AIRFLOW SUMMIT

How We Taught an Agent Airflow:

Building Self-Healing Pipelines with Otto







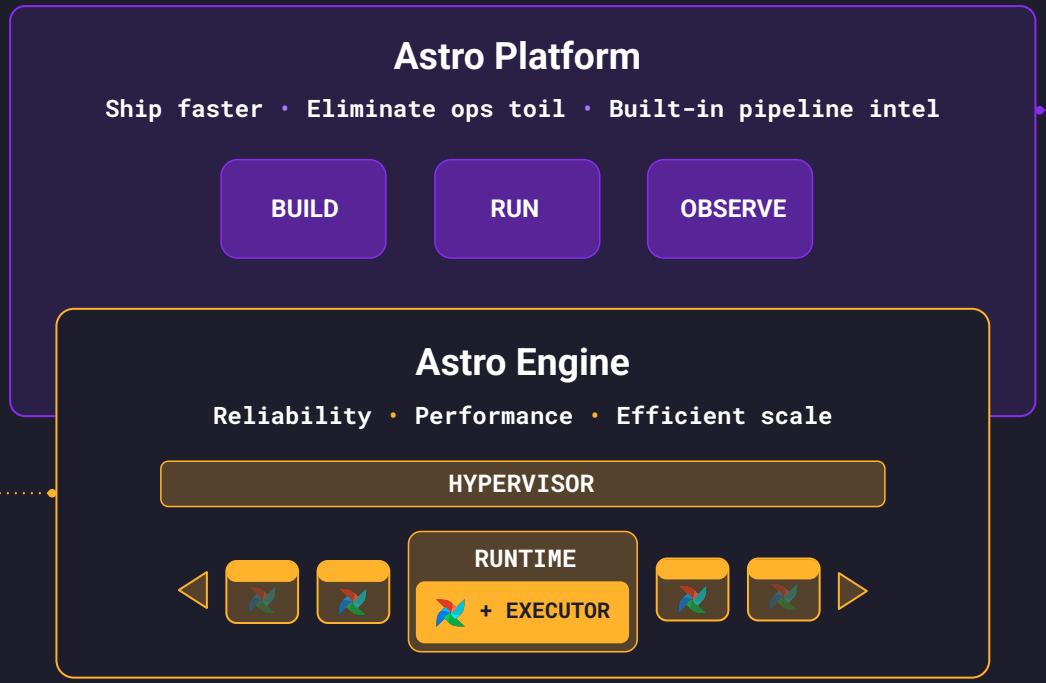
Christine Shen, Stephanie Niu

Astronomer provides the commercial Airflow platform for unified orchestration.

2x parallel load per worker ensures your pipelines stay reliable under load spikes

30%+ more DAG runs for the same spend (compared to alternatives) keeps costs down

3-4x faster task starts means fresher data, faster



10x Delivery Speed with AI-driven developer tools

Intelligent auto-scaling of task-optimized workers

Faster RCA with built-in lineage, asset catalog, and AI log summaries

Pipeline failures are a weekly tax

ACCORDING TO INDEPENDENT RESEARCH

97%

of data leaders say pipeline failures have slowed their AI initiatives

53%

of engineering capacity goes to maintaining pipelines

13 hrs

to resolve one failure. 60+ hours of downtime a month.

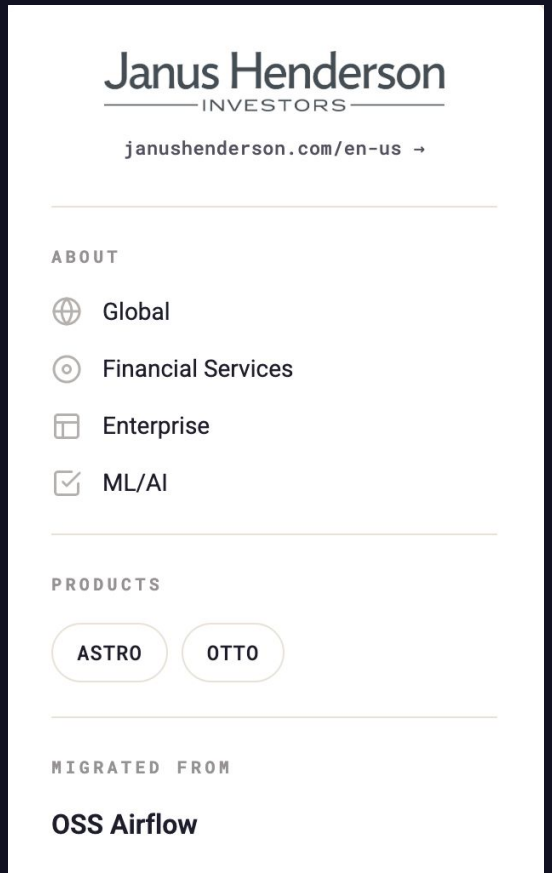
Fivetran, Enterprise Data Infrastructure Benchmark 2026 — 500 senior data leaders at 5,000+ employee enterprises, Q4 2025.

FROM OUR OWN FLEET

The median org has **13 dag failures** a day.

ASTRONOMER

Failures cost more than time



> “We get email alerts at half past two in the morning. And we’re the only team responsible. If we don’t get in until 8, then it’s already too late, especially if we’ve got downstream processes that are already running.”
>
> Kieran Rajasansir, Quantitative Developer
> Janus Henderson Investors

Broken pipelines cost more than remediation time;
they block critical downstream decisions.

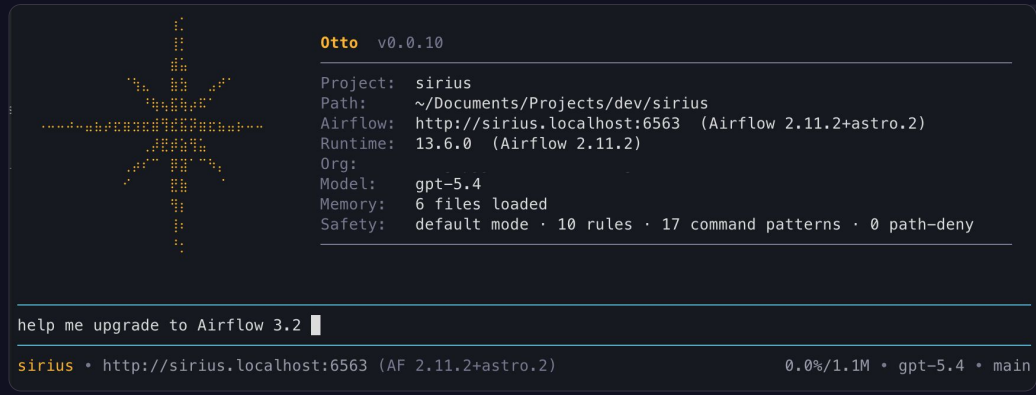
Otto

The only **data engineering agent**, built for Airflow

AVAILABLE IN LABS

- Airflow Specific Capabilities & Skills
- Purpose Built Agent Harness
- Open & Proprietary Knowledge

An Airflow expert, always available wherever you work



ASTRONOMER

Built to accelerate any data engineering workflow

Code & Data Exploration

Understand what data exists, what it means, and how to use it, without leaving your agent session.

Dag Authoring & Debugging

Describe the Dag in plain language. Otto writes it, configures connections, runs a test. Production-quality from day one.

Code Reviews

Catch Dag issues before they merge, automatically, on every pull request. Otto leaves inline comments to help you catch issues.

Failure Investigation

Reads logs, run history, and Dag code. Root cause in minutes, not hours.

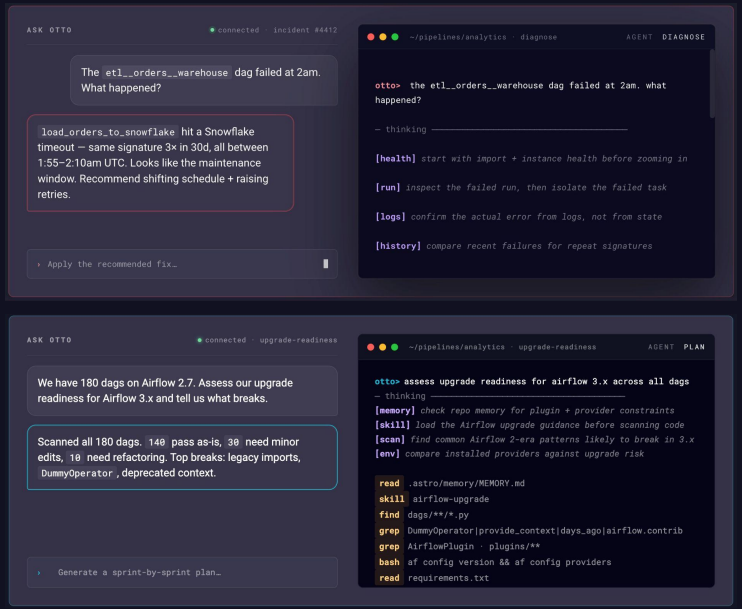
Airflow Upgrades

Scans your Dag fleet against Astronomer's compatibility KB. Identifies what breaks, proposes code changes, builds a plan.

Migrations

EARLY ACCESS

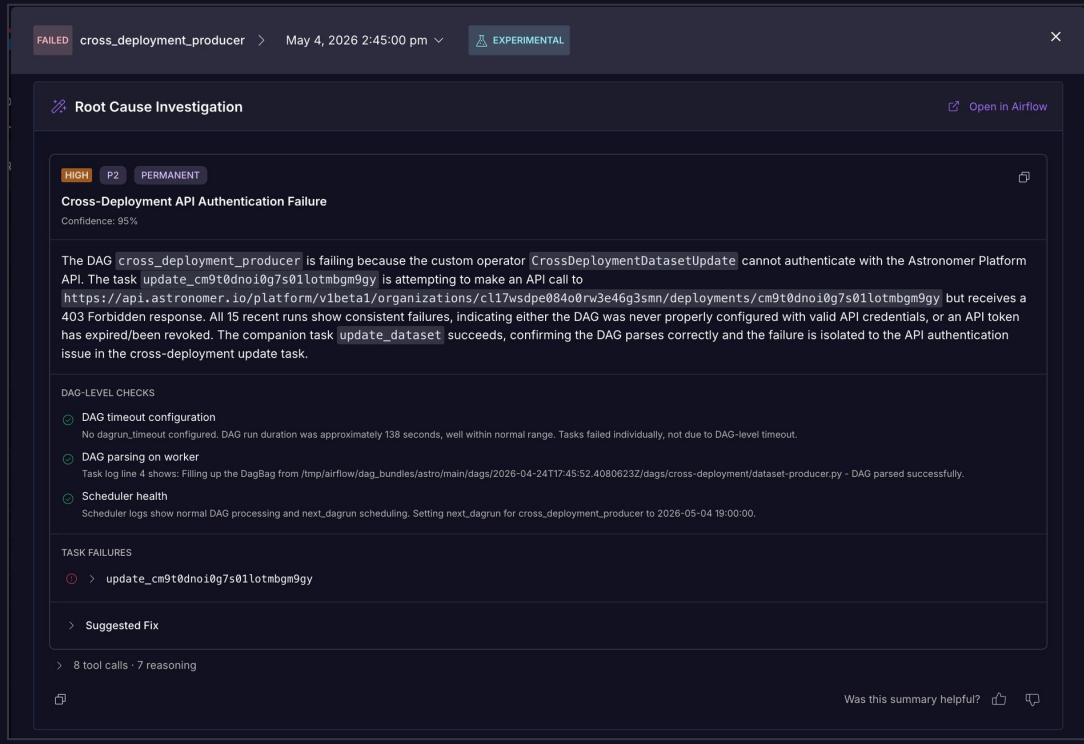
Accelerates workflow-to-workflow translation for Airflow (OSS, MWAA, MSAA) or scheduler migrations (Control-M, Autosys, Automic, Informatica).



AND MORE: Setup and manage Airflow, run dbt projects as Dags, and more. Otto extends through skills it loads on demand — including skills you can author.

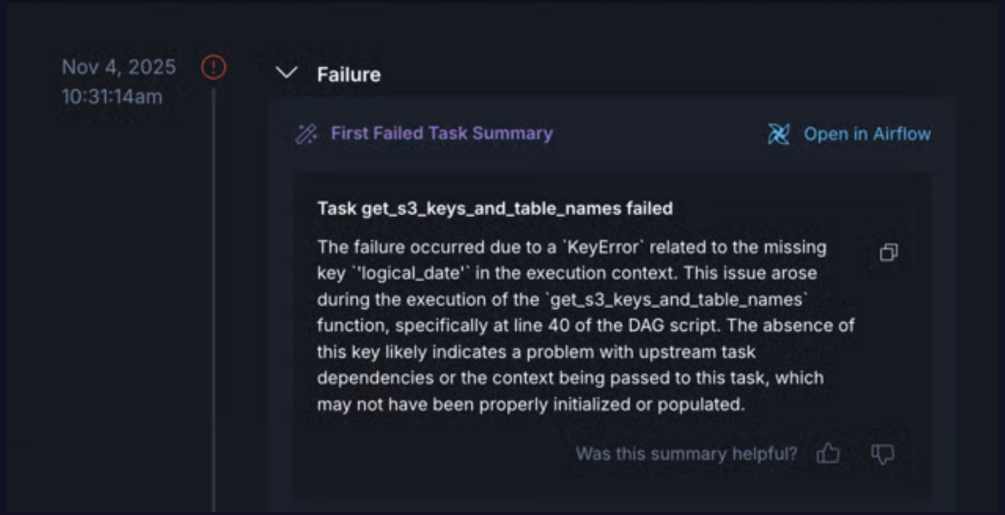
ASTRONOMER

Demo: Otto Investigations



We started where data engineers do

V0 // SUMMARIZE THE TRACEBACK, HAND IT TO THE ENGINEER



THE BET

Read the logs for them

A model that knows Airflow, pointed at the task log.

Cheap to build. Easy to demo. It held up right until we put it in front of real production failures.

V0: Task logs don't tell the whole story

What we shipped first: task logs + "you're an Airflow expert, diagnose this."

Clean Traceback

Demoed beautifully on a clean traceback. But this is the case the engineer didn't need help with: they can read a stack trace.

Missing Logs

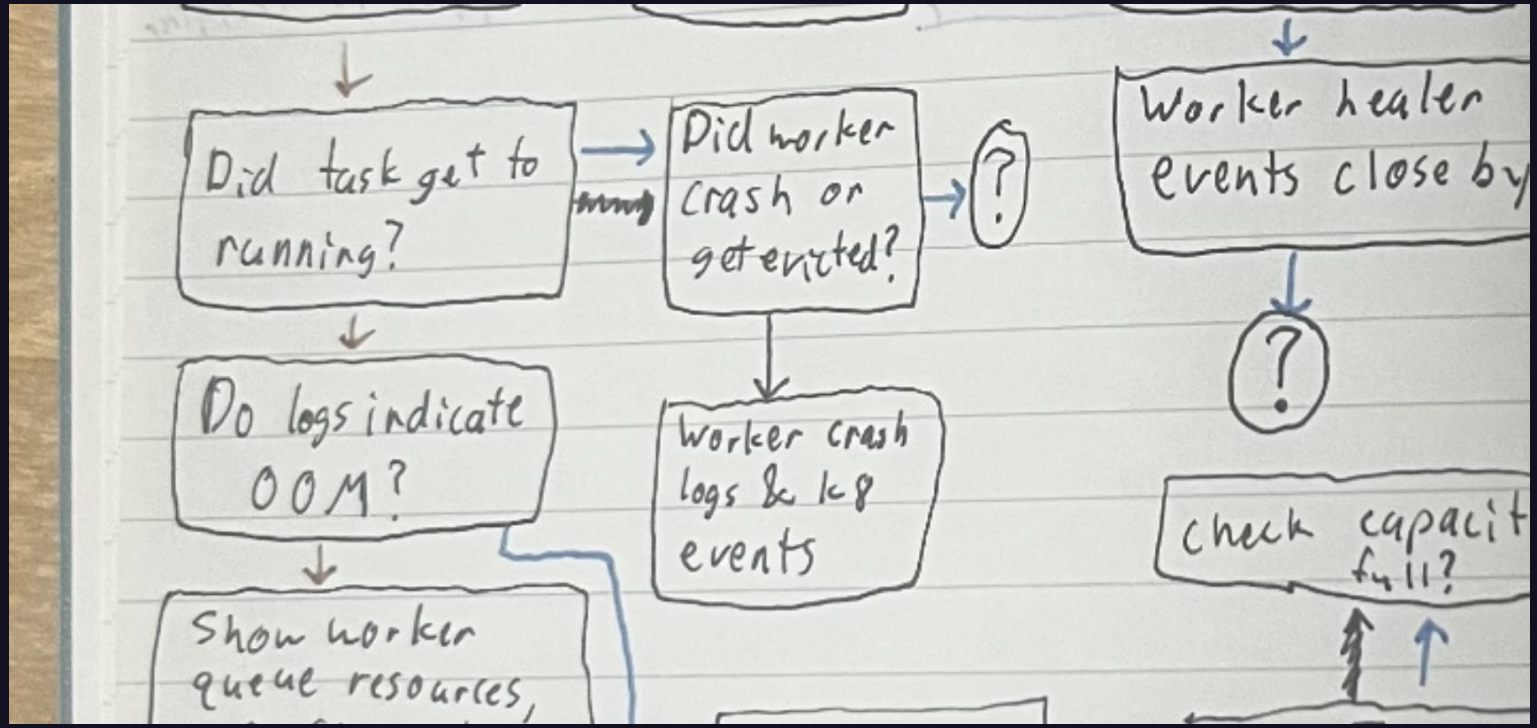
Zombie tasks, OOM kills, stuck in queued, DAGs that never parsed. With nothing to read, it would guess.

Misleading Logs

A task fails parsing a malformed value, where the root cause is an upstream producer that changed shape. The traceback is complete and correct. It confidently reports the ValueError and names the wrong fix. The engineer patches the parser, but it breaks again next week.

✦ Neither is answerable from the task log alone. One needs the execution lifecycle; the other needs everything around the run: history, deploys, lineage

So we asked the experts



Excerpt of our Head of Global Support's diagram of the dag execution lifecycle troubleshooting process.

DAG Execution Lifecycle

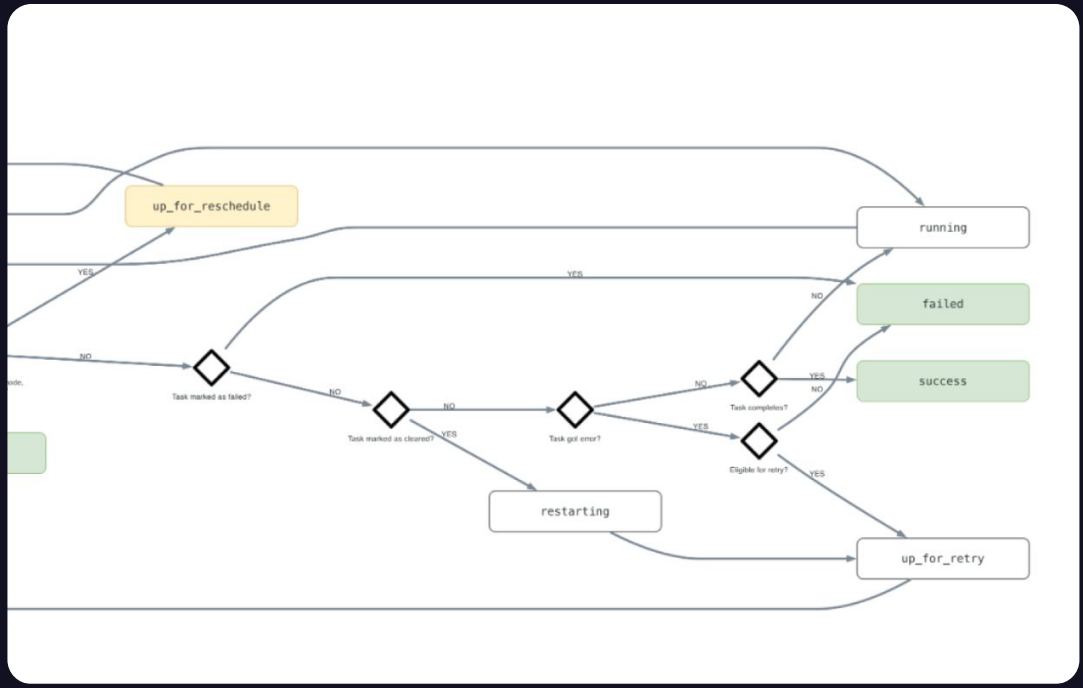
Execution Spine

The execution lifecycle is the backbone of diagnosis. Otto localizes and investigates Airflow failures based on the exact phase of the DAG.

01 Scheduled & Queued

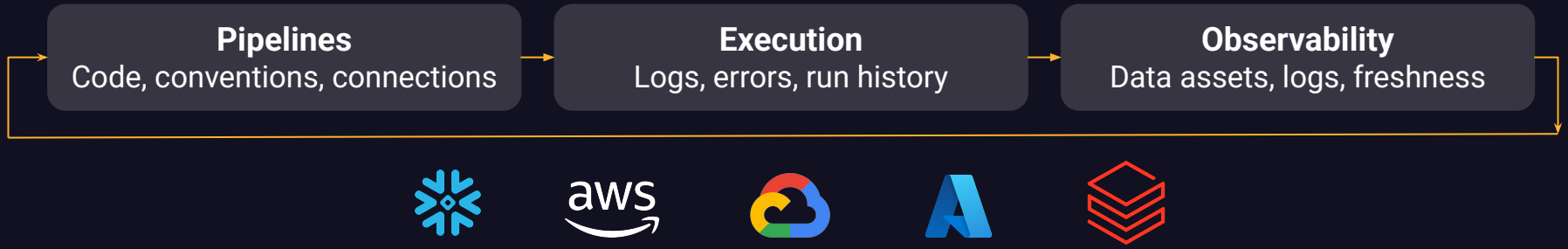
02 Assigned to Worker

03 Parsed & Running



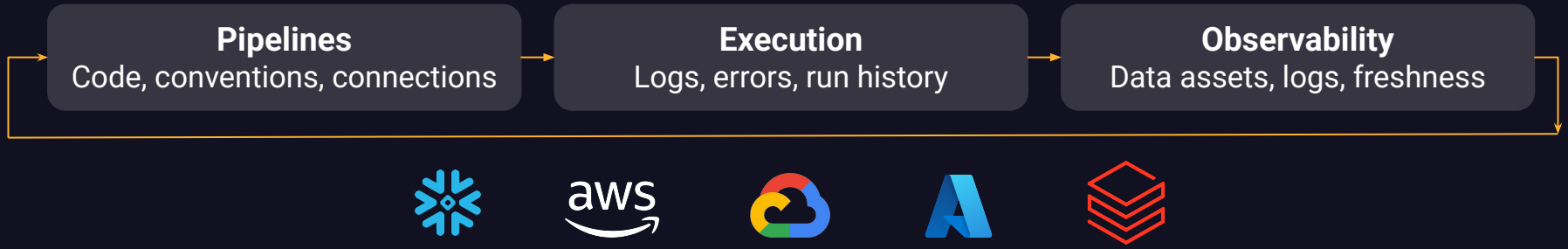
Context is the key to moving from bottleneck to strategic asset

The orchestration layer sits in the execution path of every system in your stack.



Context is the key to moving from bottleneck to strategic asset

The orchestration layer sits in the execution path of every system in your stack.



…but improving the agent's context is based on assumptions unless it's based on real production failures.

Finding ground truth: Support Tickets

Link	Subject	Relevant (fro…	RCA Accuracy
https://astronomer.zendesk.com/agent/tickets/8	Astronomer Runtime Upgrade :3.1-5 (Airflow3) Failing	yes	Correct
https://astronomer.zendesk.com/agent/tickets/8	DAG failed when starting tasks	yes	Incorrect
https://astronomer.zendesk.com/agent/tickets/8	Getting API_SERVER_ERROR while using TriggerDagRunOperator	yes	
https://astronomer.zendesk.com/agent/tickets/8	Recurring SIGKILL errors in PROD DAGs	yes	Incorrect
https://astronomer.zendesk.com/agent/tickets/8	We have customers reporting external wait tasks failing without paging them this mor	yes	Incorrect
https://astronomer.zendesk.com/agent/tickets/8	DBT DAGS are failing after upgrading astro runtime version from 9.21.0 to 13.1.0	yes	Incorrect
https://astronomer.zendesk.com/agent/tickets/8	apache-airflow-providers-microsoft-azure present in test-know workspace still broken	yes	
https://astronomer.zendesk.com/agent/tickets/8	Experiencing upstream errors frequently when trying to look at DAGs in AIrflow	yes	
https://astronomer.zendesk.com/agent/tickets/8	The triggerer does not appear to be running	yes	Partially C…
https://astronomer.zendesk.com/agent/tickets/8	Dag Run's - Weirdly over the Last few Days [Cannot replicate behavior locally]	yes	
https://astronomer.zendesk.com/agent/tickets/8	random airflow restarts	yes	
https://astronomer.zendesk.com/agent/tickets/8	Task failures due to pod not found	yes	
https://astronomer.zendesk.com/agent/tickets/8	DAG Failures on Dec 9th – Root Cause Validation Needed	yes	Correct
https://astronomer.zendesk.com/agent/tickets/8	RE: [EXTERNAL] [Astronomer Support]: Airflow Failure Alert [Critical]: core_dbt_cdh_	yes	

Finding ground truth: SMEs

> I think the key insight we could give the customer for failed DAGs is "what changed". If we can look at upstream DAGs that have changed since the last successful run of this DAG, and then use an LLM on the list of such DAGs and the nature of those changes to surface those that have the highest likelihood of being the cause of this failure. Figuring this out today is a huge pain, and often involves @ everyone has anyone changed anything that could cause this
>
> - We can see the bundle versions (on Airflow 3) and timing of deploys on Astro to determine which DAGs upstream have had changes.
> - Would be extremely valuable to be able to determine more about what changed, especially being able to tell if a particular DAG changed (inclusive of the logic it imports) or just some other DAG on the same deployment
> - If we can see if particular tasks changed, that would be huge.

A suggestion for making the investigation process resemble human investigation.

SHIPPED

Compare to the last success

One run works, one doesn't. Start from the delta.

SHIPPED

Config changes since then

What the environment did, not just the code.

SHIPPED

Dag bundle diffs

Which upstream Dag changed — including its imports.

Finding ground truth: Design Partners

> “The agent had the logs it needed, but it told us to look upstream — when we would rather it looked itself. Narrowing the cause down would save one to two hours per failure.”
>
> - Design Parter, Feb 2026

For more on lineage metadata in Airflow:
Airflow Already Knows the Root Cause - Kacper Muda
Wednesday, 11:30am | Texas Ballroom 7

WHAT WE ADDED

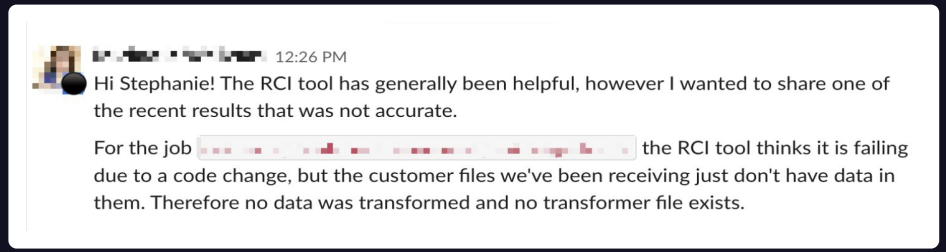
Lineage, prefetched

Three levels upstream and downstream, in the prompt before the agent starts reasoning.

It walks the asset chain itself instead of handing the walk back to the engineer.

SHIPPED · MARCH

Finding ground truth: Design Partners



The agent could see that a deploy had happened, but not the contents of the deploy.

WHAT WE ADDED

Image-layer diffs

We pull the failing deploy and the one before it, and diff the filesystem: requirements, packages, Dockerfile — everything that is not in the Dag bundle.

The agent stops inferring from “a deploy happened” and reads what actually changed.

SHIPPED · MAY

Key Context Pieces



Janus Henderson
INVESTORS

- 130 production Dags
- ~10 failures a day
- 4-5 person data platform team

“Otto acts as our autonomous first responder: it triages every failure, identifies the root cause, and routes it to the right handler before anyone wakes up.

What used to require an hour of manual investigation now happens in minutes, with no human in the loop. The team starts the day reviewing proposed fixes, not investigating failures from scratch."

Kieran Rajasansir, Quantitative Developer
Janus Henderson Investors

Janus Henderson
INVESTORS

- 130 production Dags
- ~10 failures a day
- 4-5 person data platform team

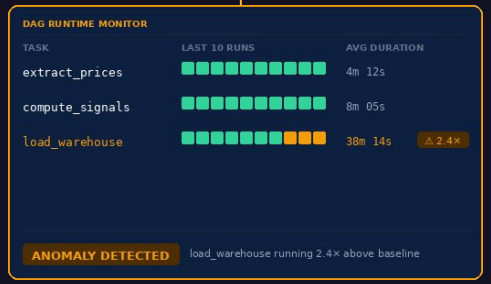
What's Coming: Self-Healing Pipelines

BEFORE FAILURE

AT FAILURE

AFTER DIAGNOSIS

Self-healing pipelines







Long running Dag detection

Otto learns each pipeline's normal runtime and flags a Dag running abnormally long, before it breaches an SLA.

Investigation from alerts

Turn investigation on for any Dag failure alert and Otto's root cause arrives in Slack, PagerDuty or email alongside it.

Otto resolves it

Transient failures Otto reruns for you. Where the fix is in the code, it writes the change and raises the pull request for your team to review.

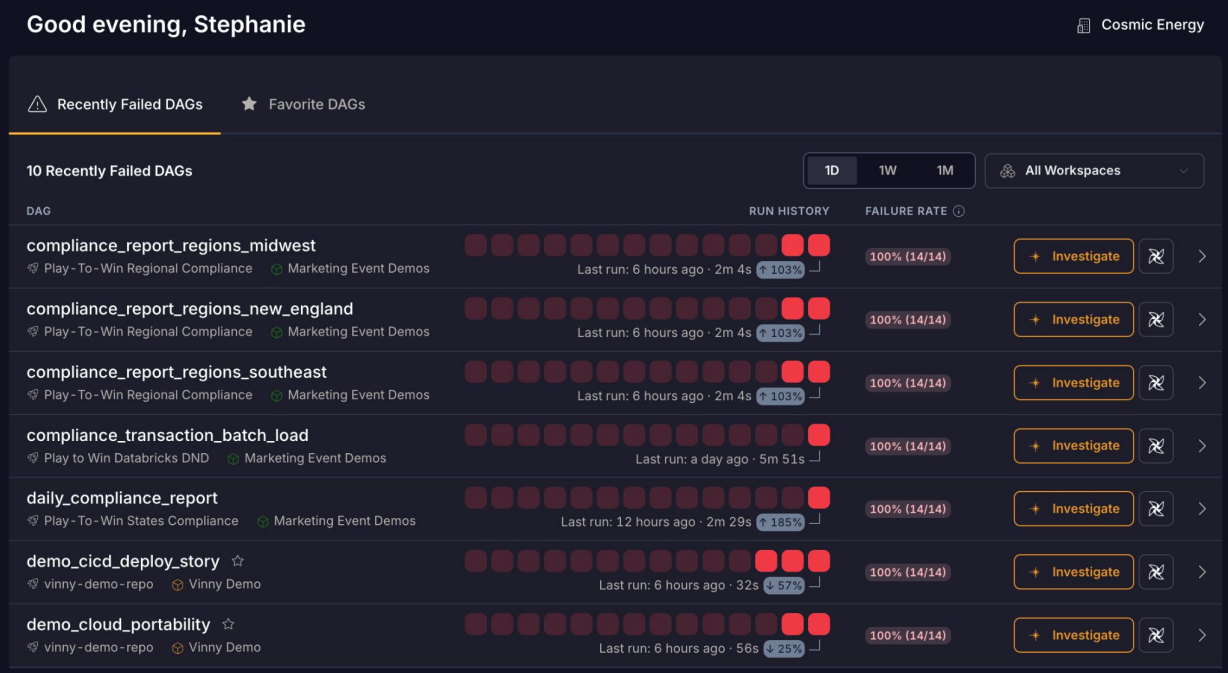
Thank you!

Try it out in Astro or sign up for a free Astro trial to get started.

https://astronomer.link/summit

✉️ christine.shen@astronomer.io

✉️ stephanie.niu@astronomer.io



The 2026 Apache Airflow® Survey is Live!

- Takes just 7 minutes
- Get a free AI Orchestration Fundamentals Certification Exam or DAG Authoring Certification (normally $150 each)



ASTRONOMER

Thank you!