Data Lineage in Open Cloud

Michał Modras

Airflow Summit
Let’s flow together
September 19-21, 2023,
Toronto, Canada
Cloud Composer: Apache Airflow in Google Cloud
Data lineage traces the relationship between data sources based on movement of data, explaining how data was sourced and transformed.
Motivation for data lineage
Customer challenges

Inability to understand and trust data

“My manager just asked me if I am using the table from the authoritative source—how can I check this quickly?”

“OK, I am taking a look into this dashboard, but where is this information is coming from? What is the database that is supporting this dashboard?”
Inability to do deterministic change management

“What happens if I drop a table/change a column?”

“We have huge systems, sometimes we change something, and we pray. We pray so not for someone, to go “oh, what happened”?

“Oh this table is changing, but I have queries, I have dashboards, I have infinite things plugged on this table.

I need to map out by hand what happened. “
Inability to do effective root cause analysis

“There are issues in the data in a given table—how can I quickly zero in on the potential cause for the issue?”

“...they reach out to the source systems, SAP systems, third party vendors etc... (debug) can take up to 2 weeks”
Customer challenges

“Inability to meet compliance requirements effectively

“How can I guarantee to authorities that I have not used prohibited data in my models to introduce bias?”

“It’s important for them to understand where everyone is getting the information, what transformations are being done along the way so that they can understand if they are comparing oranges with oranges.”

“In order to have more faith in our system, that what we’re providing to them is correct”
Customer challenges

“It’s important for them to understand where everyone is getting the information, what transformations are being done along the way so that they can understand if they are comparing oranges with oranges.”

Inability to manage data estate at scale

“Help me auto curate/auto apply policies based on lineage to automatically manage data”
Data lineage in Apache Airflow
Airflow Lineage Backends

- (until recently) Native Lineage feature in Airflow.

**Lineage Backend**

It's possible to push the lineage metrics to a custom backend by providing an instance of a LineageBackend in the config:

```python
[lineage]
backend = my.lineage.CustomBackend
```

The backend should inherit from `airflow.lineage.LineageBackend`.

```python
from airflow.lineage.backend import LineageBackend

class CustomBackend(LineageBackend):
    def send_lineage(self, operator, inlets=None, outlets=None, context=None):
        ...
        # Send the info to some external service
```
Lineage Reporting Through Airflow Lineage Backend

Processed data assets lineage: Task Inlets & Outlets → Airflow Lineage Backend

Lineage events → External Lineage Service

Apache Airflow

Google Cloud
Cloud Composer Dataplex Data Lineage Integration

Processed data assets lineage: Task Inlets & Outlets

Dedicated Airflow Lineage Backend

Lineage events

Dataplex
OpenLineage in Apache Airflow and Google Cloud
OpenLineage

- Emerging standard for open source lineage metadata transfer.
OpenLineage in Airflow

- Used to be ‘add-on’.
- **AIP-53** Introduces native support of OpenLineage.
- Airflow OpenLineage integration modernizes its architecture - e.g. abandoning lineage metadata extractors separate to Airflow operators, making lineage metadata definition close to operators.
Cloud Composer OpenLineage Adoption

- Composer is migrating towards the new Airflow lineage architecture (OpenLineage), and leveraging Dataplex’s OpenLineage API.

OpenLineage metadata emitted from operators
Cloud Composer OpenLineage Adoption

- More open and pluggable Lineage in Composer’s Airflow.

OpenLineage metadata emitted from operators
Adoption of OpenLineage in Google Cloud

Dataproc

Cloud Composer

Dataplex

EGERIA

MARQUEZ
Thank you.