

Activating operational metadata with Airflow, Atlan and OpenLineage

Kacper Muda

GetIn Data: Who are we?

Experts in **Data, Cloud, Analytics and ML/AI, and GenAI solutions**

Experience in: **media, e-commerce, retail, fintech, banking & telco**

Solution Areas



MLOps & Data Platforms



Data & ML engineering projects



Stream processing & real-time analytics

Selected technologies



Open Lineage

Selected Customers



Agenda

1. Data Lineage
2. OpenLineage
3. Airflow & OpenLineage
4. Atlan
5. Q&A

Open Lineage

OpenLineage: From Operators to Hooks

Sep-11 (tomorrow)

16:30-16:55

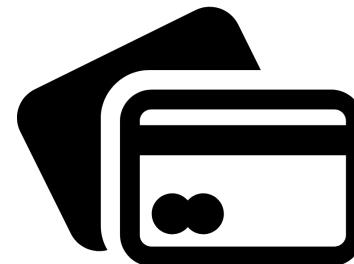
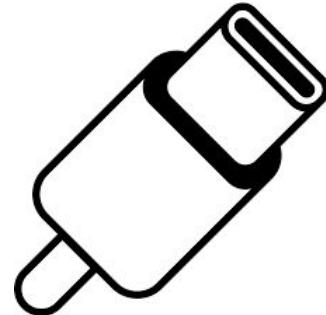
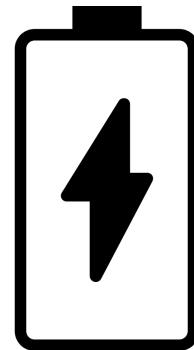
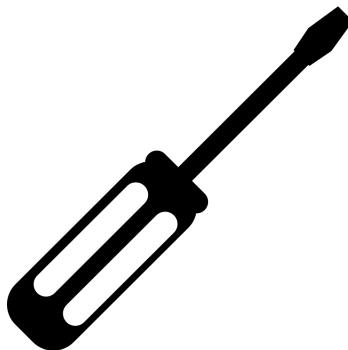
Elizabethan A+B

Speaker(s):



[Maciej Obuchowski](#)

The beauty of standards



Open Lineage

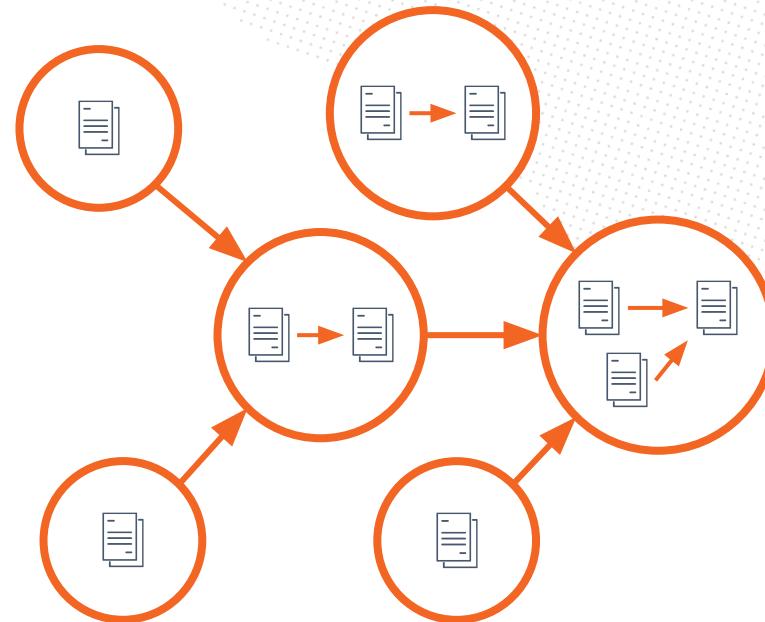
What is data lineage?

Open Lineage

Definition

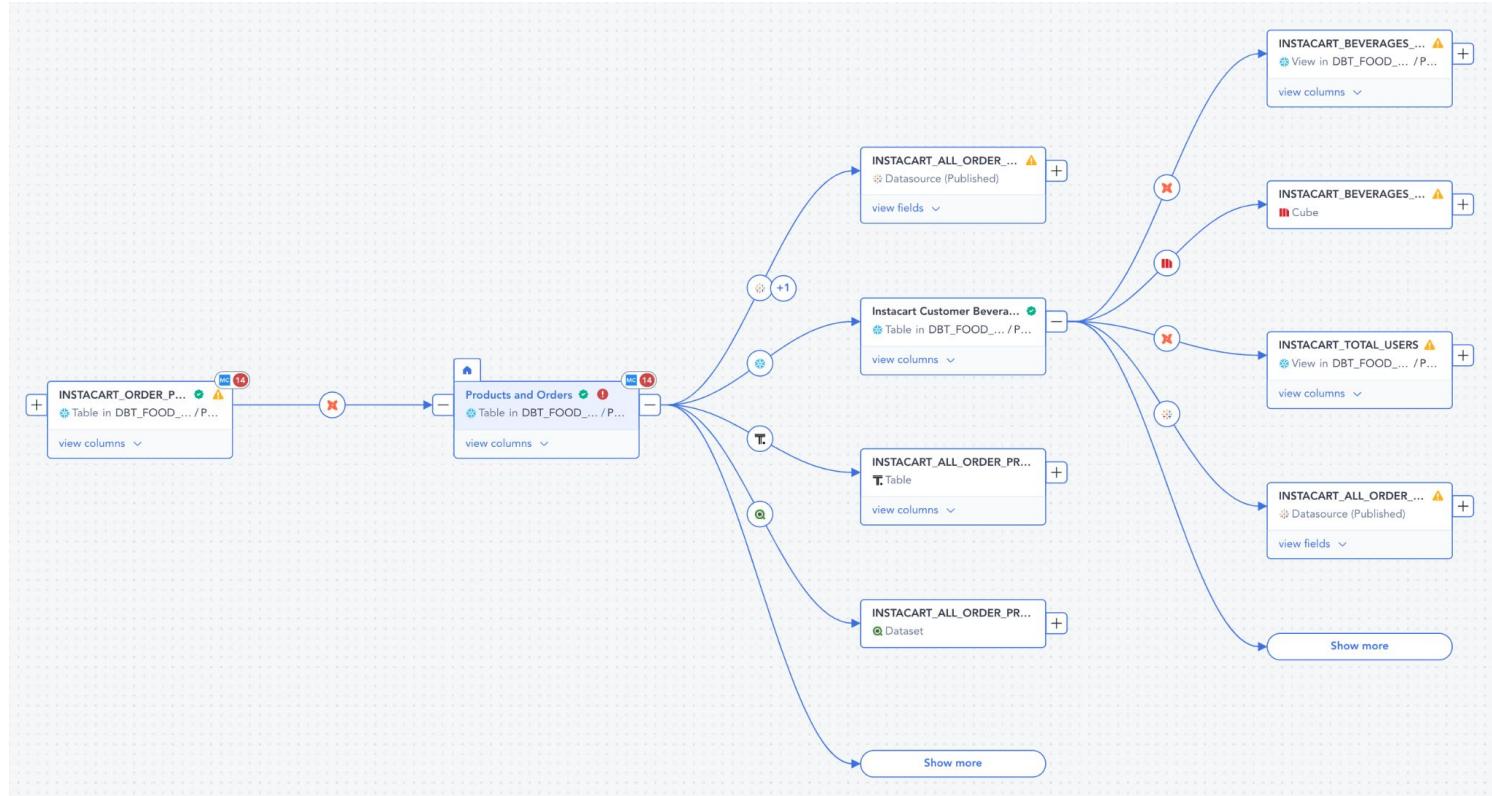
Data lineage is the set of complex relationships between datasets and jobs in data pipelines.

- Producers & consumers of each dataset
- Inputs and outputs of each job



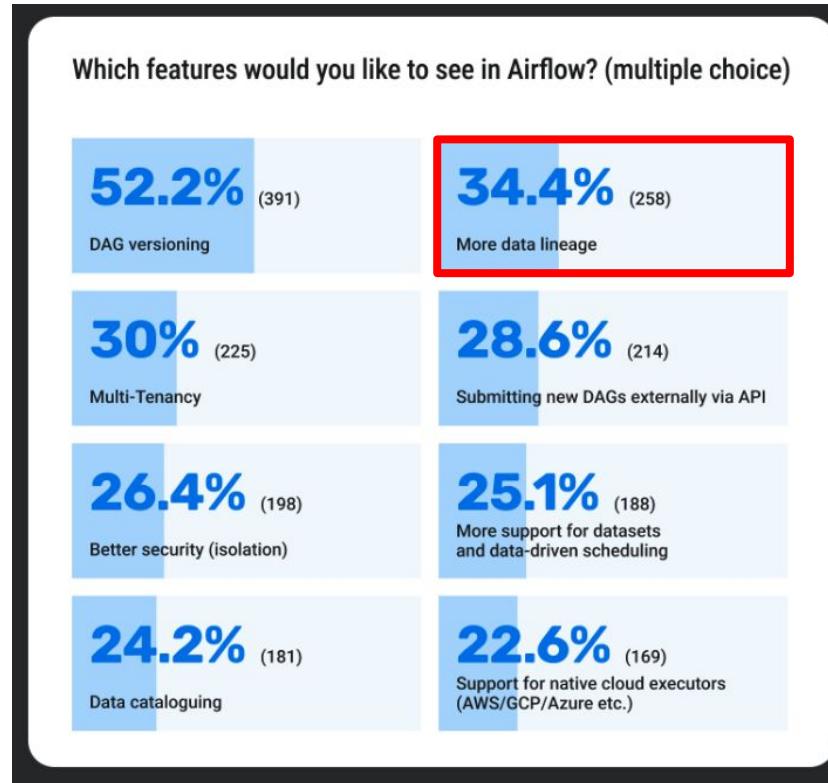
Open Lineage

Example data lineage graph



Open Lineage

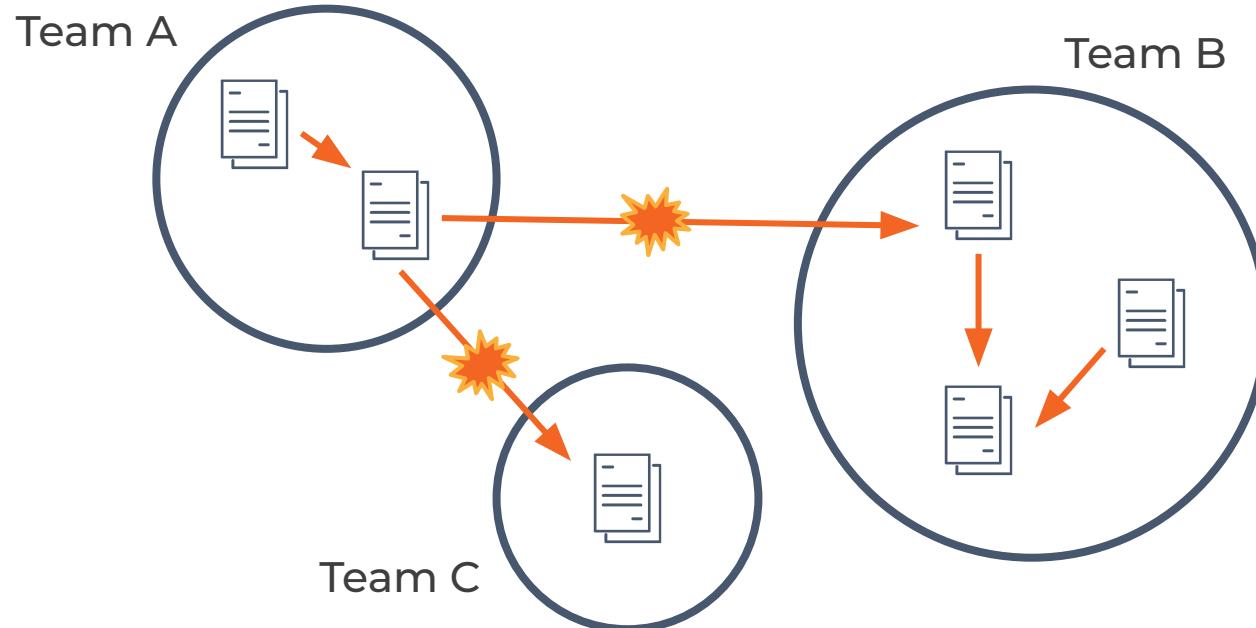
Why care about data lineage?



Open Lineage

source: <https://airflow.apache.org/survey/> (access 10.09.24)

Healthy data ecosystem



OpenLineage

The possibilities are endless

- Verifying compliance and security
- Impact analysis
- Root cause identification
- Smart backfills
- Optimizing data operations
- Dependency tracing
- Issue prioritization
- Anomaly detection
- Change management
- Historical analysis
- ...

Open Lineage

Finding the root cause of our data issue

Open Lineage

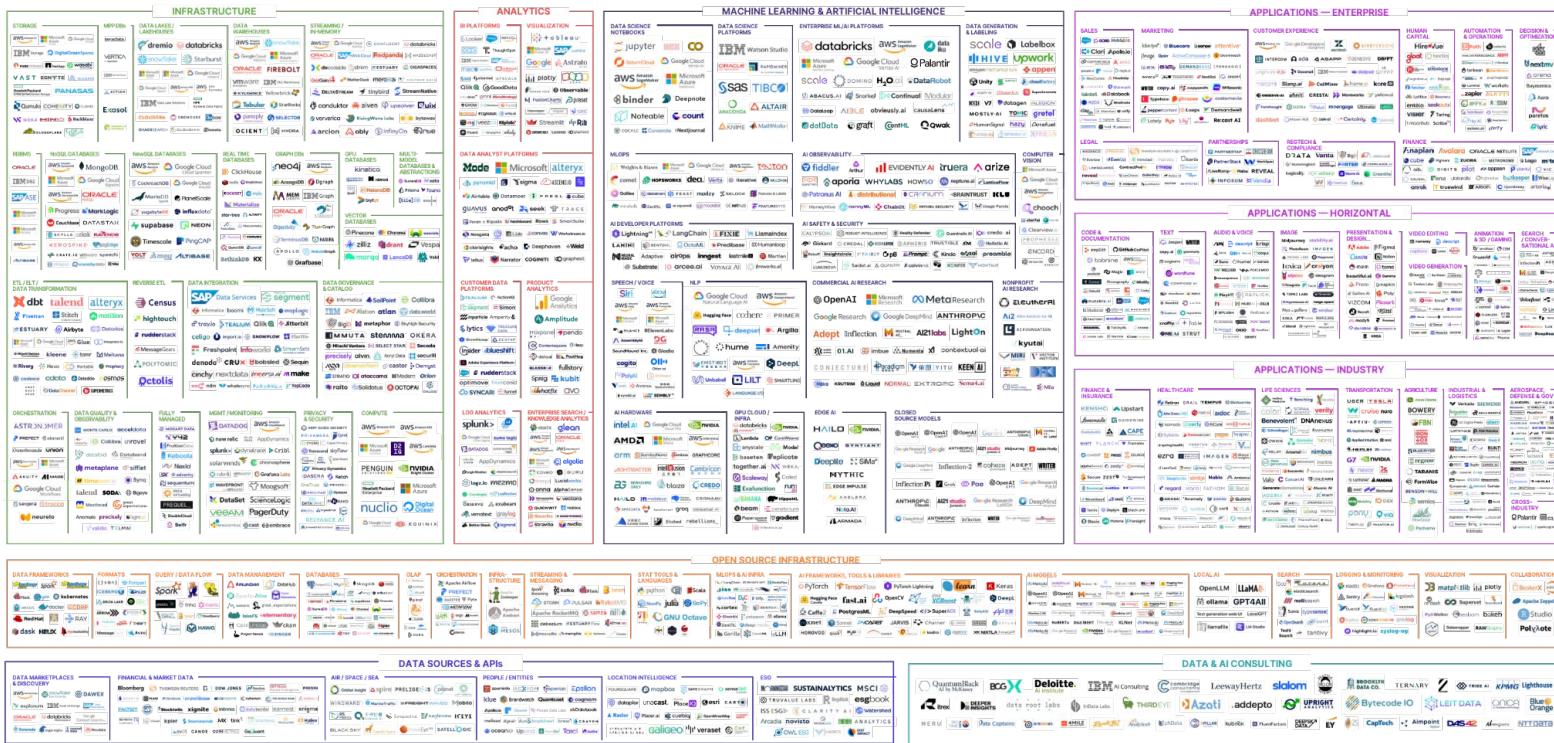
That's it ! Right?

Open Lineage

Working with data in 2024



THE 2024 MAD (MACHINE LEARNING, ARTIFICIAL INTELLIGENCE & DATA) LANDSCAPE



Version 1.0 - March 2024

© Matt Turck (@mattturck), Aman Kaheer (@AmanKaheer11), & FirstMark (@firstmarkcap).

dog-nest-mattturck.com/MAD2034 Interactive version: MAD-firstmarkcap.com

Comments? Email MAD3034@firstmarksap.com

Open Lineage

source: <https://mattturck.com/mad2024/> (access 10.09.24)

© Copyright. All rights reserved. Not to be reproduced without prior written consent.

How to collect lineage?

- Observe the pipeline
- Process logs
- Analyze source code
- Crawl data sources
- ...

Which is best? Combination.

Open Lineage

What is OpenLineage?

Open Lineage

OpenLineage's mission

To define an **open standard** for the automatic collection of lineage metadata from pipelines
as they are running.



OpenLineage

Why runtime?

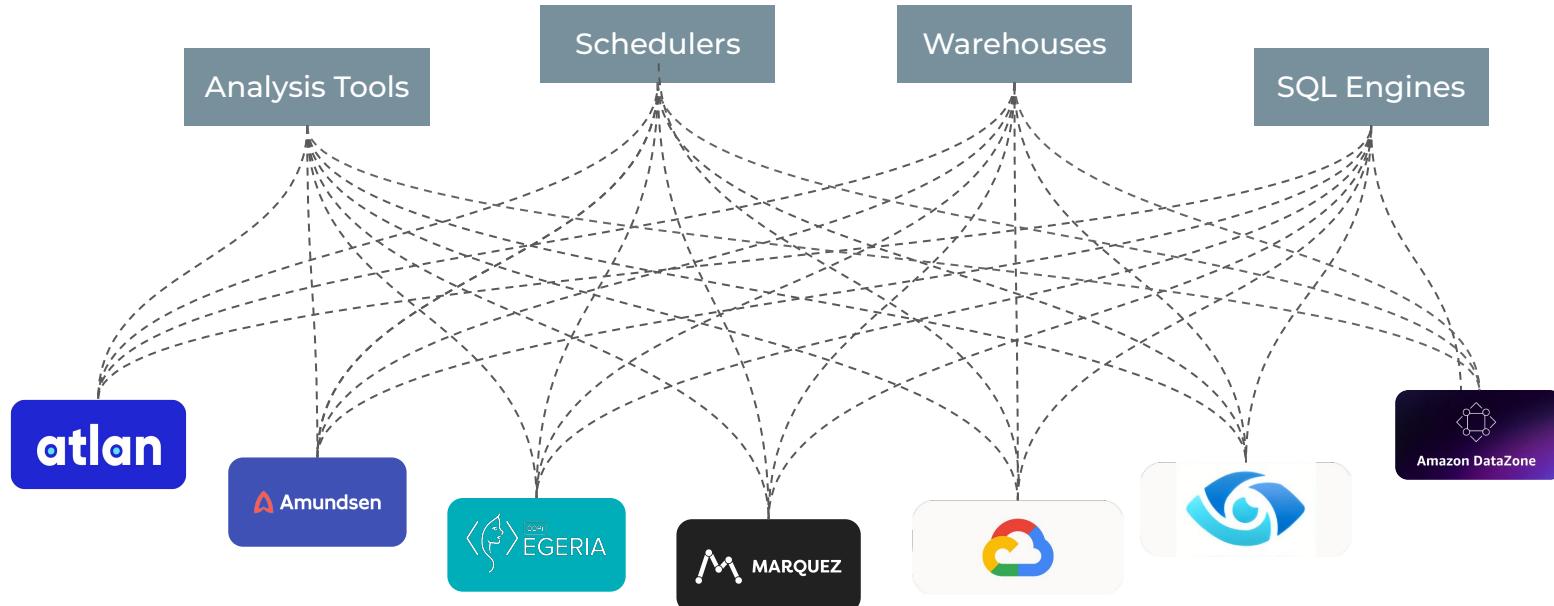


You can try to infer the date and location of an image after the fact...



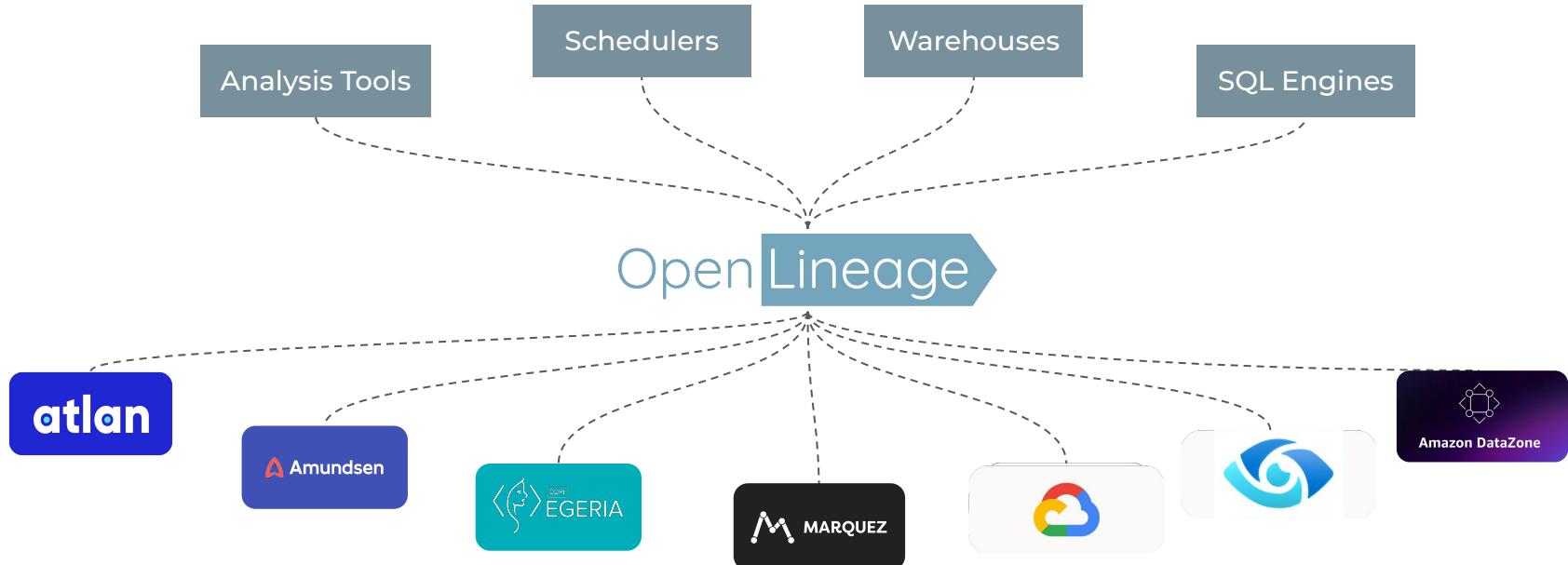
...or you can capture it when the image is originally created!

The Data World without OpenLineage



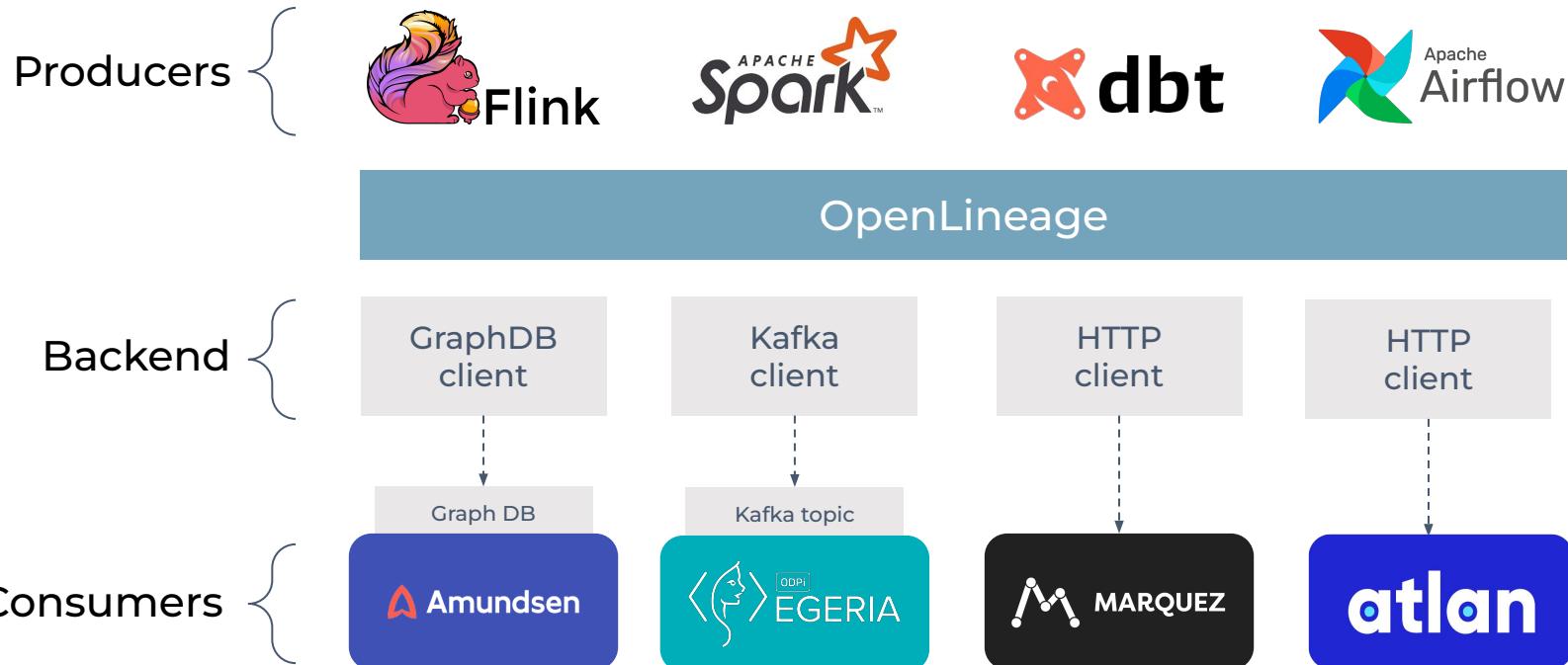
OpenLineage

The Data World with OpenLineage



Open Lineage

Where OpenLineage Fits



OpenLineage

OpenLineage Integrations

Metadata producers



Google Cloud

dagster

Open Lineage



trino



Flink



Keboola

Metadata consumers



ASTRONOMER



atlan



metaphor



Grai



Collibra



MARQUEZ

Google Cloud



DataHub

OpenLineage Contributors



ASTRONOMER

 **getindata**
Part of Xebia



DECATHLON

 Microsoft



 Amundsen



 snowflake

 EGERIA



 Apache
Airflow

 MARQUEZ



 asana

OpenLi  NATURAL INTELLIGENCE

Bloomberg®

Booking.com

OpenLineage design

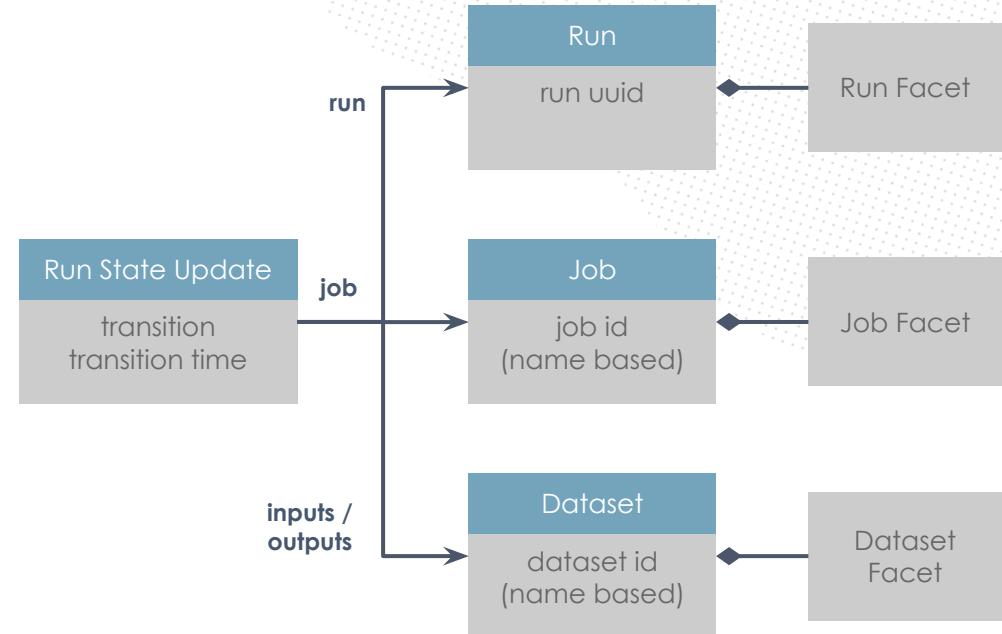
Open Lineage

Data Model

Built around core entities:

- Datasets
- Jobs
- Runs

Defined as a JSONSchema spec.



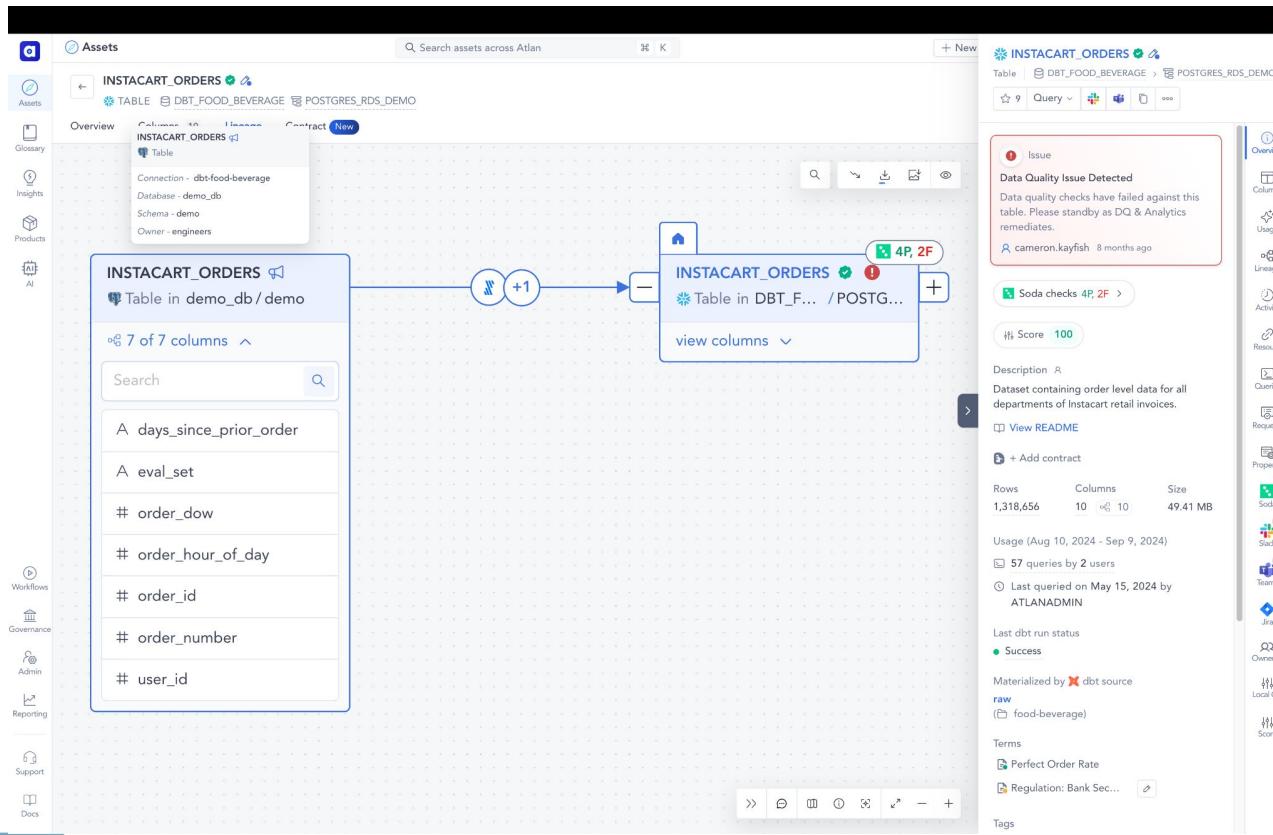
Open Lineage

Highly extensible

```
1  {
2      "eventTime": "2024-09-10T11:00:00.764517+00:00",
3      "eventType": "COMPLETE",
4      "job": {
5          "facets": {
6              "jobType": {"integration": "AIRFLOW", "jobType": "TASK", "processingType": "BATCH"}
7          },
8          "name": "report.extract2",
9          "namespace": "test-airflow"
10     },
11     "run": {
12         "facets": {"externalQuery": {"externalQueryId": "1234", "source": "postgres"}},
13         "runId": "0191c13b-a4c8-75cf-a774-37e85f91db02"
14     },
15     "inputs": [
16         {
17             "facets": {
18                 "schema": {"fields": [{"name": "uid", "type": "STRING"}, {"name": "purchases", "type": "INT"}]}
19             },
20             "name": "export.sales.weekly",
21             "namespace": "postgres://192.158.1.38"
22         }
23     ],
24     "outputs": [
25         {
26             "facets": {"documentation": {"description": "Example documentation facet."}},
27             "name": "customer_data.csv",
28             "namespace": "gs://another_bucket"
29         }
30     ],
31     "producer": "https://github.com/apache/airflow/tree/providers-openlineage/1.11.0",
32     "schemaURL": "https://openlineage.io/spec/2-0-2/OpenLineage.json#/defs/RunEvent"
33 }
```

OpenLineage

From OpenLineage to Atlan



OpenLineage

Core Facet Examples

Dataset:

- Column Lineage
- Quality Assertions
- Ownership
- Schema
- Documentation
- Version
- Output Statistics

Job:

- Documentation
- Job Type
- Ownership
- SQL Job Facet
- Source Code

Run:

- Error Message
- External Query
- Nominal Time
- Parent Run
- Processing Engine

Open  Lineage

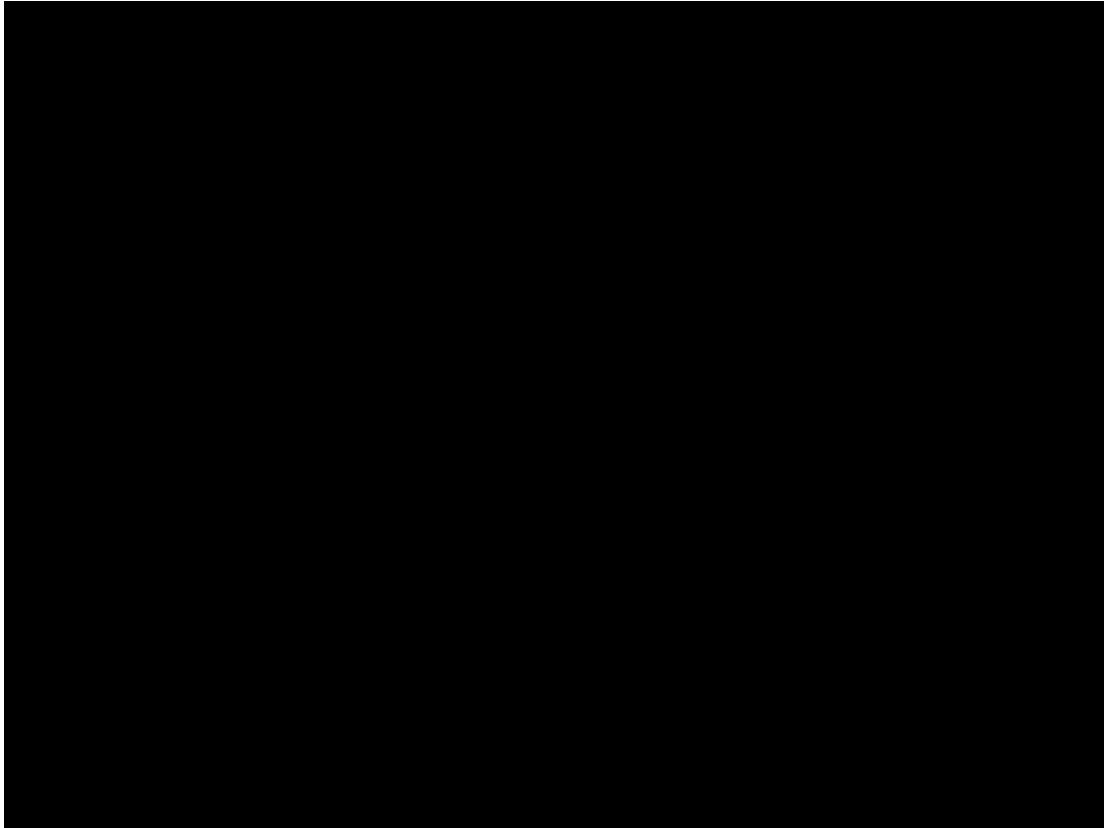
Column Level Lineage

- Emitted by some Airflow jobs and Spark jobs
- Type of transformation is important
- Fits with governance use case

Open Lineage

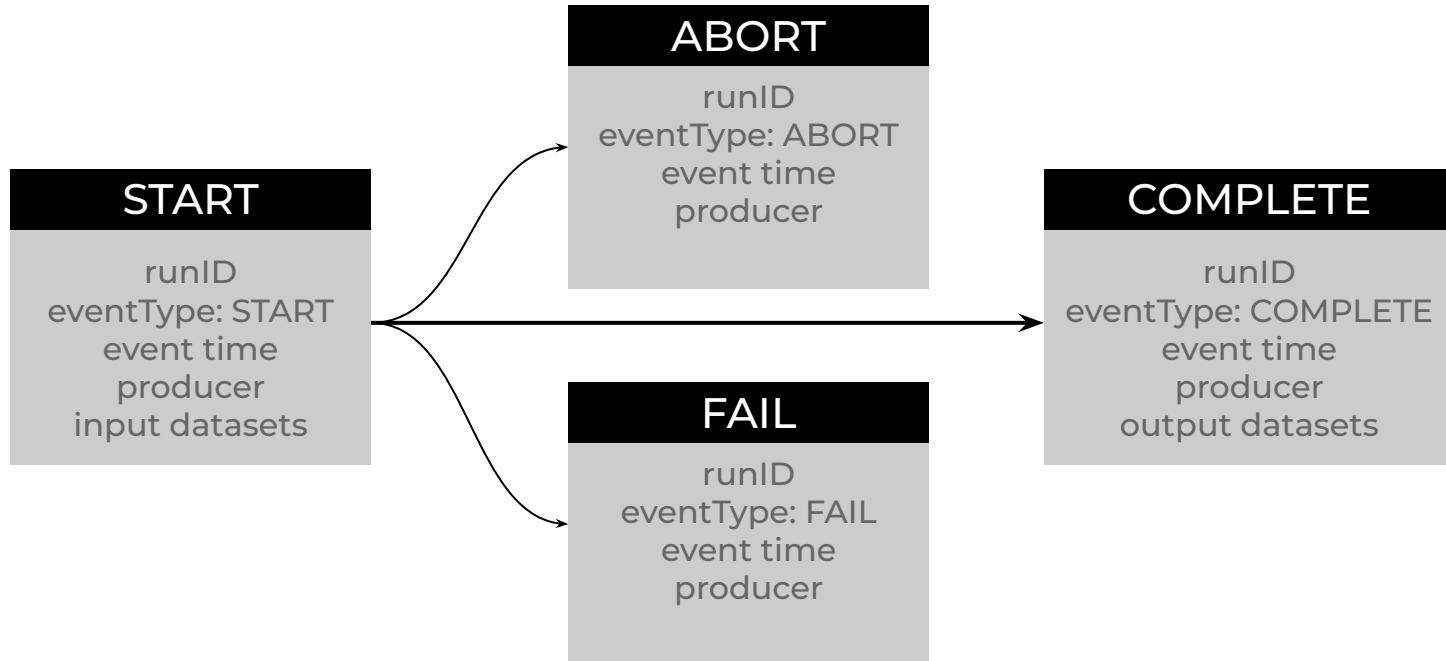
```
1 "outputs": [
2   {
3     "facets": {
4       "columnLineage": {
5         "fields": {
6           "activities": {
7             "inputFields": [
8               {
9                 "field": "clicks",
10                "name": "export.sales.weekly",
11                "namespace": "postgres://192.158.1.38",
12              },
13              {
14                "field": "views",
15                "name": "export.sales.weekly",
16                "namespace": "postgres://192.158.1.38",
17              }
18            ],
19            "transformationDescription": "",
20            "transformationType": "SUM"
21          },
22          "user_id": {
23            "inputFields": [
24              {
25                "field": "uid",
26                "name": "export.sales.weekly",
27                "namespace": "postgres://192.158.1.38",
28              }
29            ],
30            "transformationDescription": "IDENTICAL",
31            "transformationType": "IDENTITY"
32          }
33        },
34      },
35      "name": "my-sales-project.report-dataset.sales-table",
36      "namespace": "bigquery"
37    }
38  ]
39 ]
```

Column level lineage graph



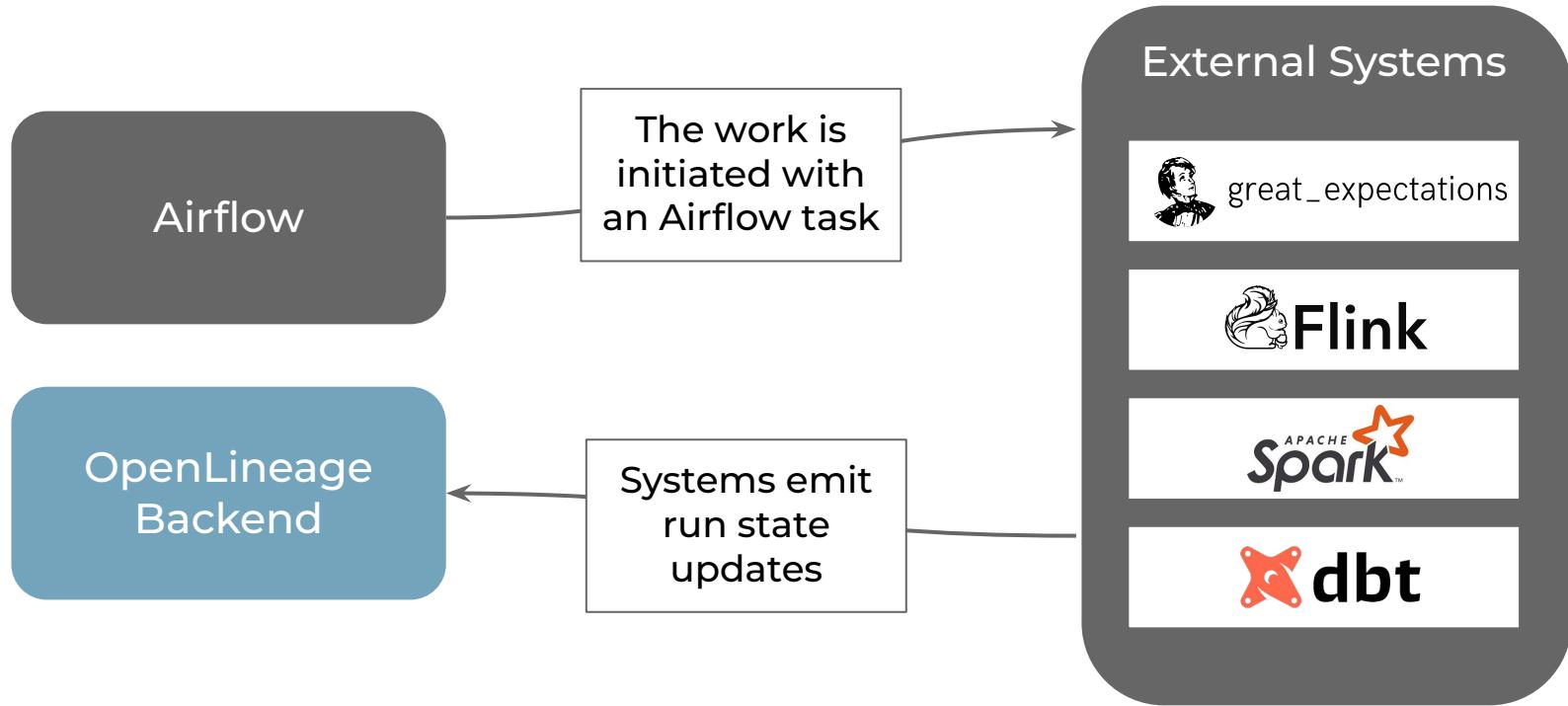
Open Lineage

Lifecycle of a simple job run



Open Lineage

Many underlying systems can be instrumented



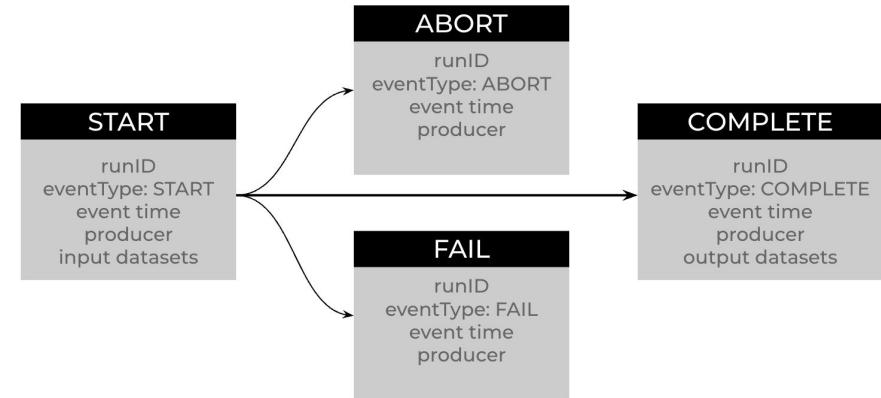
OpenLineage

OpenLineage in Airflow

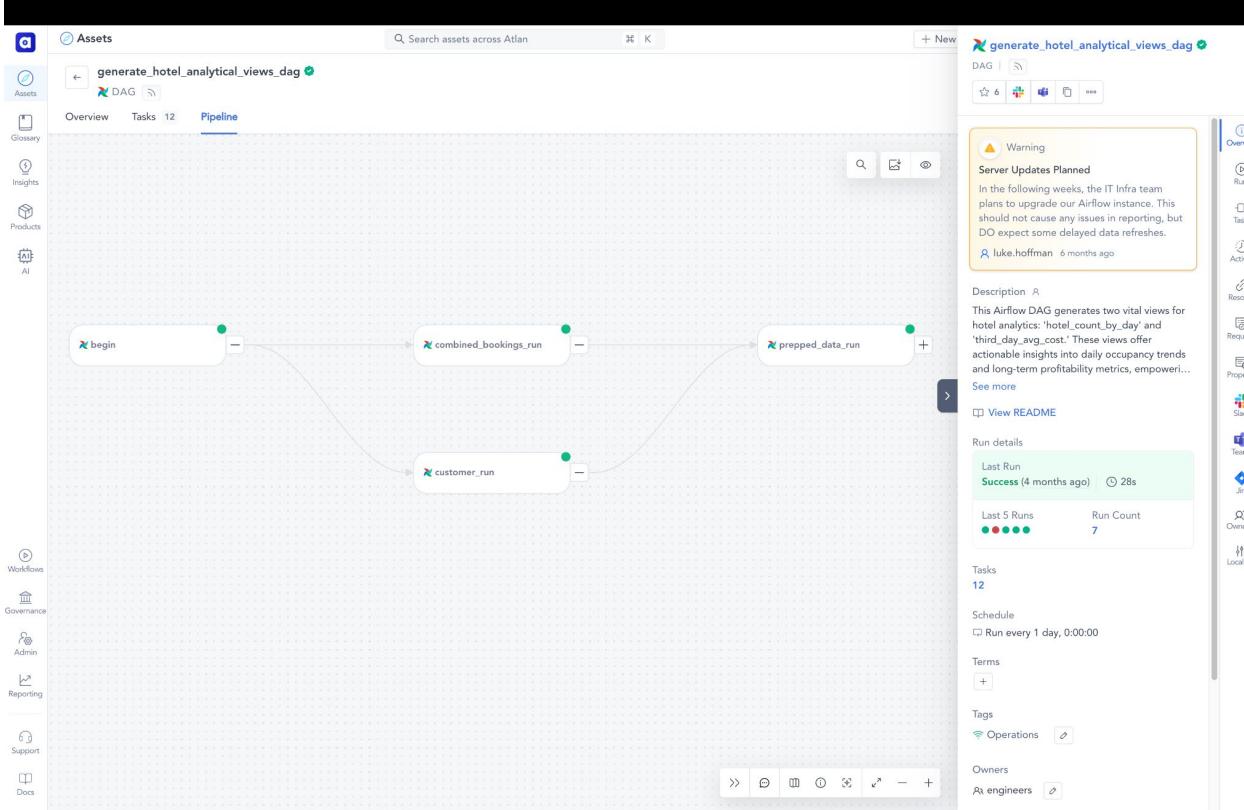
Open Lineage

Airflow 2.7+ OpenLineage Airflow provider

OpenLineage integration in Airflow works by **automatically capturing metadata** from your Airflow tasks and dags during pipeline execution.

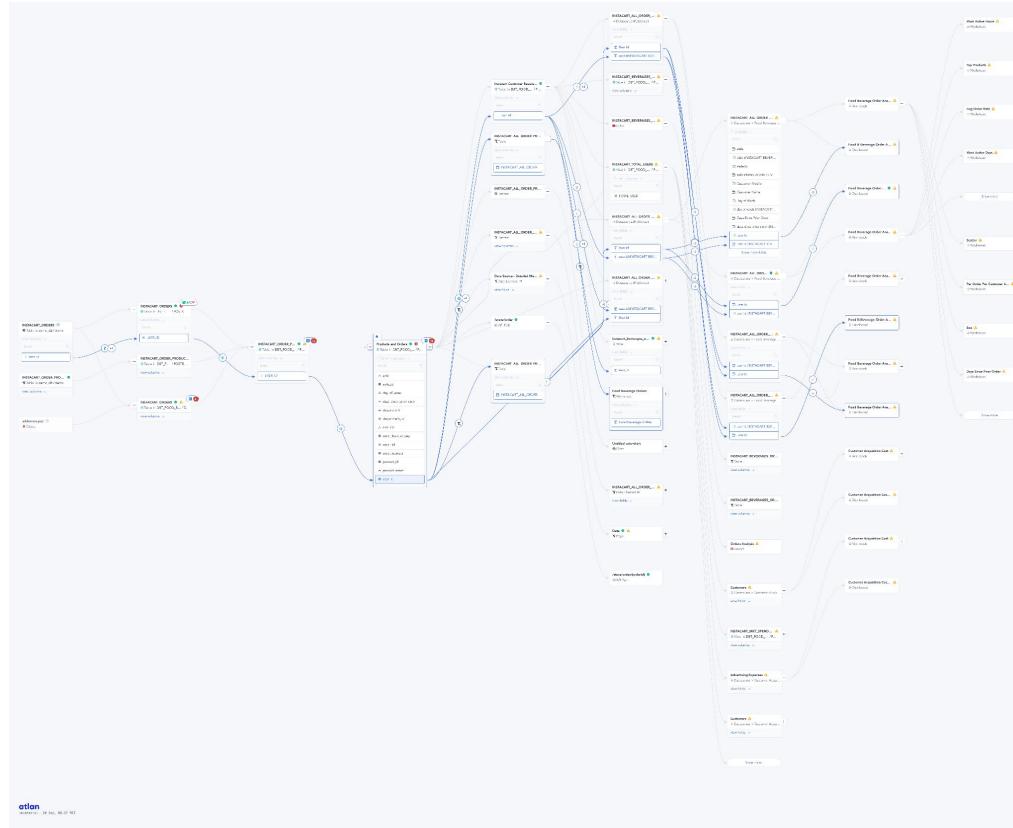


Single source of truth



Open Lineage

Single source of truth



OpenLineage

Currently Supported Operators

See the Provider [docs](#) for an auto-generated list that's always up-to-date!

Not Everything Needs To Have Lineage !

- Messaging ones: SlackAPIOperator, etc.
- “DevOps” ones: GCSCreateBucketOperator, etc.
- Protocol ones: SimpleHttpOperator, etc.
- ...

What about PythonOperator? Hook Level Lineage to the rescue !

Open [Lineage](#) 

OpenLineage methods

```
def get_openlineage_facets_on_start() -> OperatorLineage:  
    ...  
  
def get_openlineage_facets_on_complete(ti: TaskInstance) -> OperatorLineage:  
    ...  
  
def get_openlineage_facets_on_failure(ti: TaskInstance) -> OperatorLineage:  
    ...
```

OpenLineage

Let's use it !

Open Lineage

1. Install provider package: **pip install apache-airflow-providers-openlineage**
2. Provide a Transport configuration so that OpenLineage provider knows where to send the events.
 - a. within airflow.cfg file
 - b. or with environment variable

```
{"type": "http", "url": "http://example.com:5000", "endpoint": "api/v1/lineage"}
```

That's it ! OpenLineage events should be sent to the configured backend when DAGs are run.

Open Lineage

Additional config

- [namespace] - Set namespace that the lineage data belongs to.
- [disabled_for_operators] - Exclude some Operators from emitting events.
- [selective_enable] - Explicitly enable OpenLineage per DAG / Task
- [disabled]
- [extractors]
- [disable_source_code]
- [execution_timeout]
- [include_full_task_info]
- [dag_state_change_process_pool_size]
- [config_path]
- ...

Open Lineage

Inject custom data to events by providing custom facets

```
@attrs.define
class MyCustomRunFacet(RunFacet):
    """Define a custom facet."""

    name: str
    jobState: str
    uniqueName: str
    displayName: str
    dagId: str
    taskId: str
    cluster: str
    custom_metadata: dict
```

```
def get_my_custom_facet(
    task_instance: TaskInstance, ti_state: TaskInstanceState
) -> dict[str, RunFacet] | None:
    operator_name = task_instance.task.operator_name
    custom_metadata = {}
    if operator_name == "BashOperator":
        return None
    if ti_state == TaskInstanceState.FAILED:
        custom_metadata["custom_key_failed"] = "custom_value"
    job_unique_name = f"TEST.{task_instance.dag_id}.{task_instance.task_id}"
    return {
        "additional_run_facet": MyCustomRunFacet(
            name="test-lineage-namespace",
            jobState=task_instance.state,
            uniqueName=job_unique_name,
            displayName=f"{task_instance.dag_id}.{task_instance.task_id}",
            dagId=task_instance.dag_id,
            taskId=task_instance.task_id,
            cluster="TEST",
            custom_metadata=custom_metadata,
        )
    }
```

Composite Transport: send lineage to multiple backends

```
1  {
2      "type": "composite",
3      "transports": [
4          {
5              "type": "kafka",
6              "config": {
7                  "bootstrap.servers": "localhost:9092"
8              },
9              "topic": "random-topic",
10             "messageKey": "key",
11             "flush": false
12         },
13         {
14             "type": "http",
15             "url": "http://example.com:5000",
16             "endpoint": "api/v1/lineage"
17         }
18     ]
19 }
```

Open Lineage

Quickly start consuming OpenLineage events

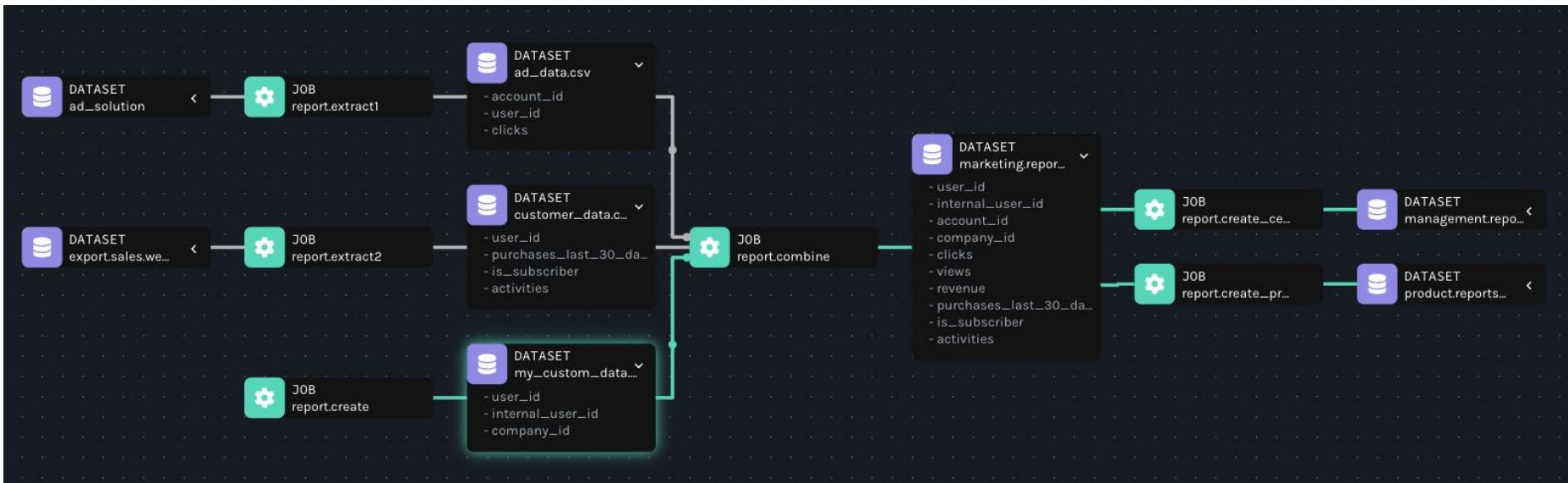
If using Marquez (OpenSource, LF AI & Data Foundation project):

1. Clone the project (github.com/marquezproject/marquez)
2. run `./docker/up.sh`
3. Configure OpenLineage http transport to point to Marquez.
4. View lineage in a browser at <http://localhost:3000> !



Marquez: the Complete OpenLineage solution
Metadata repository, lineage API and GUI
<https://marquezproject.ai>

Example data lineage graph

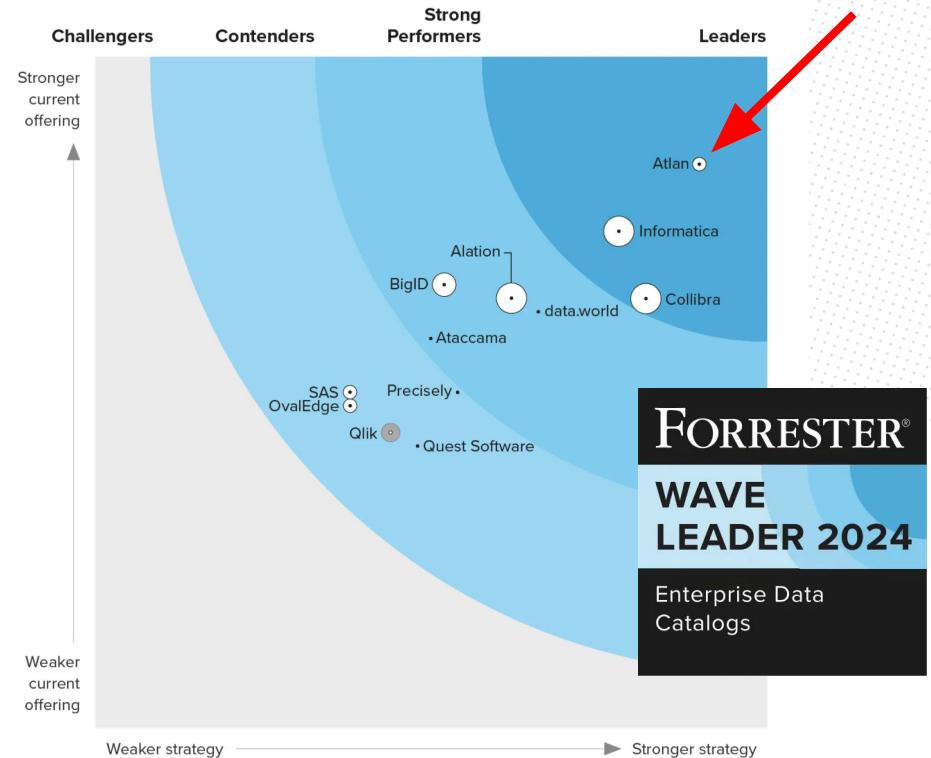


source: Marquez (<https://marquezproject.ai/>)

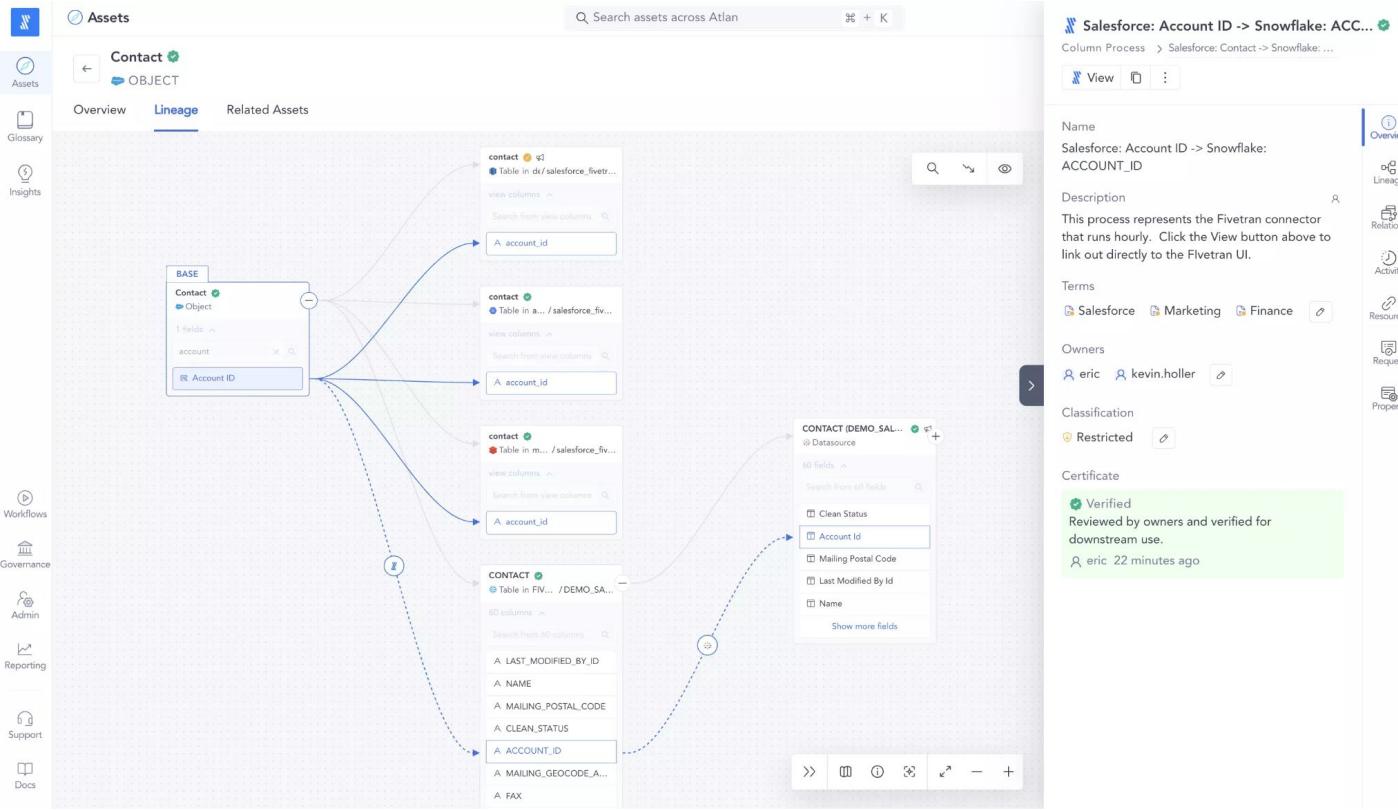
Open Lineage 

Atlan Data Catalog is an active metadata platform unifying cataloging, data discovery, lineage, and governance experience.

Open Lineage



Lineage map

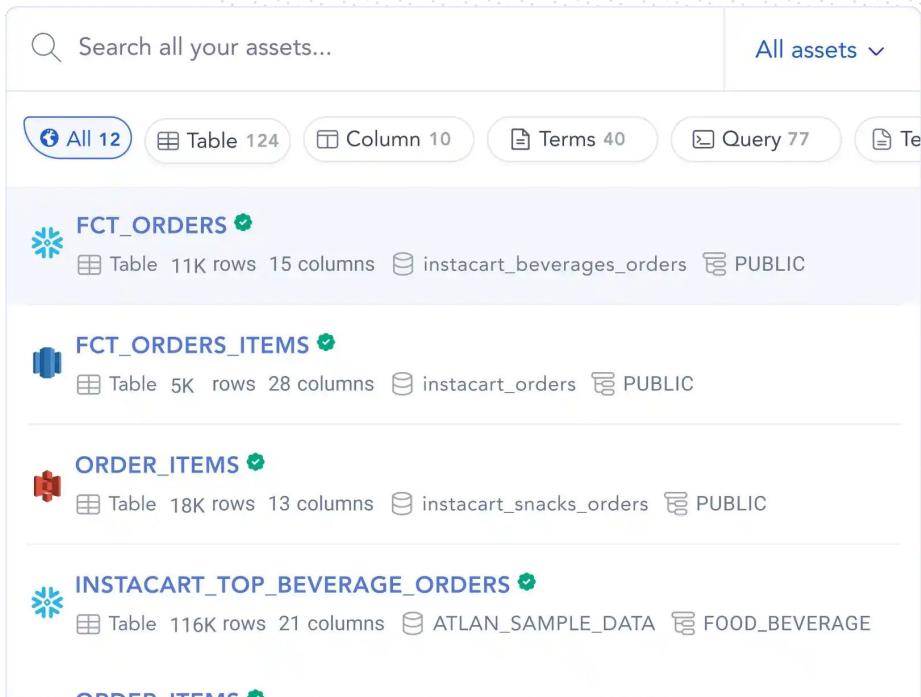


Open Lineage

Google for your data

Search across all your data
from a single place using
natural language or business
context.

Open Lineage



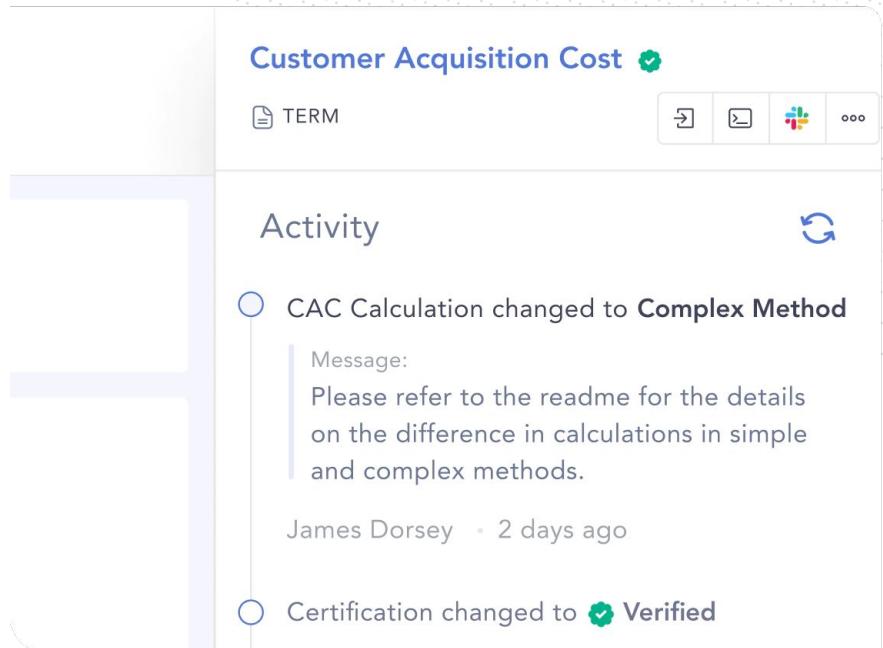
The screenshot shows a search interface for 'All assets' with 12 results. The results are:

- FCT_ORDERS** (green checkmark)
Table 11K rows 15 columns instacart_beverages_orders PUBLIC
- FCT_ORDERS_ITEMS** (green checkmark)
Table 5K rows 28 columns instacart_orders PUBLIC
- ORDER_ITEMS** (red checkmark)
Table 18K rows 13 columns instacart_snacks_orders PUBLIC
- INSTACART_TOP_BEVERAGE_ORDERS** (green checkmark)
Table 116K rows 21 columns ATLAN_SAMPLE_DATA FOOD_BEVERAGE

Track activity

See how a metric formula has changed over time and why.

Open Lineage



The screenshot shows a software interface for tracking metric activity. At the top, there's a header for 'Customer Acquisition Cost' with a green checkmark icon. Below it, a 'TERM' section includes icons for edit, delete, share, and more. The main area is titled 'Activity' with a refresh button. It lists two items: 1) A change from 'Simple Method' to 'Complex Method' with a message explaining the difference in calculations. 2) A change in certification status to 'Verified'. Both items show they were made by James Dorsey 2 days ago.

Customer Acquisition Cost 

TERM    

Activity 

CAC Calculation changed to **Complex Method**
Message:
Please refer to the readme for the details on the difference in calculations in simple and complex methods.
James Dorsey • 2 days ago

Certification changed to  **Verified**

Add owners

Know who's responsible for every asset and who you can reach out to.

Open Lineage



Customer Acquisition Cost 

 TERM    

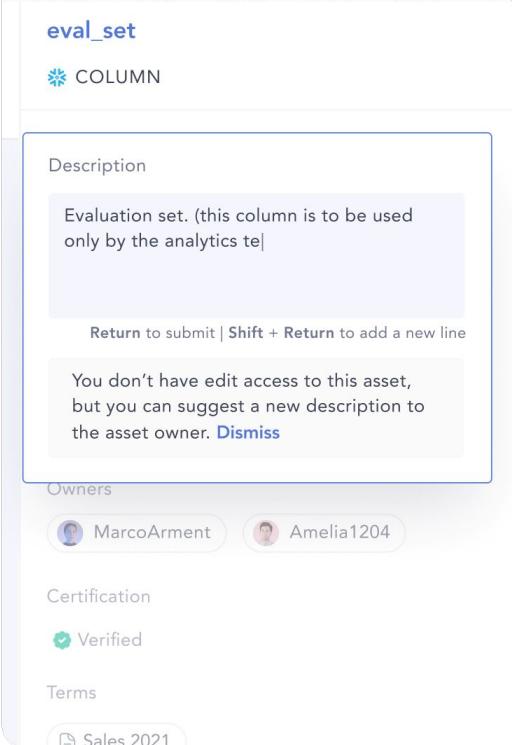
 Add Owners

 Marketing Team  Adam Keane

 Adam  

 Herman	<input type="checkbox"/>
 Adam Keane	<input checked="" type="checkbox"/>
 Jenna	<input type="checkbox"/>

Members suggest, Admins decide



eval_set
COLUMN

Description

Evaluation set. (this column is to be used only by the analytics te|)

Return to submit | Shift + Return to add a new line

You don't have edit access to this asset, but you can suggest a new description to the asset owner. [Dismiss](#)

Owners

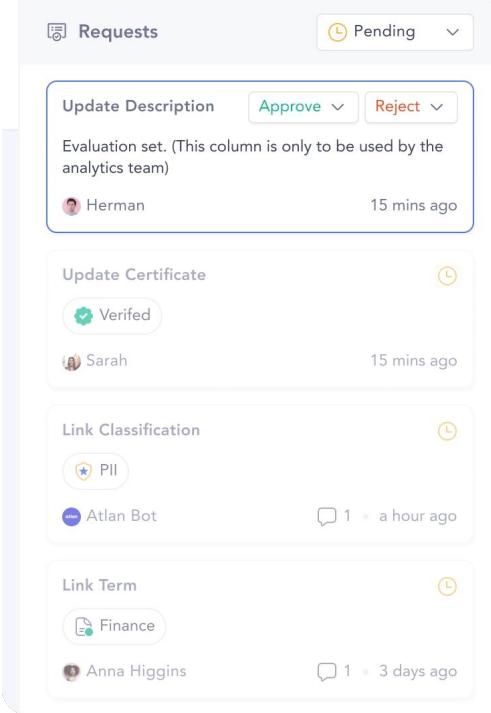
MarcoArment, Amelia1204

Certification

Verified

Terms

Sales 2021



Requests

Pending

Update Description

Evaluation set. (This column is only to be used by the analytics team)

Herman 15 mins ago

Approve Reject

Update Certificate

Verified

Sarah 15 mins ago

Link Classification

PII

Atlan Bot a hour ago

Link Term

Finance

Anna Higgins 3 days ago

OpenLineage

Add announcements

Bubble up issues and warnings in Atlan.

Open Lineage



The screenshot shows the Atlan platform interface for a table named "instacart_top_beverage_orders_by_sales". The table is categorized under "Metastore" and "sales_orders". Below the table name, there are three tabs: "Overview" (which is active), "Lineage", and "Queries 56". A prominent blue announcement bubble is displayed, containing the following text:

New columns will be added to this table on June 1, 2022.
OPPORTUNITY_HAS_BEEN_ORDERED will be “added” to this table to replace IS_OPPOORTUNITY ORDERED.

Adam Keane 02 May 2022

Below the announcement, there is a "Table Summary" section with a table icon and the text "Table Summary".

Alert other users

Open Lineage

Impact analysis customer story

Company: **Dr. Martens**

Industry: **footwear**

Challenge: **drive self-service to their data stack**

Result:

Impact Analysis Effort reduced **from 4-6 weeks to 30 minutes**

Open Lineage 

Usage popularity

Access the most widely used tables, columns, queries.

Open Lineage 

Column Preview		Sample Data	
Search columns		All 168	A varchar 35
#	Column Name	Data Type	Description
1	USED OFTEN (last 30 days)	STRING	---
2	225 queries by 8 users	STRING	---
3		NUMBER	UUID from
4		STRING	---
5	supplier_id	STRING	supplier in
6	sale_datetime_column	DATETIME	Date of sal
7	captured_datetime_column	DATETIME	---

Popularity customer story

Company: **Mistertemp**

Industry: **recruitment**

Challenge: **improve the navigability / usability of their data**

Result:

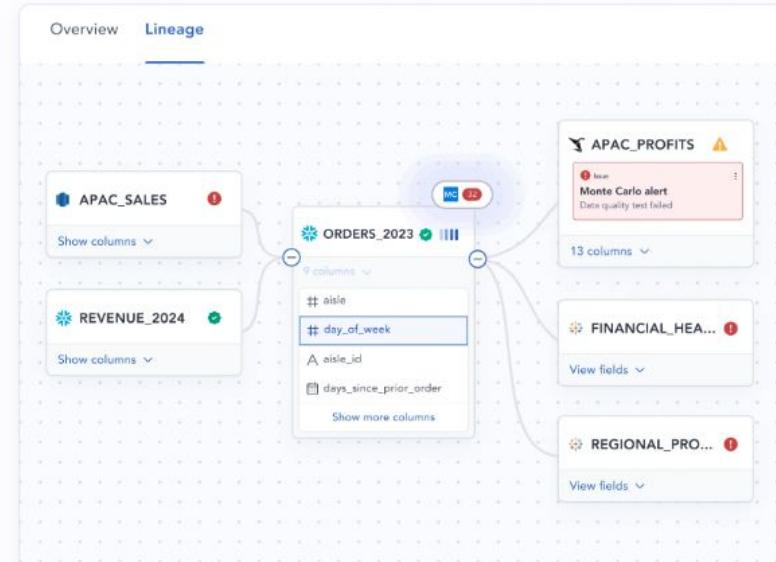
Deprecated **more than half** of their Snowflake tables

Removed **more than 60%** of their Looker assets

Open Lineage 

Column level lineage

Visualize column-level relationships from your data sources to your BI dashboards.



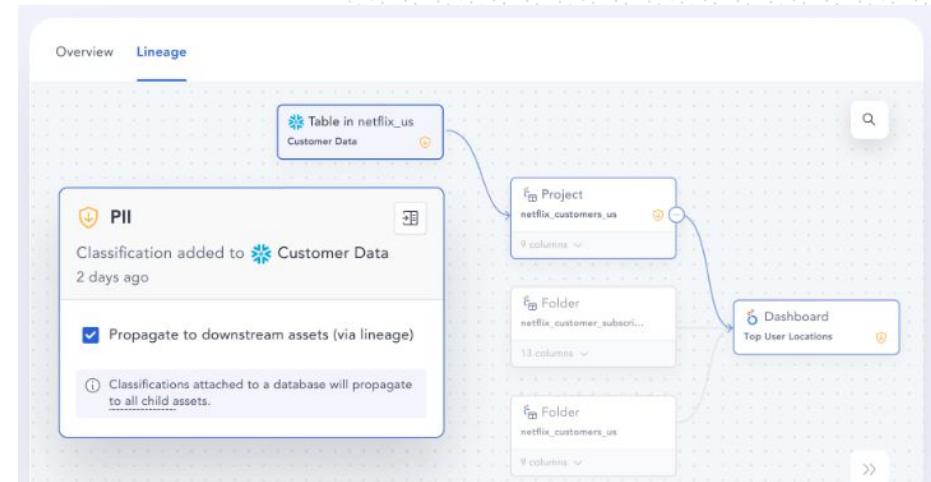
Open Lineage 

Where does column lineage come from?

Open Lineage

Protect sensitive data from
the source

Every asset derived from a PII
source column inherits
restricted policies.



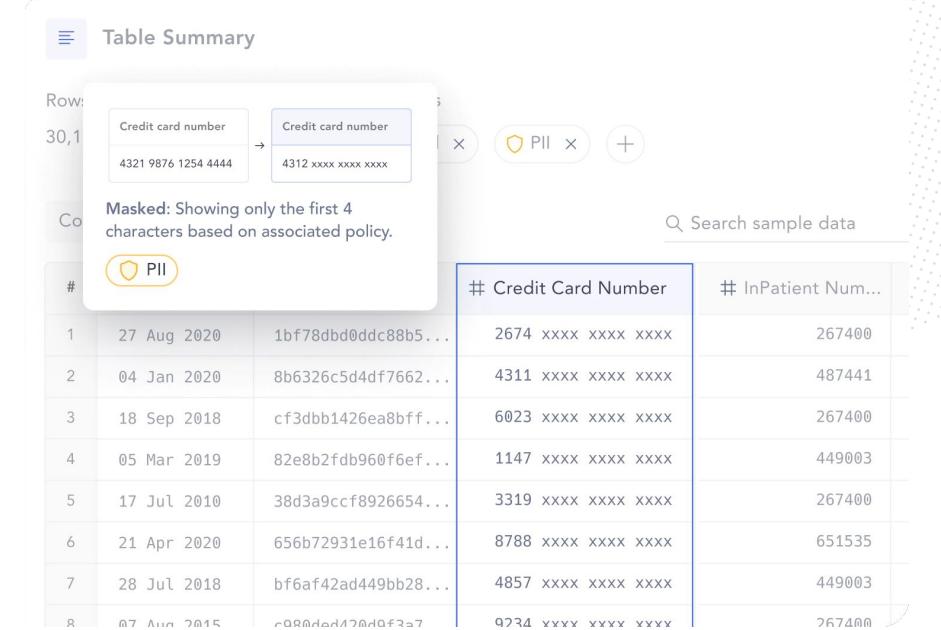
Open Lineage 

Preview data

Get an overview of what the data in your columns actually looks like.

Sensitive data is automatically hashed, redacted, or nullified based on access policies.

Open Lineage

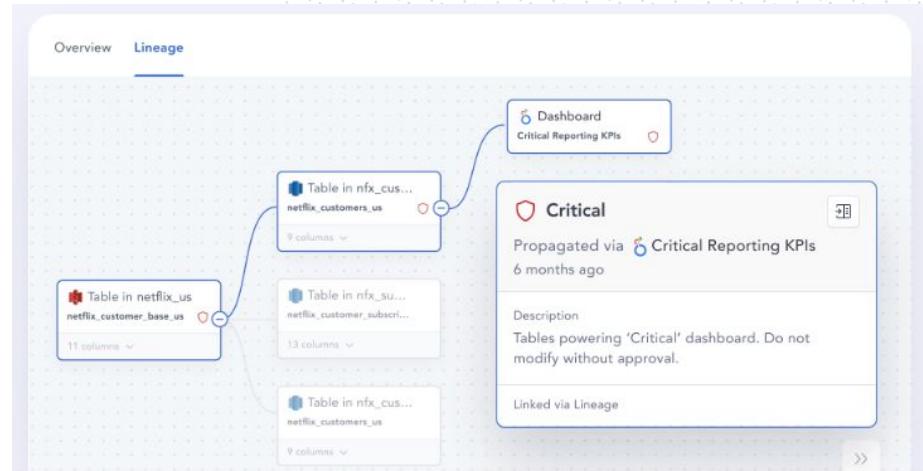


The screenshot shows a data preview interface. At the top, there's a 'Table Summary' section with a row example from 'Row: 30,1'. It shows two columns: 'Credit card number' with the value '4321 9876 1254 4444' and 'Credit card number' with the value '4312 xxxx xxxx xxxx'. A note below says 'Masked: Showing only the first 4 characters based on associated policy.' To the right of this summary is a search bar labeled 'Search sample data'. Below the summary is a table with several rows of data. The first column is labeled '# PII' and contains a blue button with a yellow border. The second column is labeled '# Credit Card Number' and contains masked values like '2674 xxxx xxxx xxxx'. The third column is labeled '# InPatient Num...' and contains values like '267400'. The table has a light gray background with white text and blue headers.

# PII	# Credit Card Number	# InPatient Num...
1	27 Aug 2020	1bf78dbd0ddc88b5...
2	04 Jan 2020	8b6326c5d4df7662...
3	18 Sep 2018	cf3dbb1426ea8bff...
4	05 Mar 2019	82e8b2fdb960f6ef...
5	17 Jul 2010	38d3a9ccf8926654...
6	21 Apr 2020	656b72931e16f41d...
7	28 Jul 2018	bf6af42ad449bb28...
8	07 Aug 2015	c08ad6d120d0f357

Notify users of critical source tables

Propagate a “Critical” tag from your dashboard to upstream source tables.



Open Lineage 

Scale tagging and enrichment with rule-based automations for all metadata in Atlan.

Open Lineage

Playbooks

 Deprecate unused Snowflake assets Daily at 12:00 PM	 3 actions	2 mins ago 
 Remove underutilized dashboards Weekly on Mondays 00:00 AM	 No actions	2 days ago 
 Auto-PII tagging Monthly 01:00 PM	 4 actions	4 days ago 
 Automated ownership model Daily at 01:00 PM	 1 action	5 days ago 

Compliance customer story

Company: **Tide**

Industry: **banking**

Challenge: **improve compliance with GDPR's Right to Erasure**, commonly known as the “Right to be forgotten”.

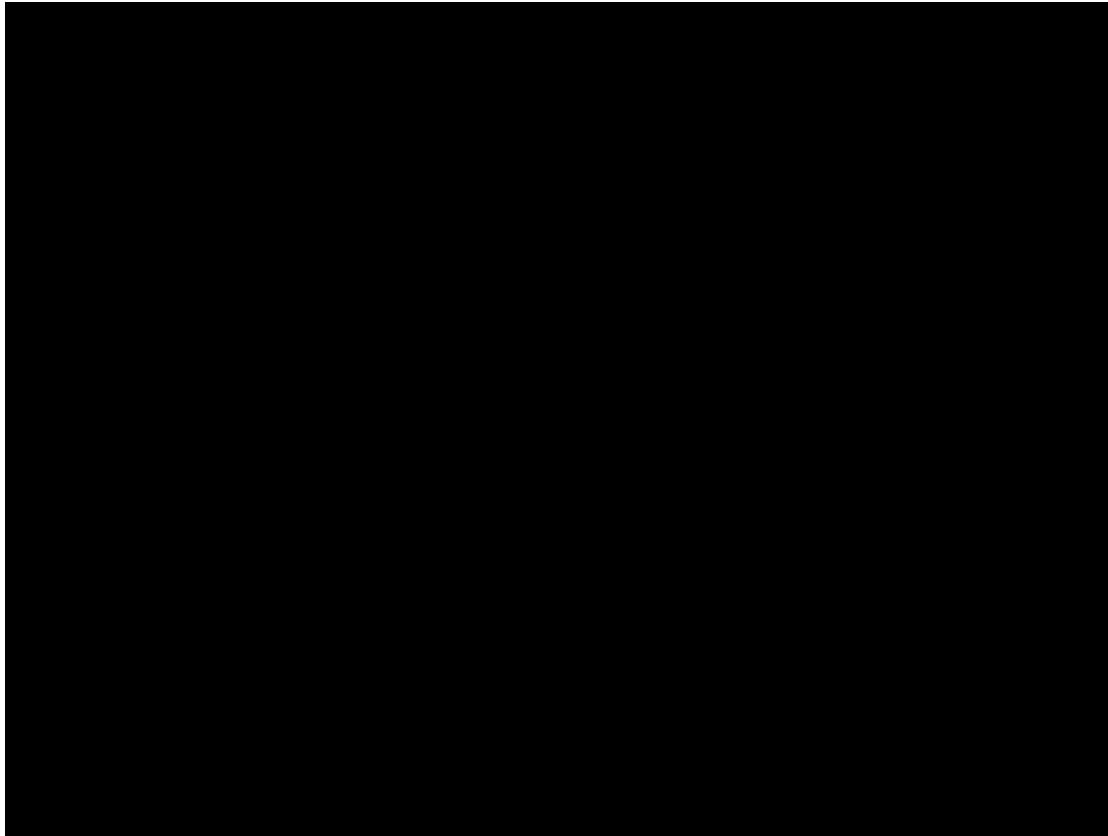
Result:

PII Tagging Effort reduced from 50 days to 5 hours

GDPR Compliance strengthened for 500k customers

Open Lineage 

Single asset details



Open Lineage

Governance at scale customer story

Company: Porto

Industry: insurance and banking

Challenge: improve data literacy across their organization

Result:

Time spent on governance **reduced by 40%**

Automated governance of **over 1 million data assets**

Open Lineage 

Next steps: try it !

1. Run Marquez (free OSS - backend for OpenLineage)
2. Add OpenLineage to your solution (Airflow, Spark)
3. See the lineage graph (and more!) in your web browser.
4. Enjoy lineage benefits !

Open Lineage 

Can I buy a SaaS of OpenLineage?

No. It's open source and free.

You can use it with free backend like self hosted Marquez.

But, if needed, GetInData is here to assist.

Whether you're implementing OpenLineage or integrating it as a data consumer, our team has experience successfully delivering solutions for our clients.

Open Lineage 

Q&A

Thursday 6:00 PM,
September 12th
(the day after tomorrow)

Astronomer offices,
8 California St
(1 mile from here)

Join us for in-depth talks &
discussion over dinner!

Open Lineage

OpenLineage meetup

 getindata
Part of Xebia



<https://www.meetup.com/meetup-group-bnfqymxe/events/302718127>



Open Lineage

Thank you !



@kacpermuda



Muda.Kacper@gmail.com

Extra slides

Open Lineage

OpenLineage vs OpenTelemetry

Blog post:

*How OpenLineage takes
inspiration from
OpenTelemetry*

by Julien le Dem
OpenLineage Project Lead

Open Lineage

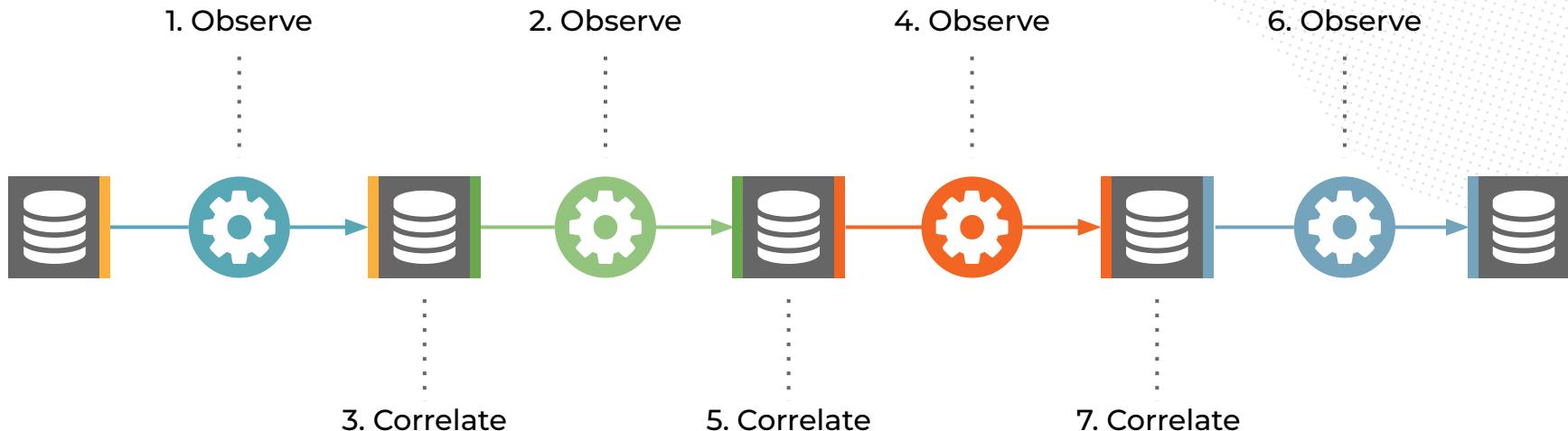


[https://openlineage.io/blog/openlineage-takes-i
nspiration-from-opentelemetry](https://openlineage.io/blog/openlineage-takes-inspiration-from-opentelemetry)

© Copyright. All rights reserved. Not to be reproduced without prior written consent.

Lineage is built on correlations

Dataset names are used to stitch together observations of job runs into a lineage graph.



Open Lineage

Naming conventions

	Formulae	Examples
Datasets	host + database + table bucket + path host + port + path project + dataset + table	postgres://db.foo.com/metrics.sales s3://sales-metrics/orders.csv hdfs://stg.foo.com:salesorders.csv bigquery:metrics.sales.orders
Jobs	namespace + name namespace + project + name	staging.load_orders_from_csv prod.orders_etl.count_orders
Runs	Client-provided UUID	1c0386aa-0979-41e3-9861-3a330623effa

Open Lineage

How to debug?

AIRFLOW__LOGGING__LOGGING_LEVEL=DEBUG

AIRFLOW__OPENLINEAGE__DEBUG_MODE=True

Where to get help?

1. OpenLineage docs / OpenLineage provider docs
2. Slack (bit.ly/OpenLineageSlack)
3. GitHub issue (github.com/apache/airflow/issues)

When seeking help always provide the following:

- Airflow scheduler logs with the logging level set to DEBUG
- Airflow worker logs (task logs) with the logging level set to DEBUG
- OpenLineage events with debug_mode enabled

Open Lineage