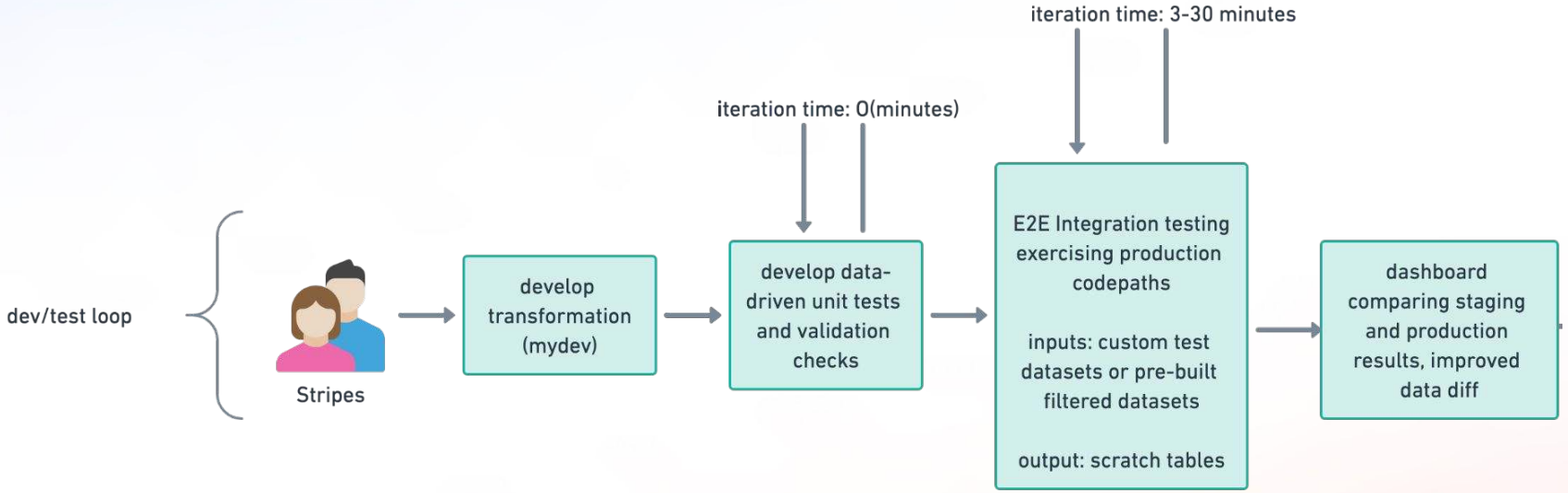

**Simple Pipelines:**

yaml-based task authoring using SparkSQL

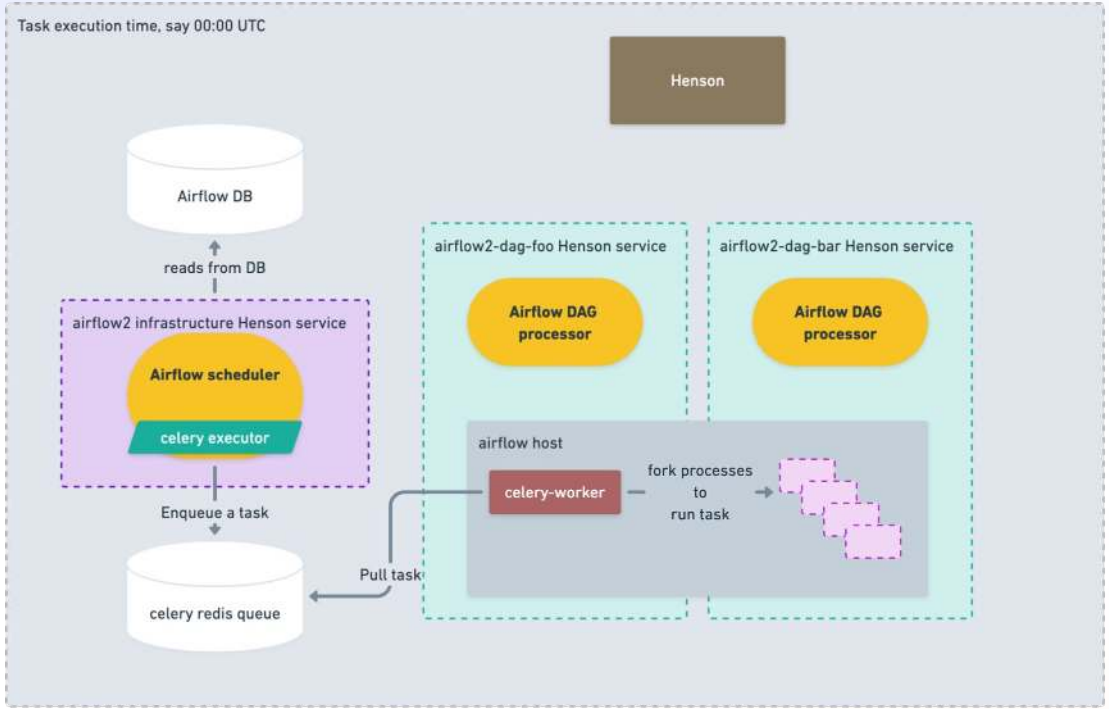
**Pipeline Studio:**

no code task authoring using SQL

# User Scope Mode




Key: production inputs, non-production outputs




## Improvements

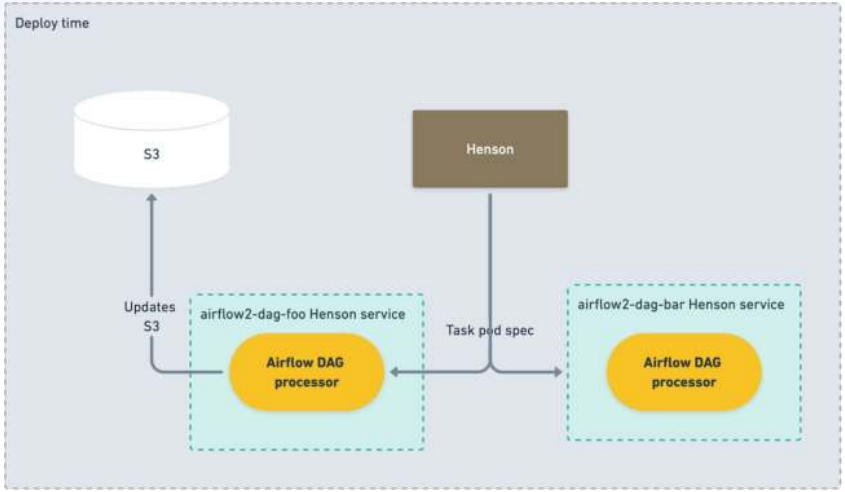
- 1 project : 1 service : 1 dag processor : 1 owning team

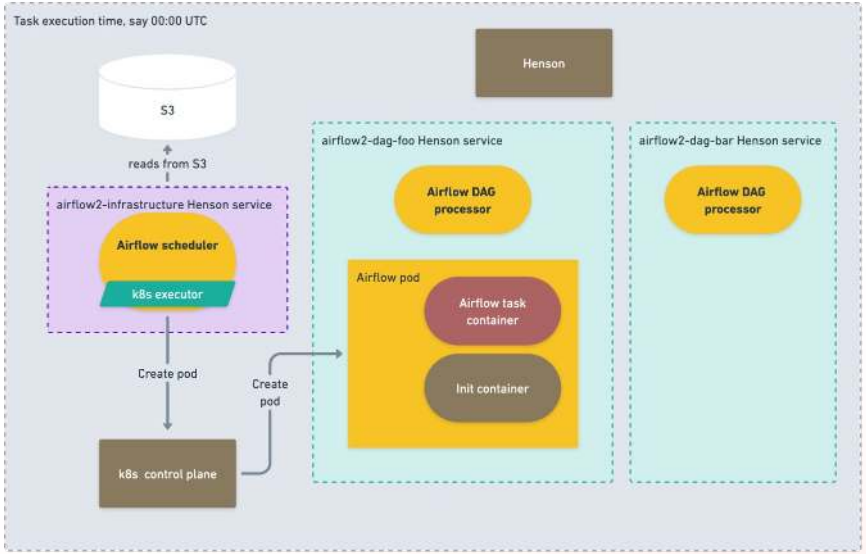
## Weaknesses

- Hard to scale deployments
- Manual Celery shard management
- Shared workload identity

# Step 1 to multitenancy with EC2 and Celery

# Multitenancy with KubernetesExecutor




Independent execution and configuration

# Customizations with the KubernetesExecutor

- **[scalability]** Multi-shard support
  - One executor per Kubernetes shard
  - We add a `cluster_context` attribute to the `kube_client`
  - Dedicated Kubernetes shard for Airflow workloads
- **[compliance]** SOX controls for workloads that touch financial data
  - Separate Kubernetes namespaces
  - Lifecycle management for long-running, stateful workloads
- **[efficiency]** Semi-managed compute
  - API for requesting CPU and memory
  - Automatic bin-packing for short-lived tasks

… and much more!

# … but we haven’t won yet!


Sabrina Liu

Senior Software Engineer

LinkedIn:



Sharadh Krishnamurthy

Engineering Manager

LinkedIn: