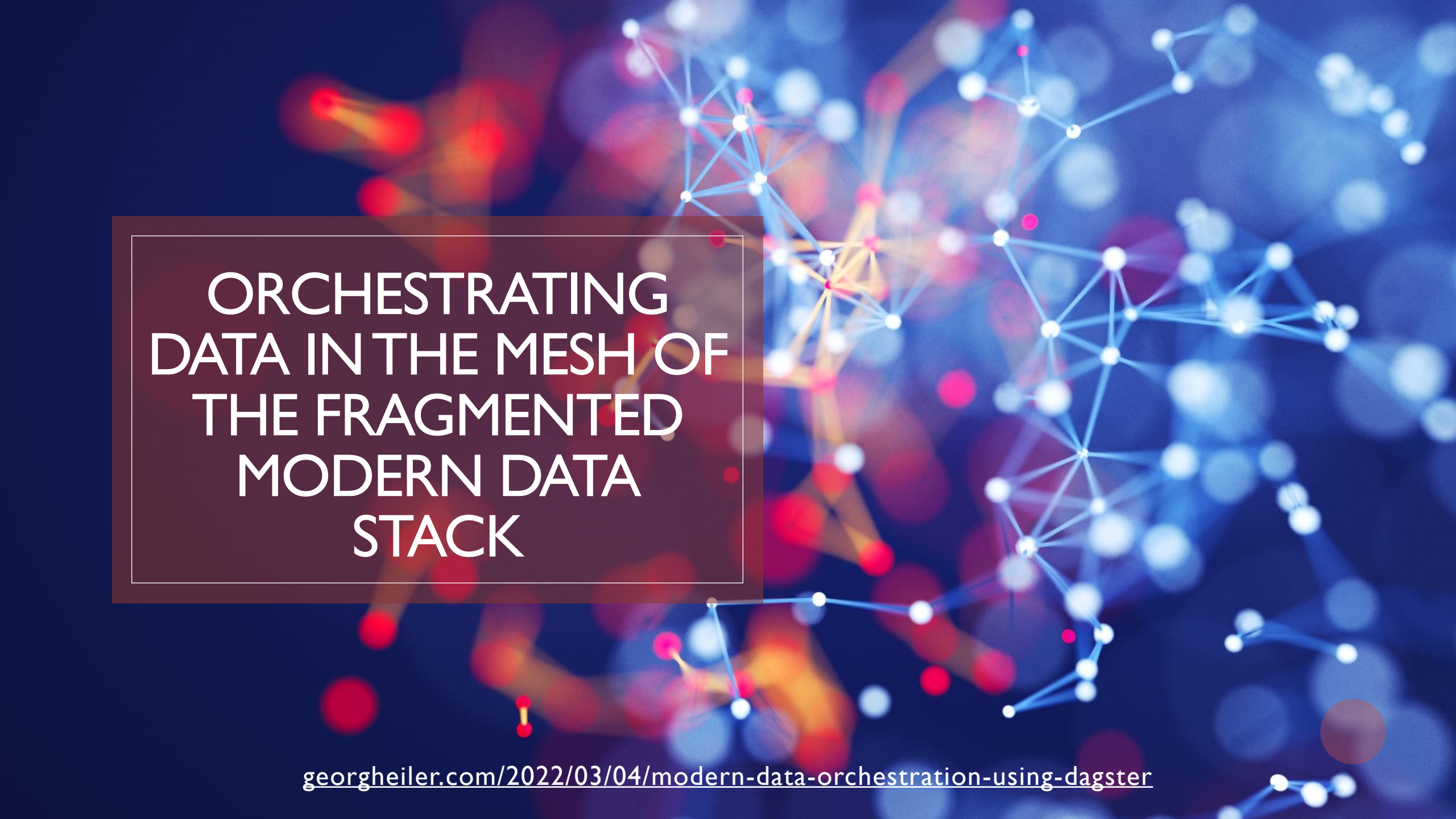

ORCHESTRATING DATA IN THE MESH OF THE FRAGMENTED MODERN DATA STACK
georgheiler.com/2022/03/04/modern-data-orchestration-using-dagster

Agenda

- About me: georgheiler.com (researcher @CSH&TU Wien, lecturer & senior DS @Magenta)
- What is the traditional data stack?
- Modern data stack (MDS)
- Problems with Airflow
- Unbundling Airflow: Silos? Orchestration?
- Introduction to dagster (with demo)
  - Hello world
  - Assets, Ingestion-Transformation-ML ➔ E2E orchestration

Traditional data stack (ETL)

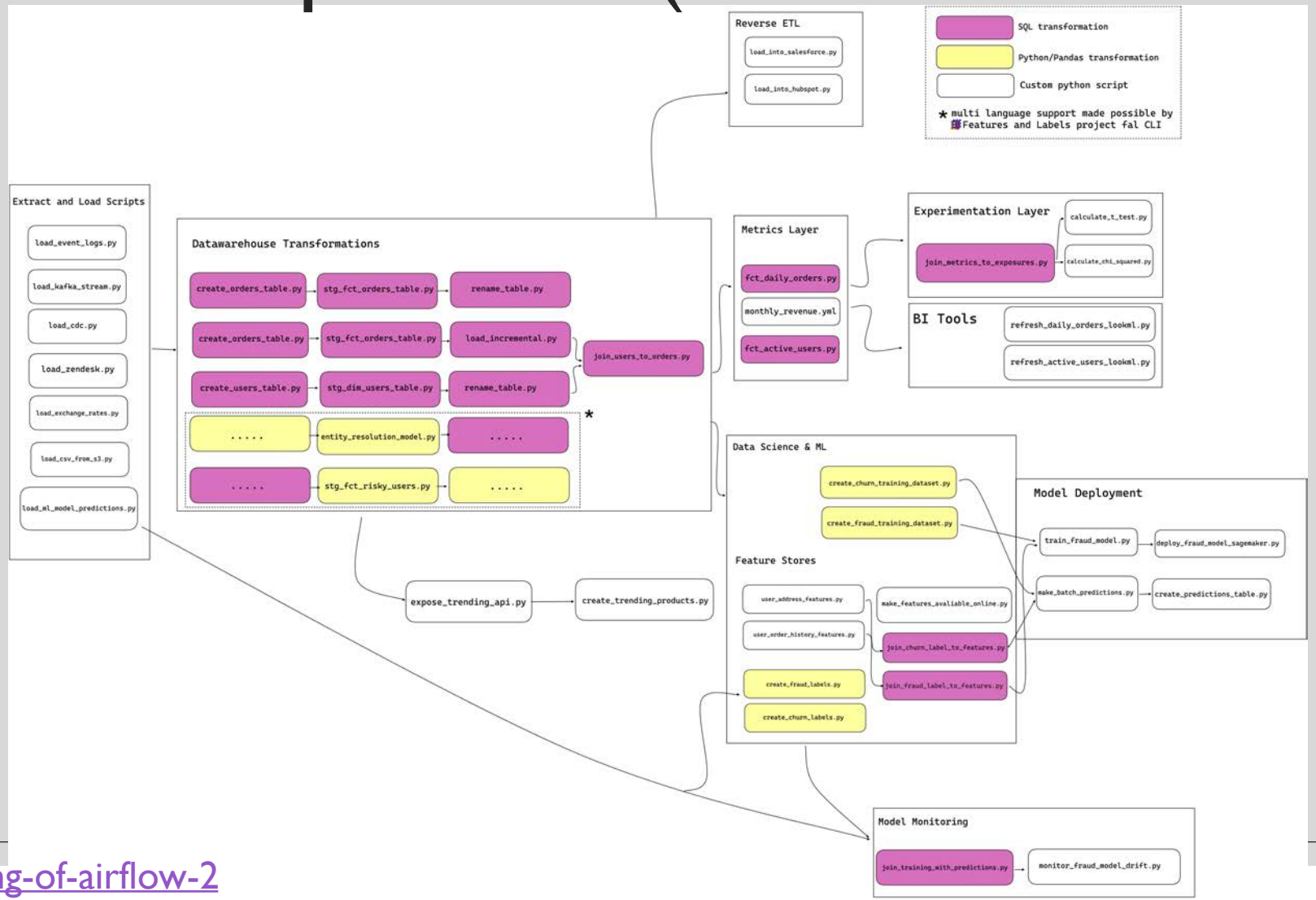
- (Often) custom ingestion processes
- Data warehouse transformations with proprietary tools (Informatica, Talend, …)
  - Not a single source of truth
  - Not a single E2E lineage
  - Multiple separate transformation tools (used by various departments)
- Data mart presentation layer consumed by tools like Tableau, Qlick, …
- Perhaps accompanied by a lake/lakehouse operated in similar fashion

Modern data stack (MDS, ELT)

- Often SaaS (cloud-based) infrastructure is used (snowflake, bigquery, redshift, …)
  - Transformation can be deferred due to the speed of cloud resources (ELT)
- Fivetran, Stitch, Airbyte simplify ingestion with ready made connectors
- Pipeline orchestration oftentimes from **Airflow**
- BI visualization using Looker, Mode, Periscope, Chartio, Metabase, Redash,…

blog.getdbt.com/future-of-the-modern-data-stack

Airflow E2E platform (before unbundling)



blog.fal.ai/the-unbundling-of-airflow-2

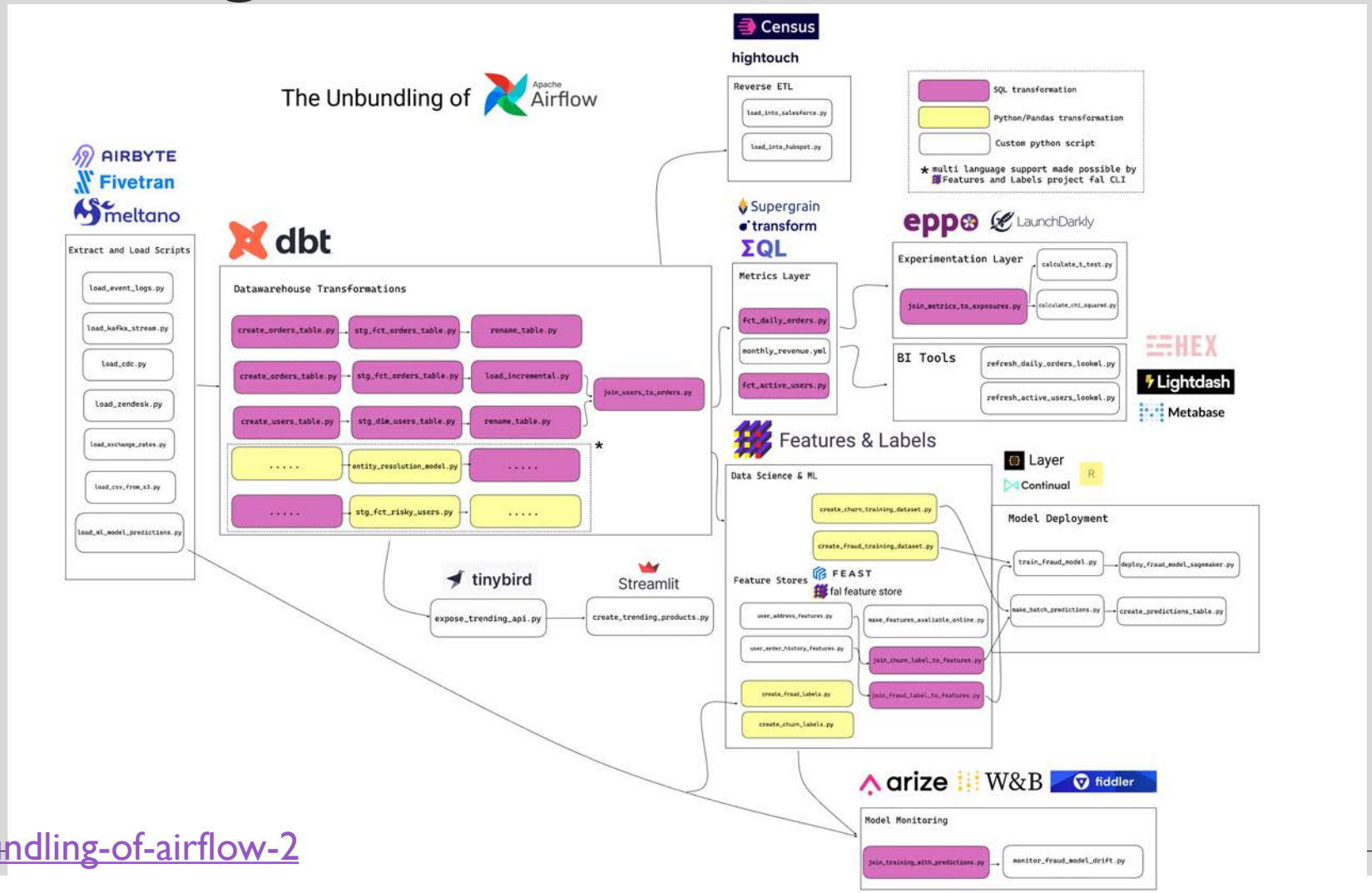
Problems with Airflow

Apache Airflow

- Operator madness:
  - varying quality of operators
  - mixup with businss logic
- No native data dependencies
- No separation of business logic & resources (i.e. IO, cloud services, APIs) => bad testability & bad developer productivity
- No DAG versioning
- Workers not isolated from user code, no resource usage limit per task

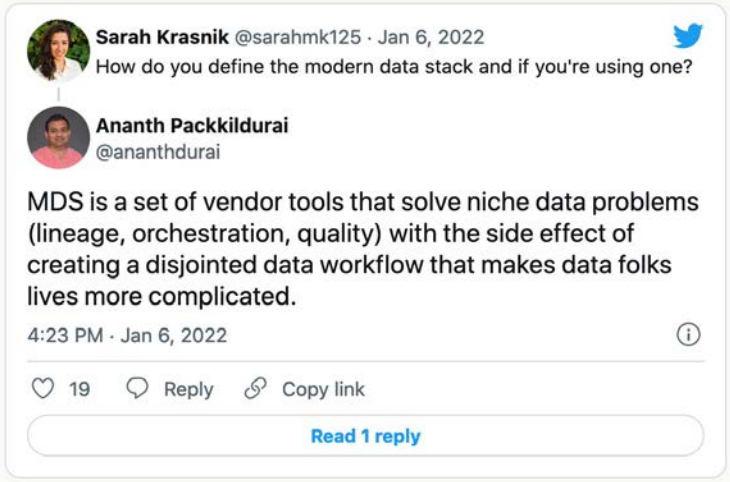
eng.lyft.com/orchestrating-data-pipelines-at-lyft-comparing-flyte-and-airflow-72c40d143aad
medium.com/bluecore-engineering/were-all-using-airflow-wrong-and-how-to-fix-it-a56f14cb0753

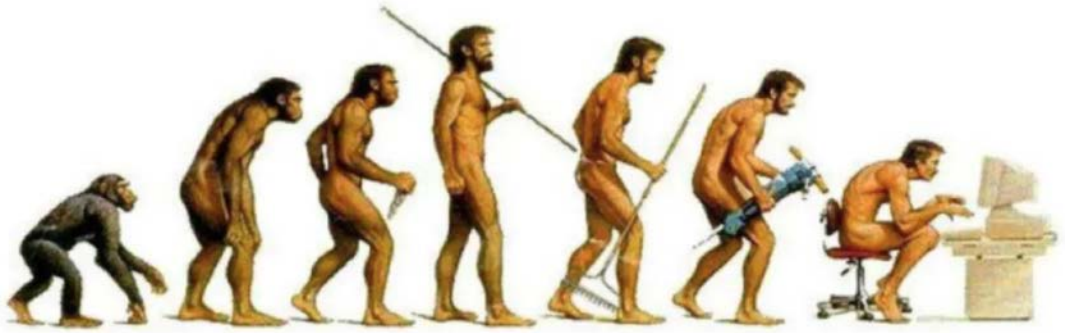
Unbundling Airflow: Silos? Orchestration?



blog.fal.ai/the-unbundling-of-airflow-2

Ananth Packkildurai, the author of the Data Engineering Weekly newsletter, summarizes this state of affairs well:





Overlapping Crons | Workflow Engines | Overlapping Crons in MDS

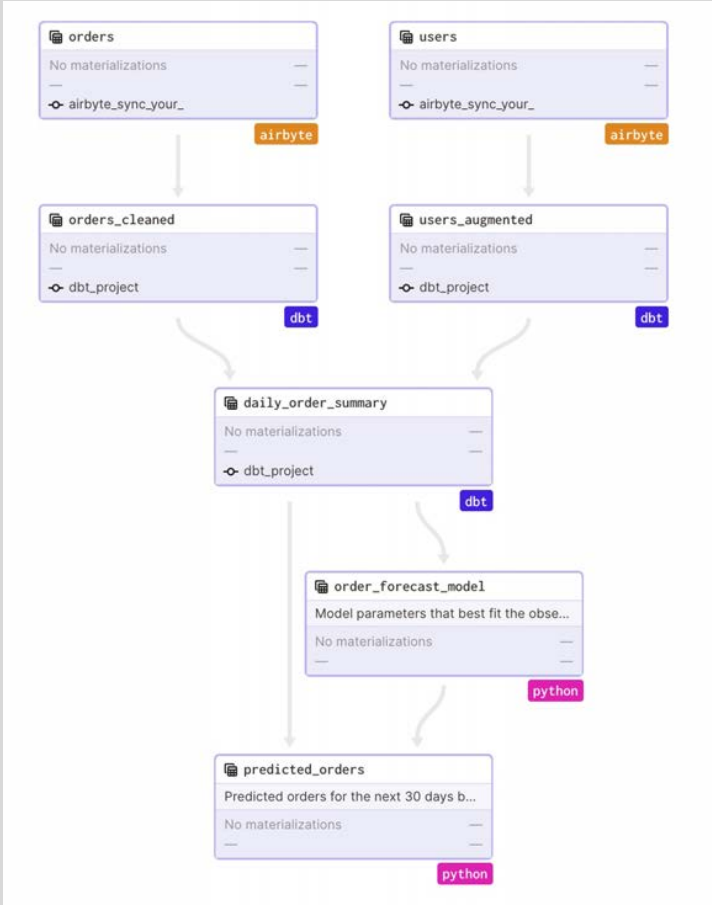
https://dagster.io/blog/rebundling-the-data-platform

UNBUNDLING AIRFLOW:

SILOS? ORCHESTRATION?

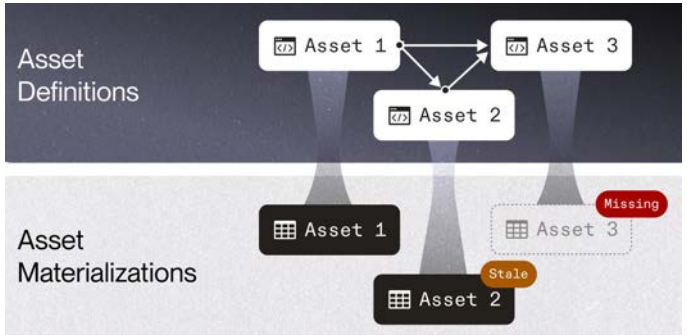
Dagster: Overall orchestration

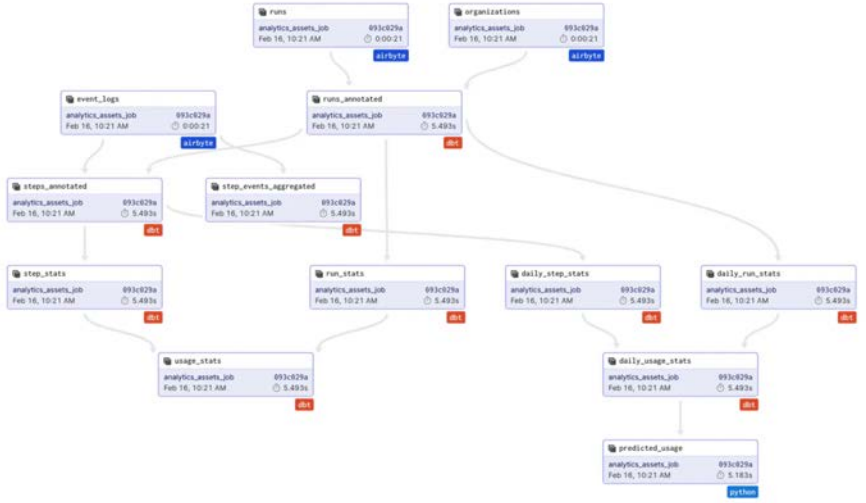
- Alleviate Airflow`s problems
  - Testability first (resources allow for separation of business logic and IO, cloud services, APIs)
  - Increase developer productivity (i.e. locally E2E DEV-test the pipeline with local resources
  - Native data dependencies
  - E2E orchestration (ingest, transform, ML)
    - Lineage first – improve governance
- Assets: Turning the pipeline inside out ➔ Rebundling



georgheiler.com/2022/03/04/connector-goodness-from-airbyte-e2e-lineage

REBUNDLING WITH DAGSTER





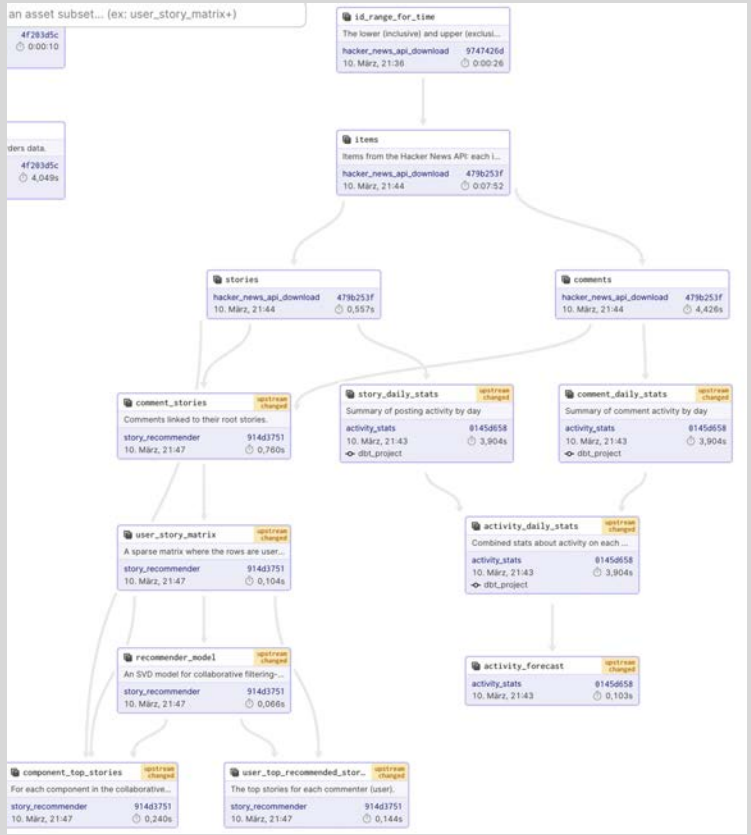
https://dagster.io/blog/rebundling-the-data-platform

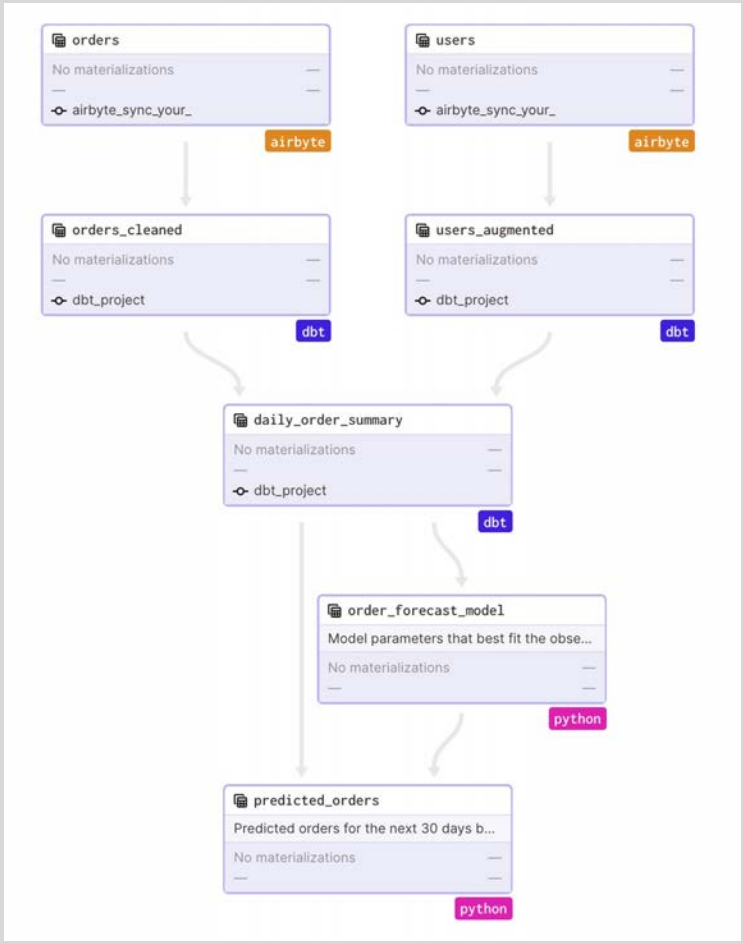
Demo 1 (basics: Hello World + Validation)

```python
from dagster import job, op


@op
def get_name():
    return "dagster"


@op
def hello(name: str):
    print(f"Hello, {name}!")


@job
def hello_dagster():
    hello(get_name())
```

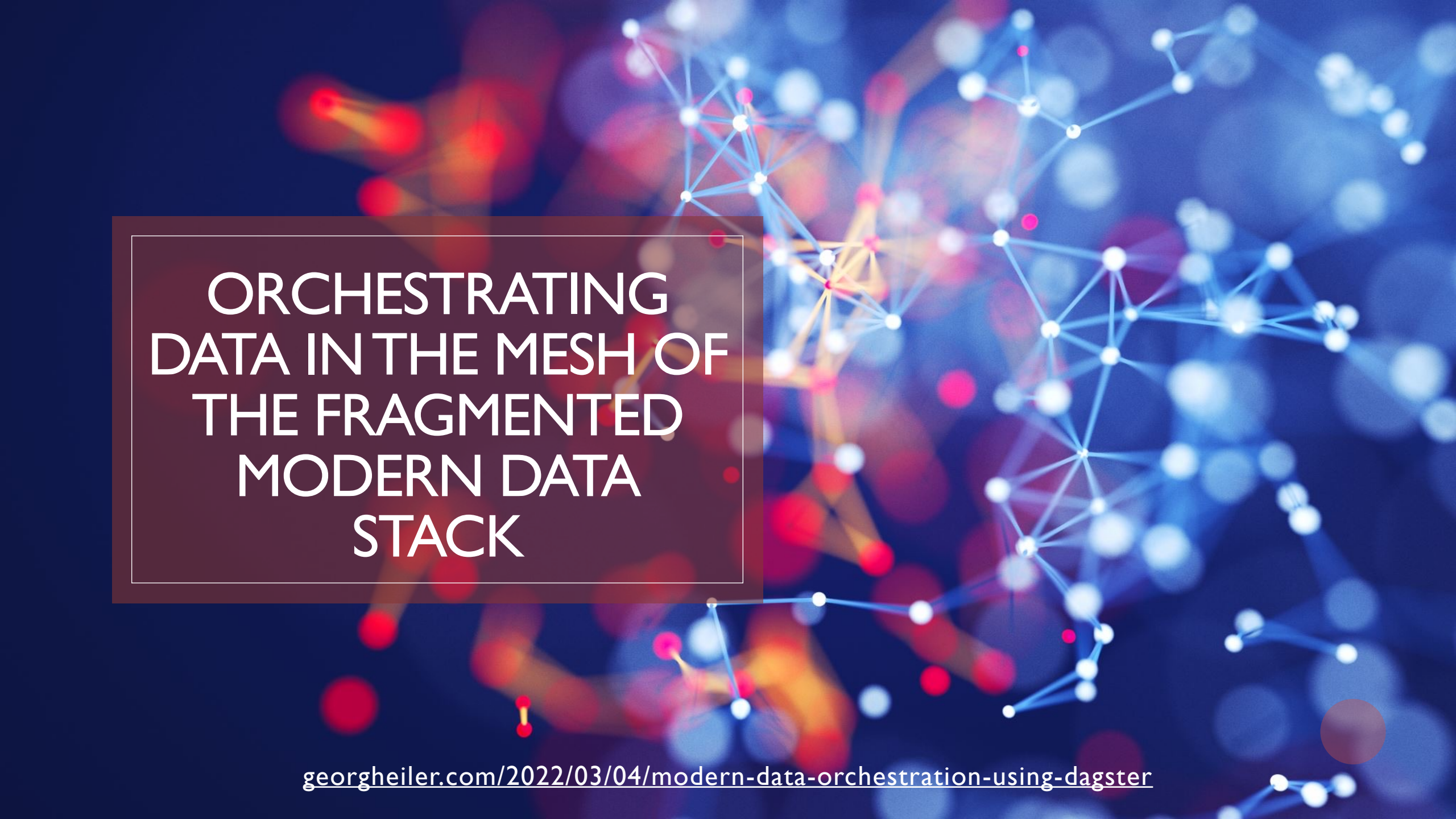
georgheiler.com/2022/03/04/from-hello-world-to-simple-pipelines

Demo II assets, airbyte, DBT, python





georgheiler.com/2022/03/04/connector-goodness-from-airbyte-e2e-lineage
georgheiler.com/2022/03/04/fully-fledged-example-with-resources


ORCHESTRATING DATA IN THE MESH OF THE FRAGMENTED MODERN DATA STACK
georgheiler.com/2022/03/04/modern-data-orchestration-using-dagster