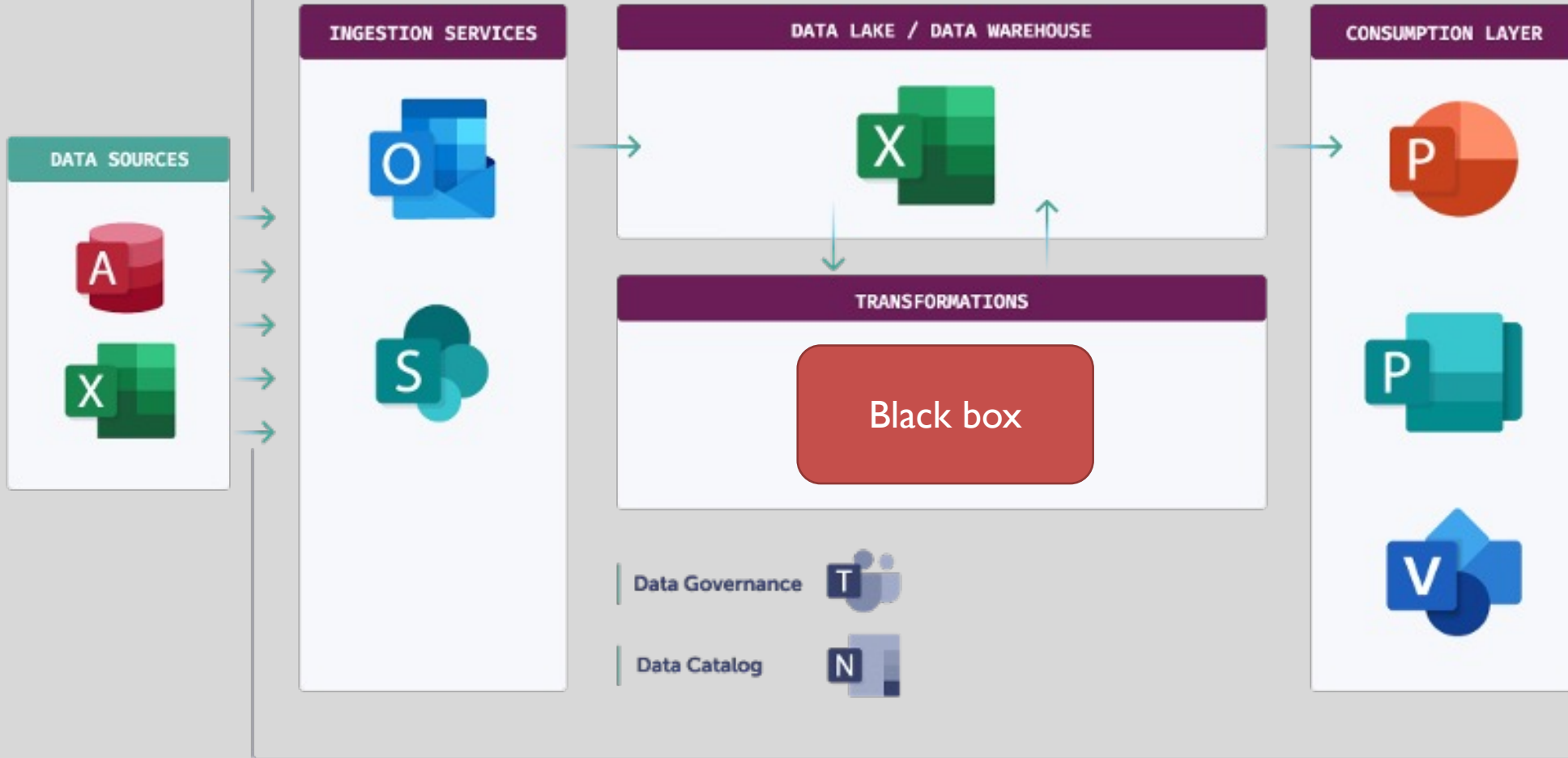


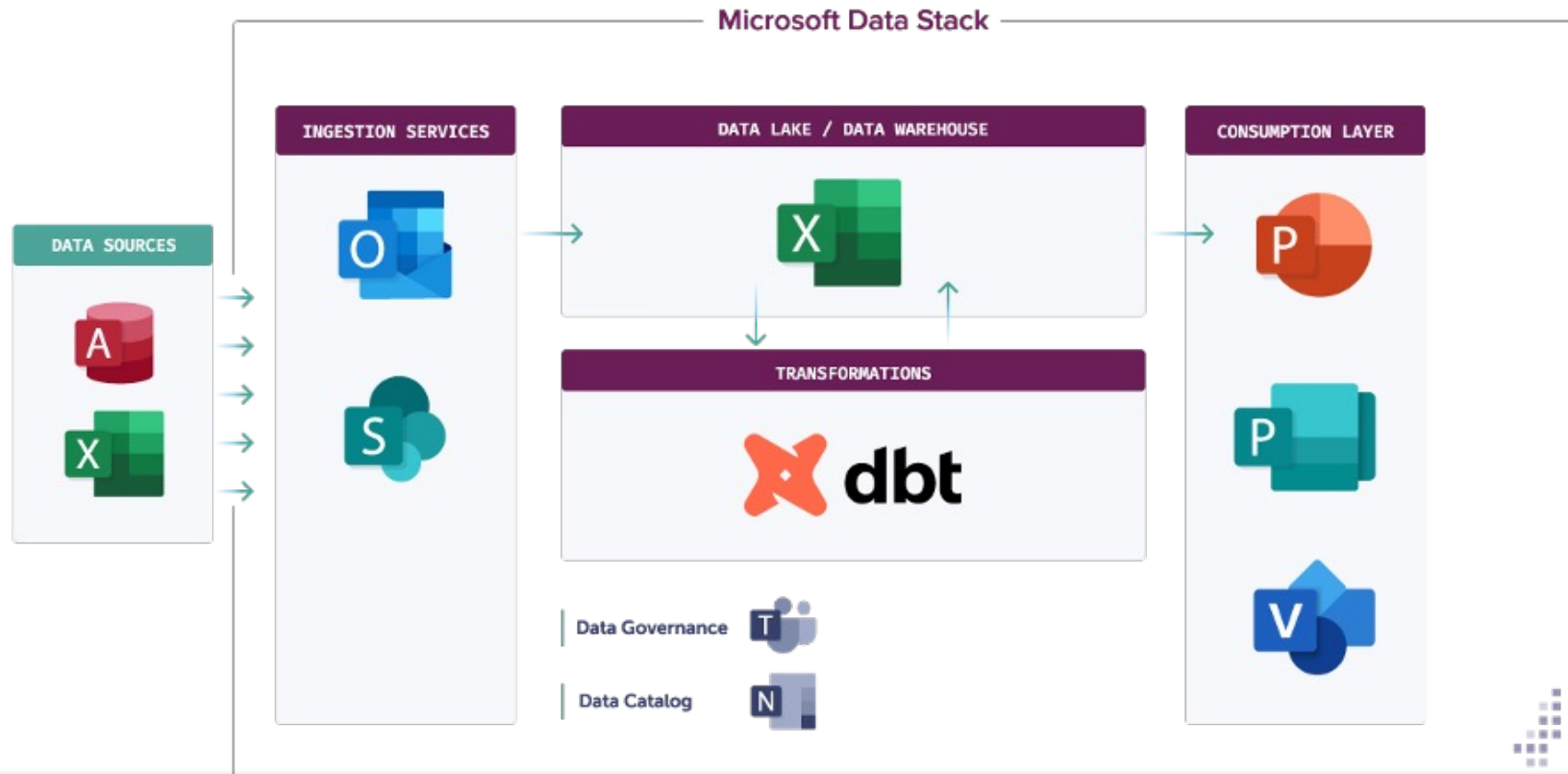
From 0 6 \* \* \* to 

Georg Heiler | 2023

 Magenta

# Microsoft Data Stack





DBT-EXCEL :)

# About me:

[georgHeiler.com](http://georgHeiler.com)



Lecturer, speaker, meetup organizer (VDSG)



Senior software engineer with a specialization in data



PhD in informatics (TU Wien, CSH)

# Agenda



TRADITIONAL  
PLATFORM OVERVIEW



FUTURE STACK

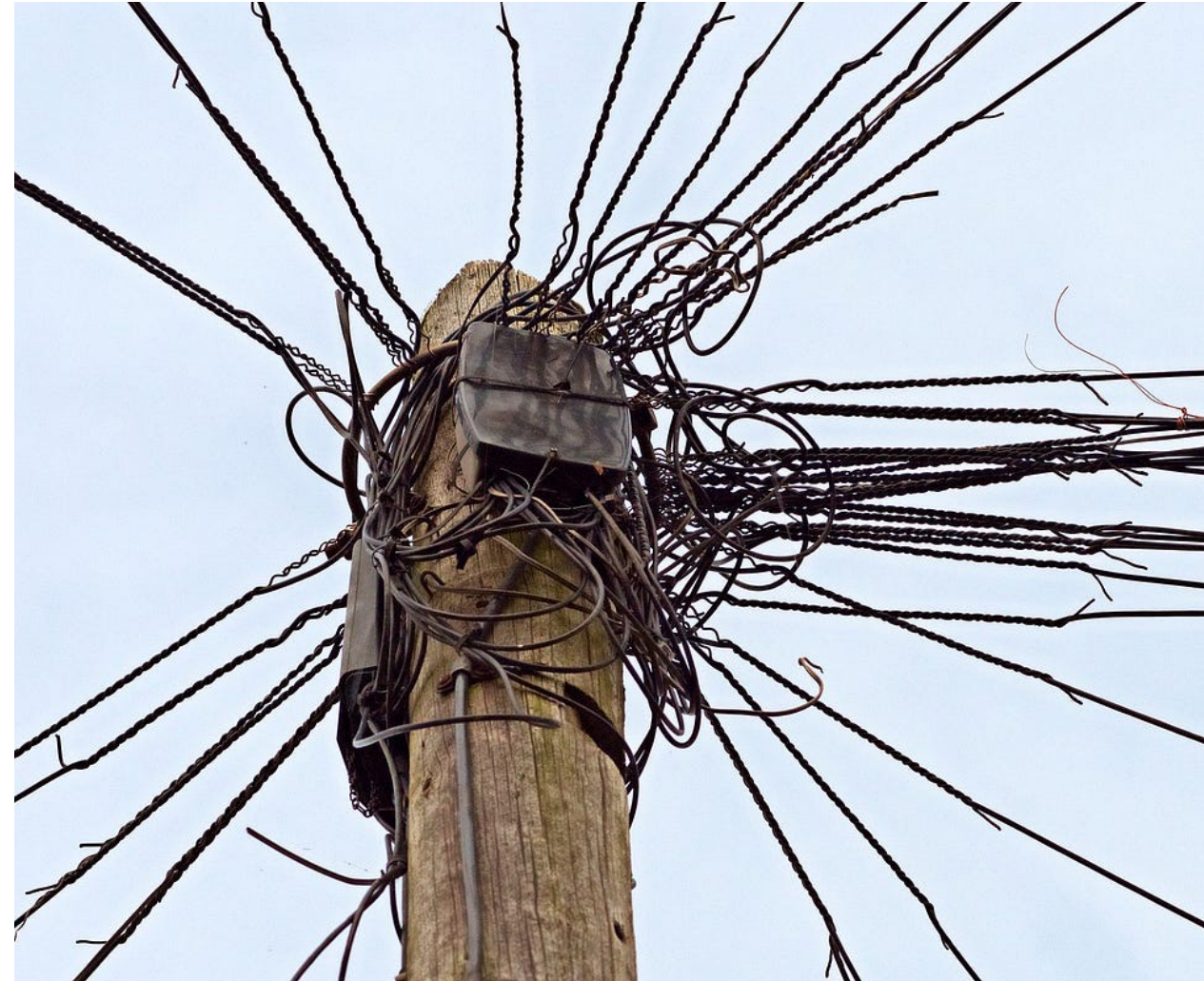
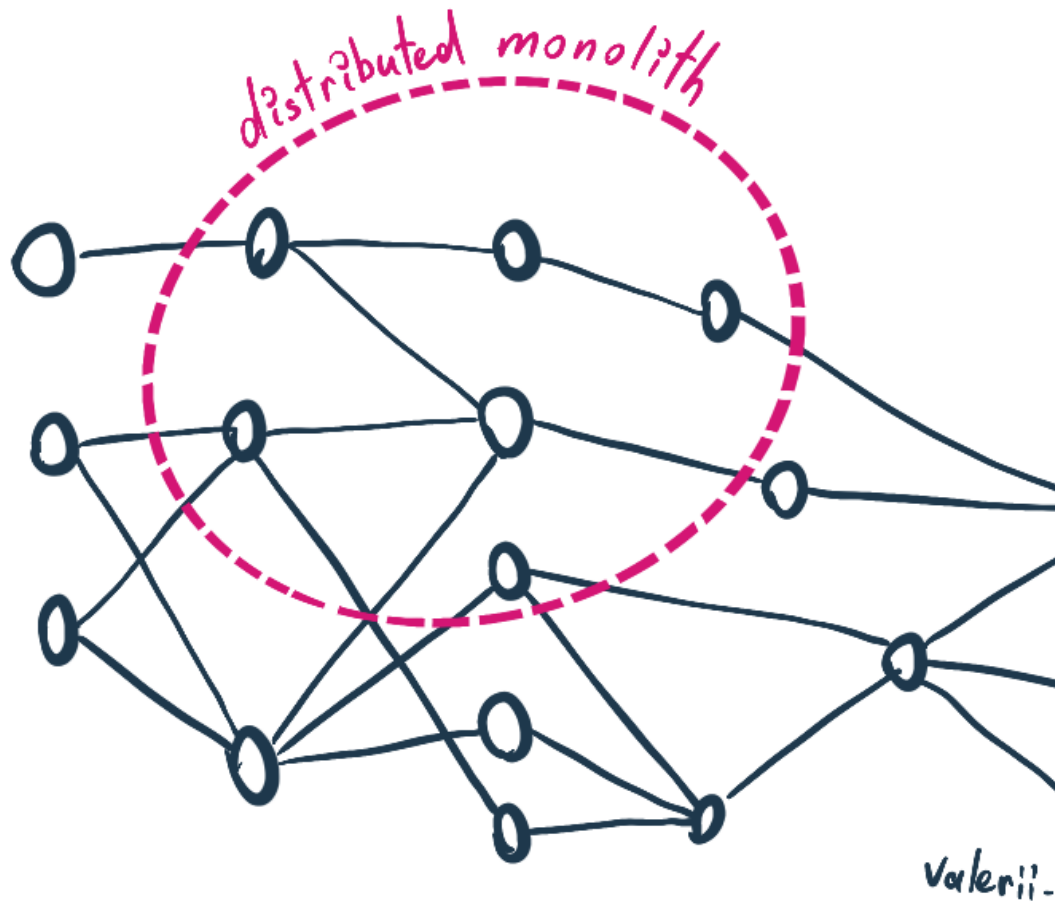


DEMO

# Traditional data stack (ETL)

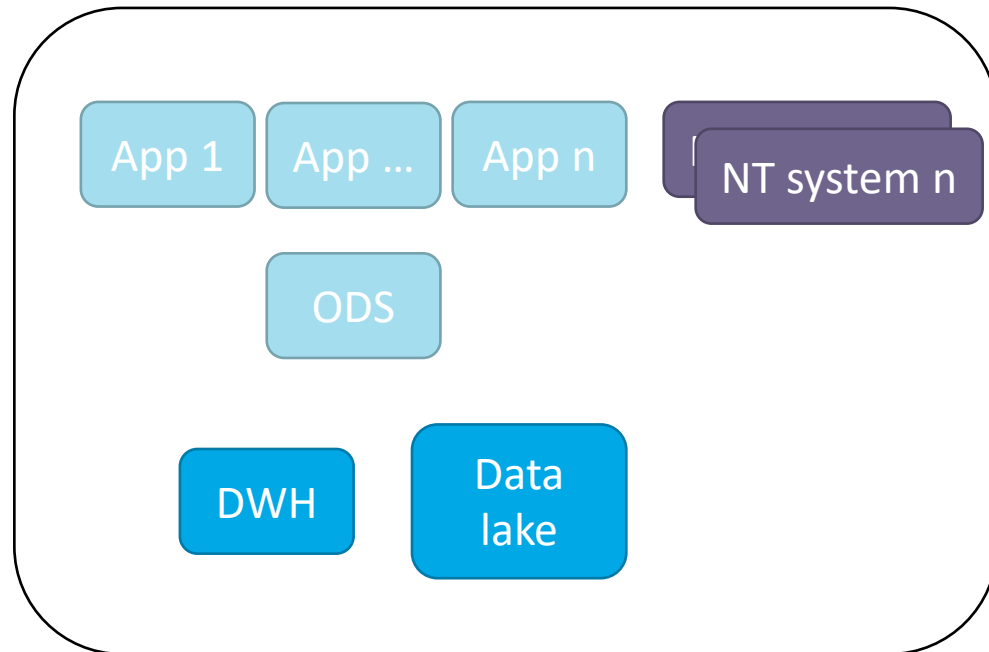
- (Often) custom ingestion processes
- Data warehouse transformations with proprietary tools (Informatica, Talend, plsql, ...)
  - Not a single source of truth
  - Not a single E2E lineage
  - Multiple separate transformation tools (used by various departments)
  - No central scheduler
- No clear separation of layers and domains
- Layer over layer – starving in abstractions
  
- Data mart presentation layer consumed by tools like Tableau, Qlick, ...
- Way too often (hot) fixes are deployed in analytical systems
- Tools chosen where some other joinable data resides (not DDD)

# Distributed (data) monolith

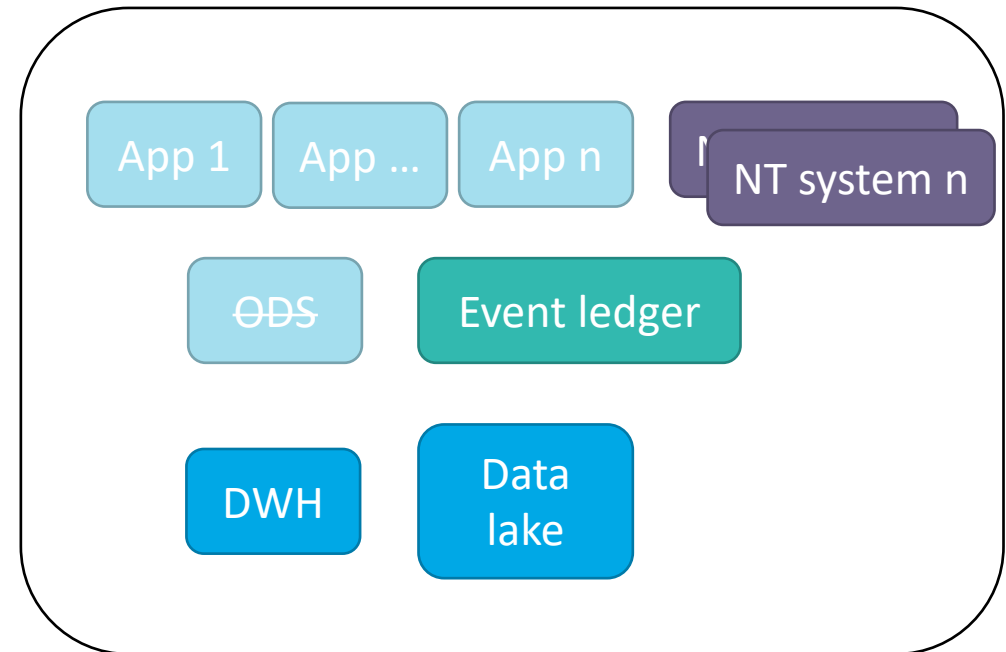


# Data platform

old



current



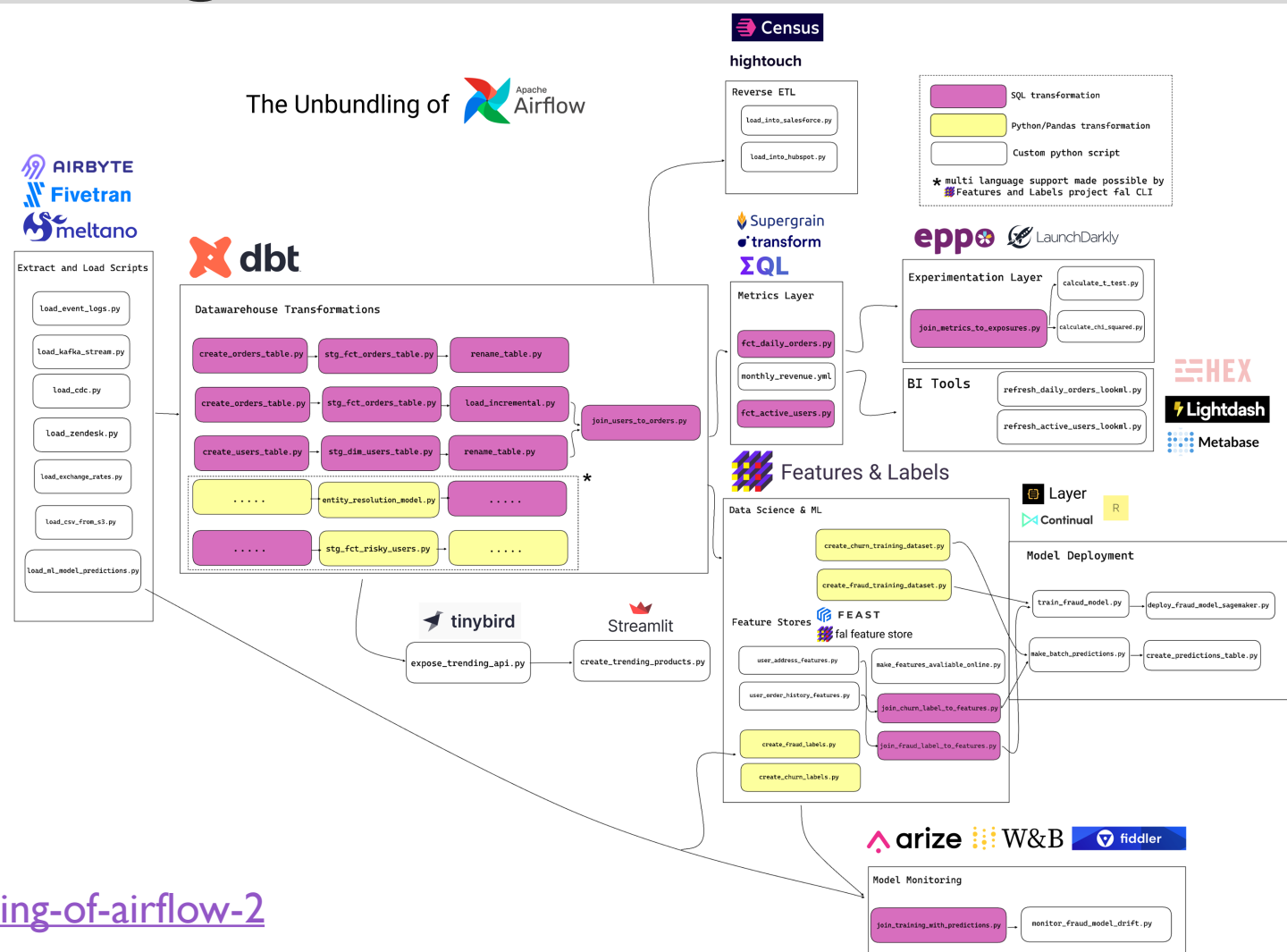




# TECH | MDS

How well are \*modern\* tools integrated?

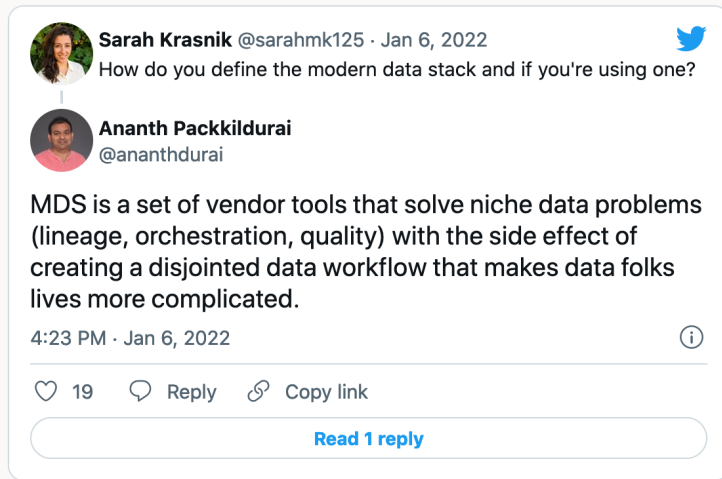
# Unbundling Airflow: Silos? Orchestration?



A group of business professionals in a meeting. A woman in a grey blazer is pointing at a tablet held by another person. Other people are visible in the background, some holding coffee cups. The scene is brightly lit, likely from a window.

WHAT  
MANAGES THE  
MANAGED  
SERVICE?

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



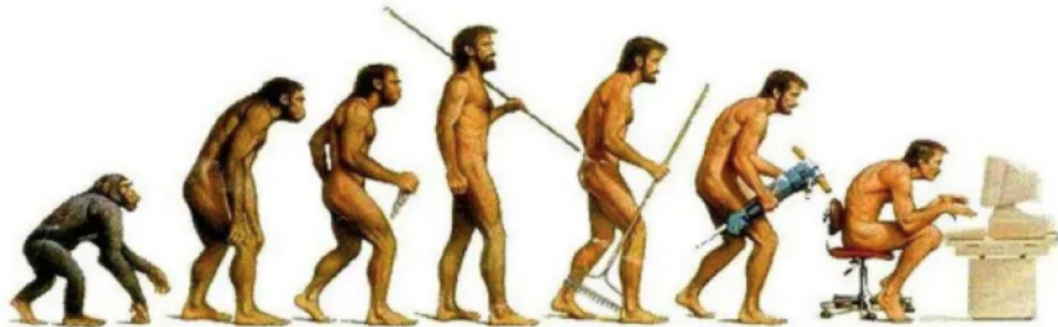
**Sarah Krasnik** @sarahmk125 · Jan 6, 2022  
How do you define the modern data stack and if you're using one?

**Ananth Packkildurai** @ananthdurai  
MDS is a set of vendor tools that solve niche data problems (lineage, orchestration, quality) with the side effect of creating a disjointed data workflow that makes data folks lives more complicated.

4:23 PM · Jan 6, 2022

19 Reply Copy link

[Read 1 reply](#)



**Overlapping  
Crons**

**Workflow  
Engines**

**Overlapping  
Crons in MDS**

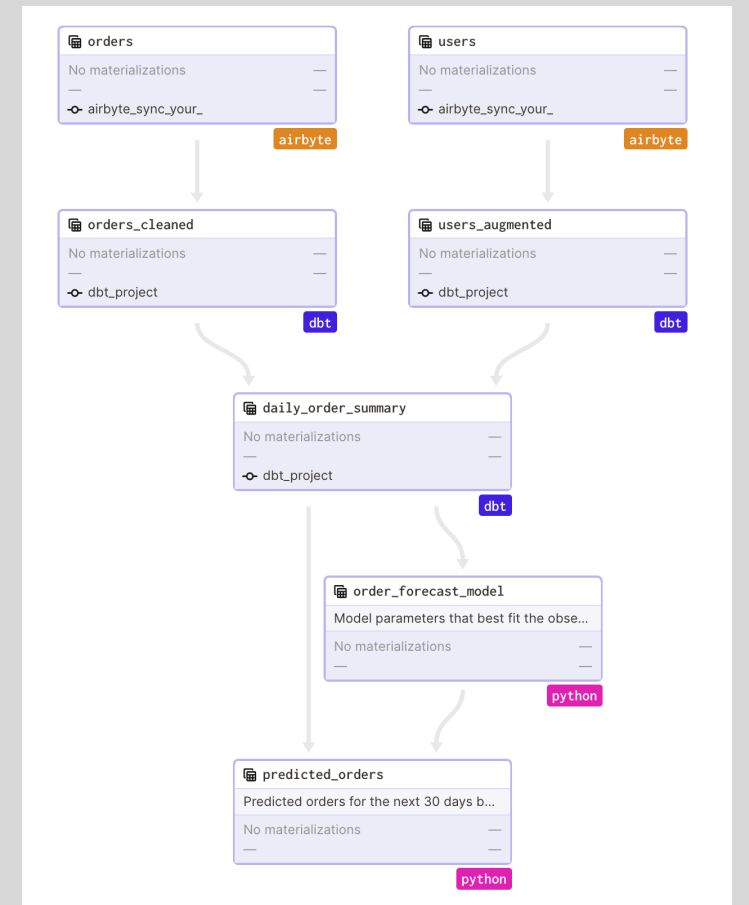
<https://dagster.io/blog/rebundling-the-data-platform>

**UNBUNDLING  
AIRFLOW:  
SILOS?  
ORCHESTRATION?**

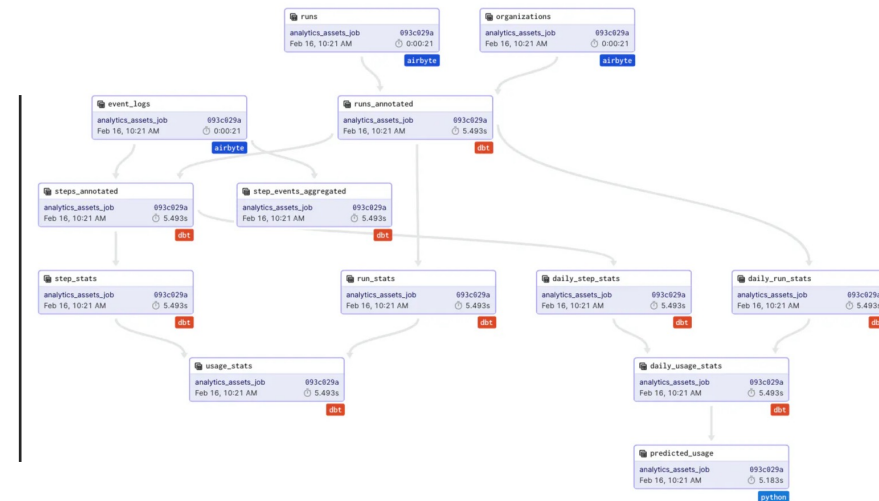
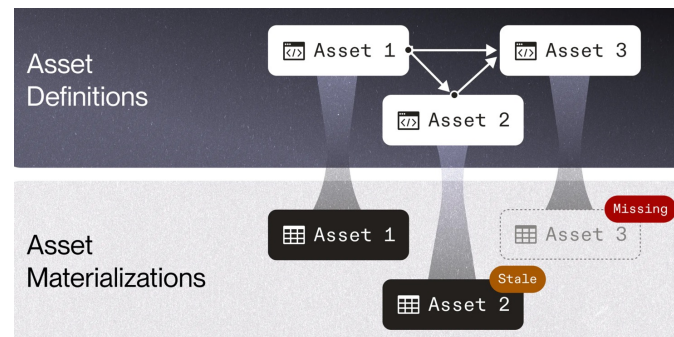
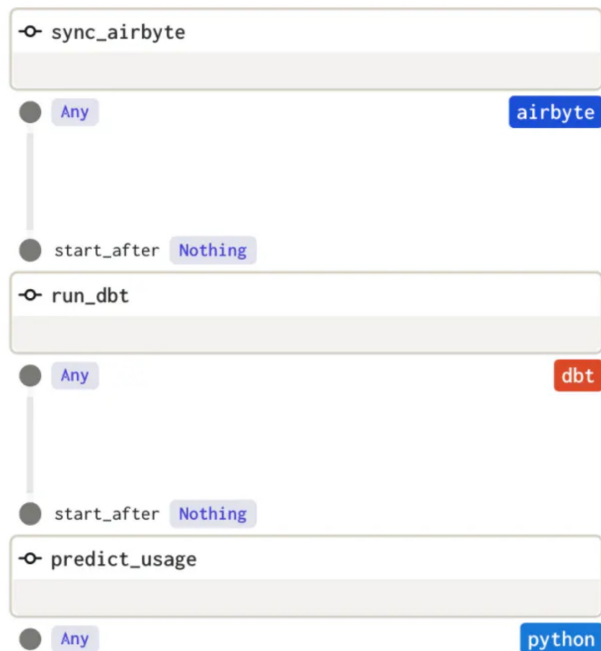


# Dagster: Overall orchestration

- 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



# REBUNDLING WITH DAGSTER





ODPi  
**EGERIA**



Collibra

# OpenLineage

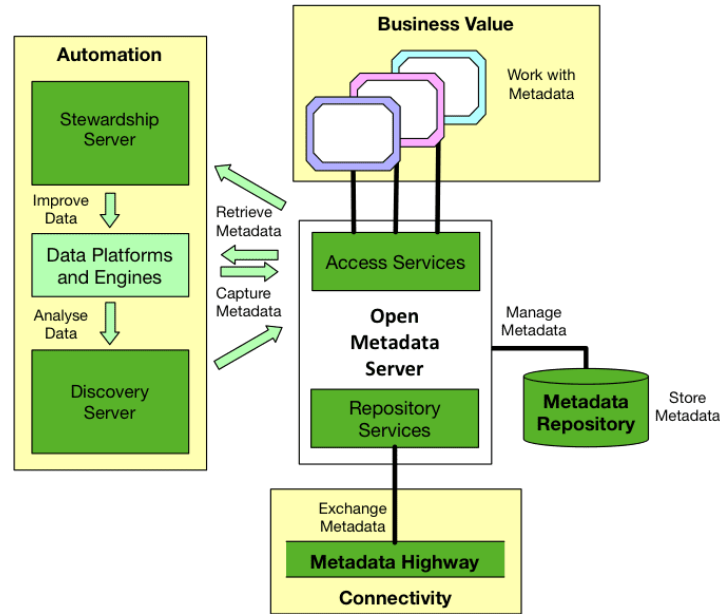
AN OPEN FRAMEWORK FOR DATA LINEAGE COLLECTION AND ANALYSIS

Data lineage is the foundation for a new generation of powerful, context-aware data tools and best practices. OpenLineage enables consistent collection of lineage metadata, creating a deeper understanding of how data is produced and used.



DataHub

## GOVERNANCE



Apache **Atlas**



# Open Metadata

- Simple: write documentation into a file [Intake](#)
- Complex (distributed, enterprise) EGERIA
- MDS? → Open Metadata



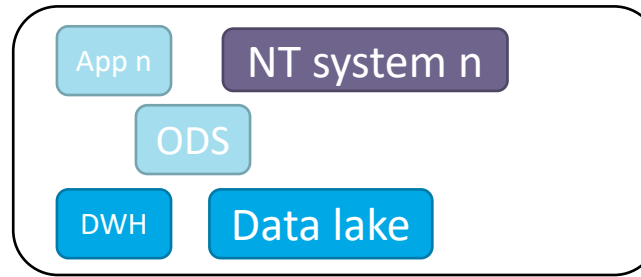
# FUTURE PLATFORM

>> work in progress >>

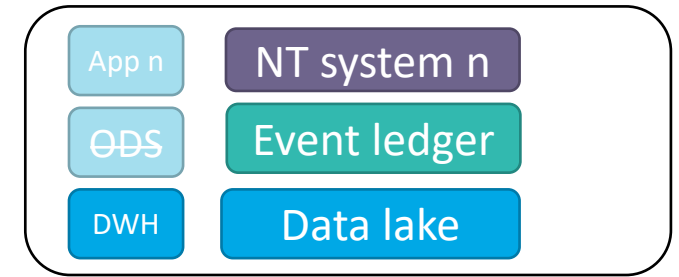


# Future data platform

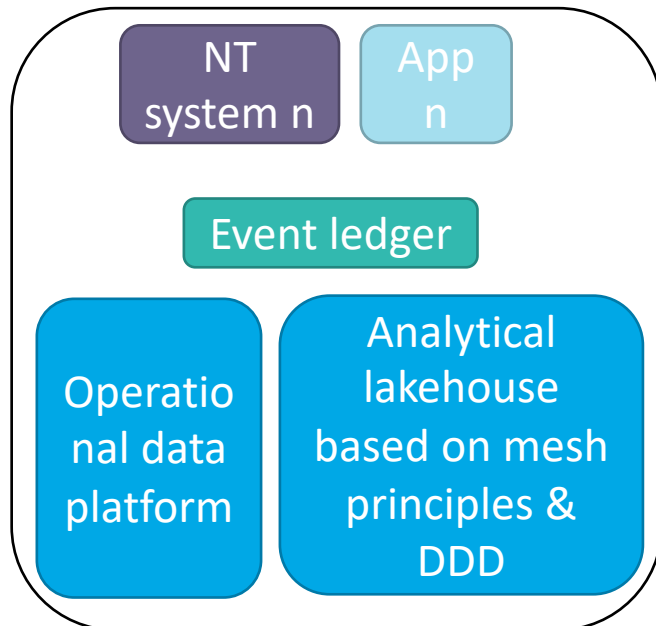
old



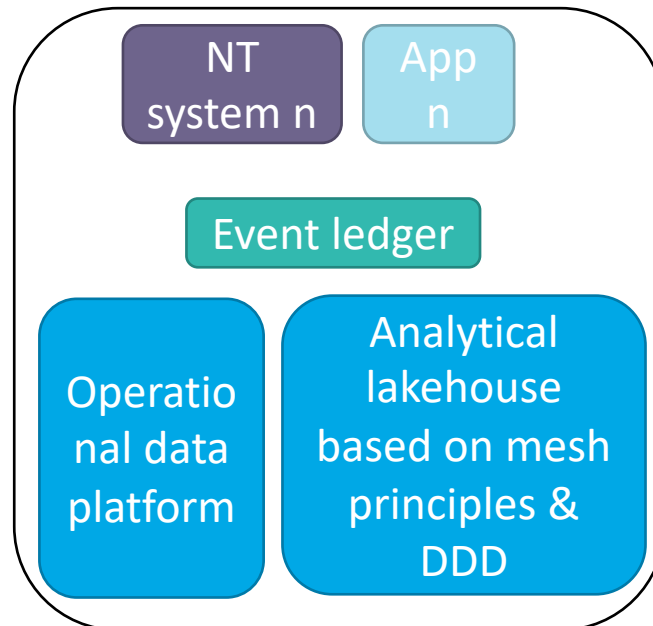
current



On-prem cloud



(public) cloud [n]



Shared:

- Data governance
- Orchestrator
- IaC
- Domain separation (mesh)
- Unify data access

Various SaaS apps



# DEMO EXAMPLES

Dagster, DBT, OpenMetadata + various storage/compute engines (duckdb and more)



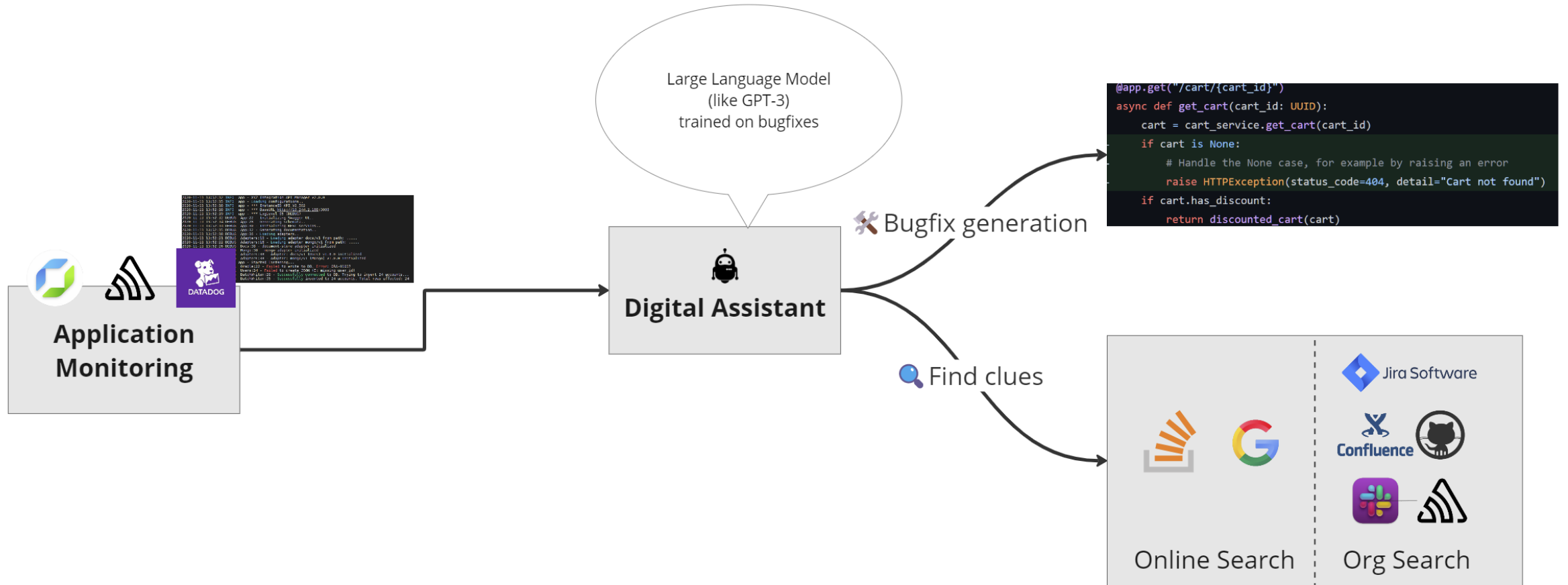
# Bugbusters.ai

---

AI-Assisted bug fixing

*Eliminating bug fixing pain*

# Bugbusters.ai



# bugbusters.ai

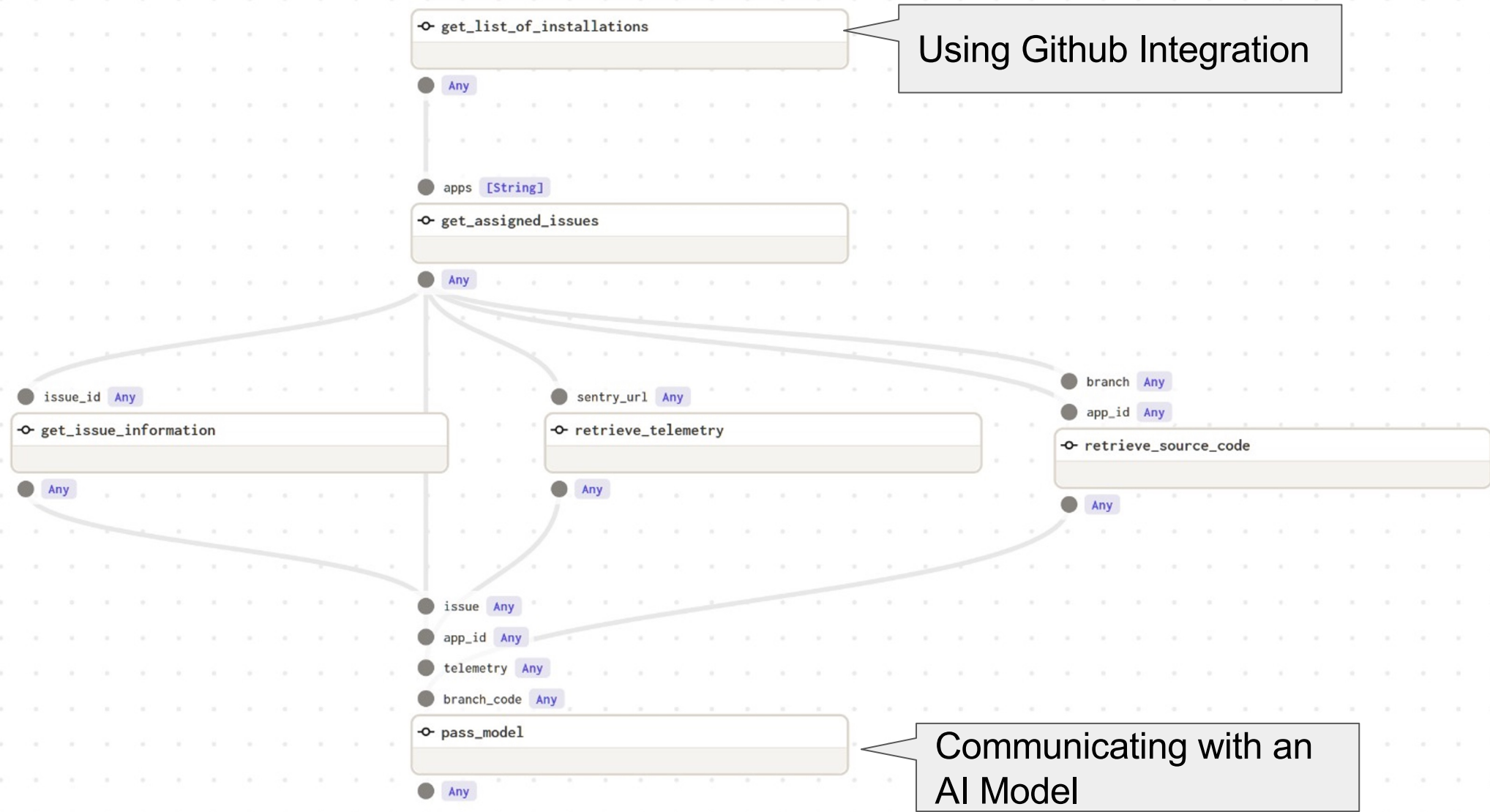
---

- Bugbusters AI  
“Digital assistant, powered by an LLM, capable of generating bugfixes and finding solutions.”
- Eliminating pain and challenges involved by delegating tasks associated with fixing bugs to the assistant.
- Think ChatGPT, optimized for bug fix generation and equipped with internal data

# Challenges

---

- Lots of different monitoring systems (Kibana, Sentry, Datadog) to support
- Differently trained models per customer, i.e. different data pipeline per customers
- Unified way to address data extraction and workflow
- -> dagster



# Findings

## 01

### DDD

- Clear data contracts
- Domain based data ownership
- E2E ownership  
Reorganization (data product)

## 02

### Foundational pillars

- **Data governance**
- **Central data orchestration tool** with support for lazy coupling of domains
- Revival of SQL

## 03

### Enablement

- Hybrid cloud setup (new types of hardware, faster provisioning)
- Use established SaaS tools (less build your own)
- Invest in tooling data engineers love for easier hiring



Foundational principles to help Magenta become more data driven.





# Dagster vs. Airflow resources

Responsible for delivering **advanced analytics & machine learning** solutions.

Data Science team currently consists of 6 people, Data & Insights in total is ~35 internals + ~15 externals.

Central team providing DS solutions to stakeholders across the company.

## Topics include:

- Airflow's operational complexity: Blogpost outlining many of the reasons Airflow is hard to run at scale in production due to architecture limitations. <https://dagster.io/blog/dagster-airflow>
- Airflow's developer pain (esp with local testing): Group 1001 testimonial <https://www.youtube.com/watch?v=Kf0iURfebdA>
- Benefits of declarative scheduling for large, multi-team projects: Whatnot testimonial <https://www.youtube.com/watch?v=ZZZO33MEvF4>
- Overview D-A comparison <https://docs.google.com/spreadsheets/d/1TM-DrqCnMv6QK3jw710GhPU0UP9NQhws/edit#gid=492840621> (request access required)

# Dagster resources

- <https://github.com/dagster-io/hooli-data-eng-pipelines/tree/master>
- <https://github.com/slopp/dagteam>
- <https://stkbailey.substack.com/p/orchestration-isnt-going-anywhere>