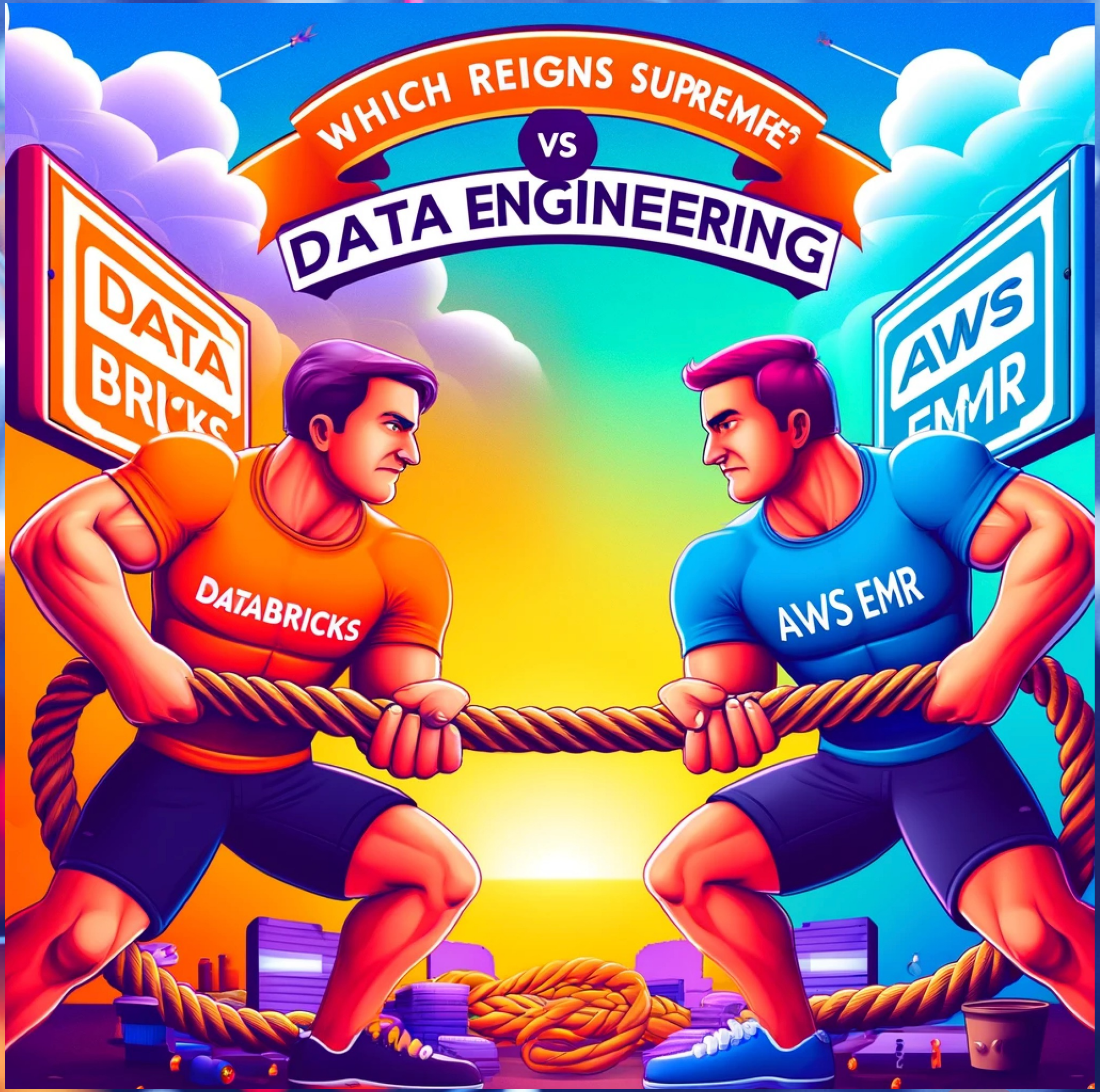
CLOUD ARBITRAGE FOR SPARK PIPELINES

Georg Heiler / Hernan Picatto



georgheiler.com/2024/05/02/cost-efficient-alternative-to-databricks-lock-in

About the Speakers

- Data pipelines & AI
- Academia & Industry (telco)
- Supply Chain, Text analytics & data pipelines, graphs, spatial time series
- Meetup speaker & organizer

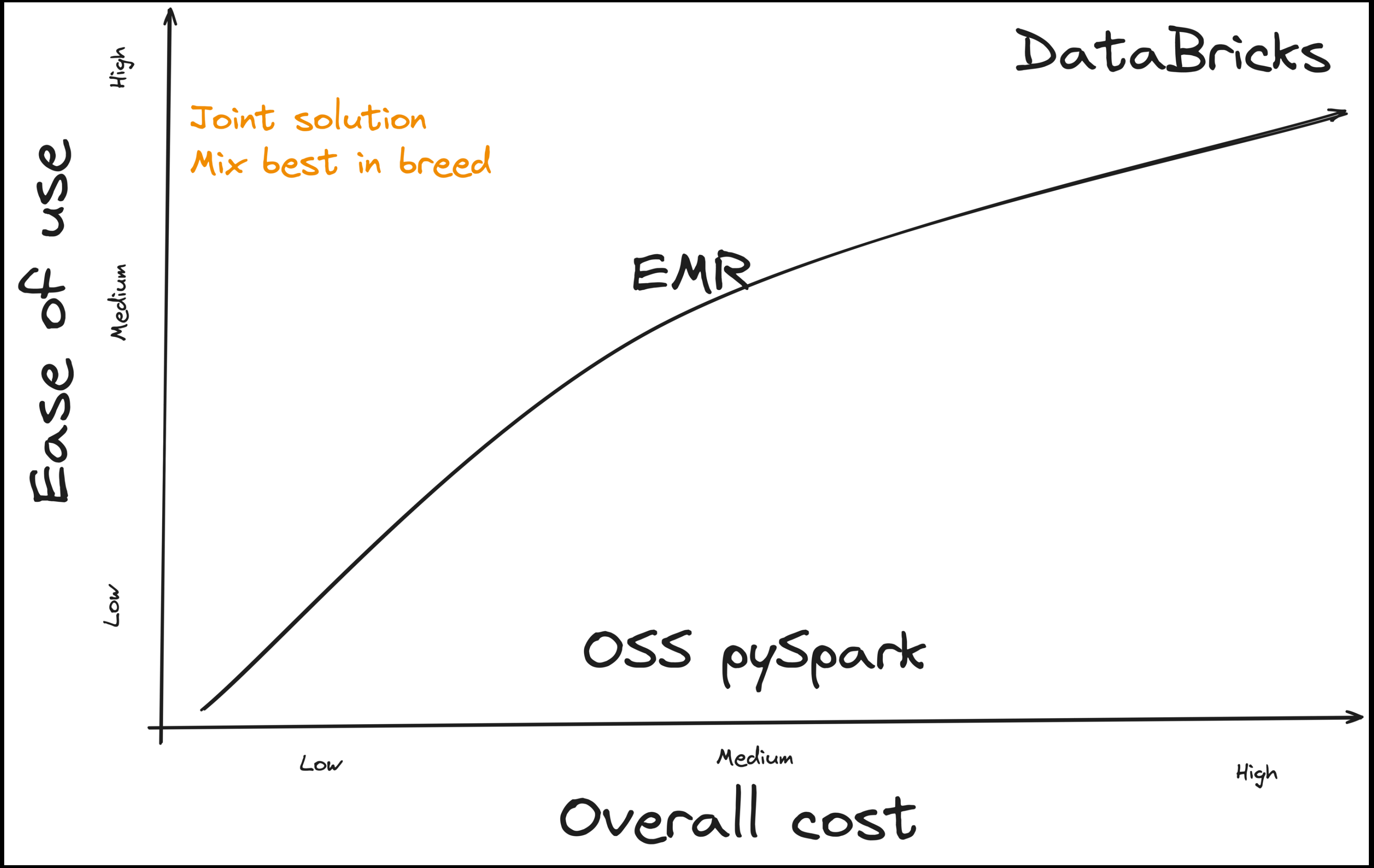
- Researcher @ASCII
- Former JPMorgan Chase
- Time series forecasting and causality detection, EVT analysis

Agenda

- Results Overview
- History
- Problem Description & Vision
- Technology Introduction
- Implementation Architecture
- Results
- Learnings

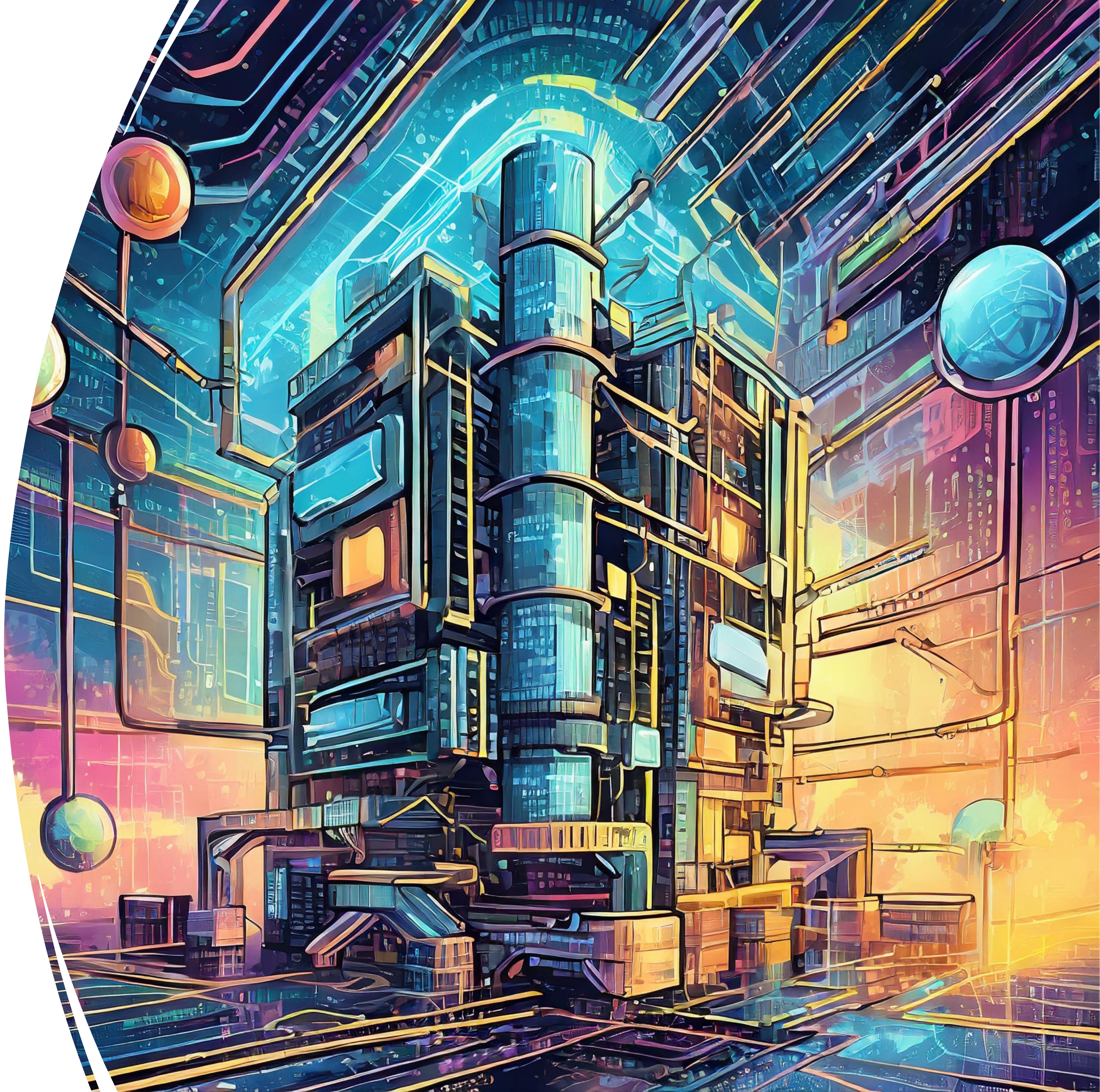
Results at a glance

- Achieved 43% Cost Reduction
- Software Engineering Practices
- Flexibility

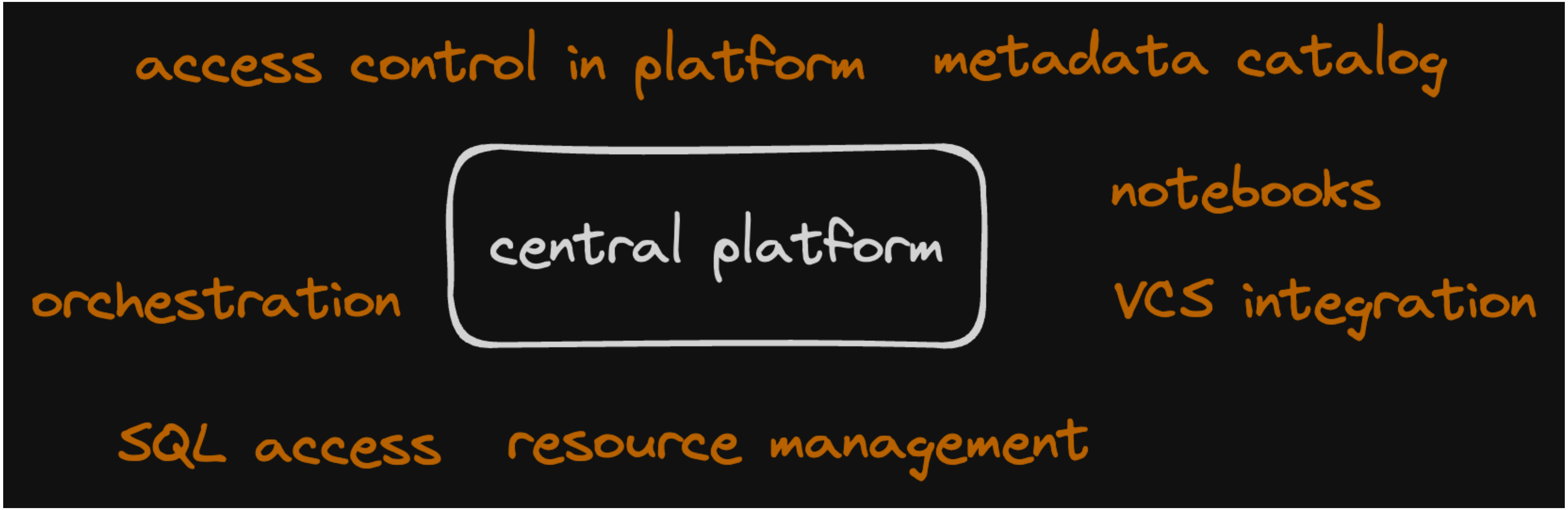


History

- Mainframe
- Data warehouse
- Big Data (Hadoop)
- SQL on large data (Hive, Spark)
- Cloud DWH (Snowflake, bigquery)



PaaS offering



PaaS Solution Comparison

Databricks (DBR)

- Easy to use
- Can be expensive
- Lock-in features (permissions, catalog)
- Proprietary Photon engine

AWS Elastic Map Reduce (EMR)

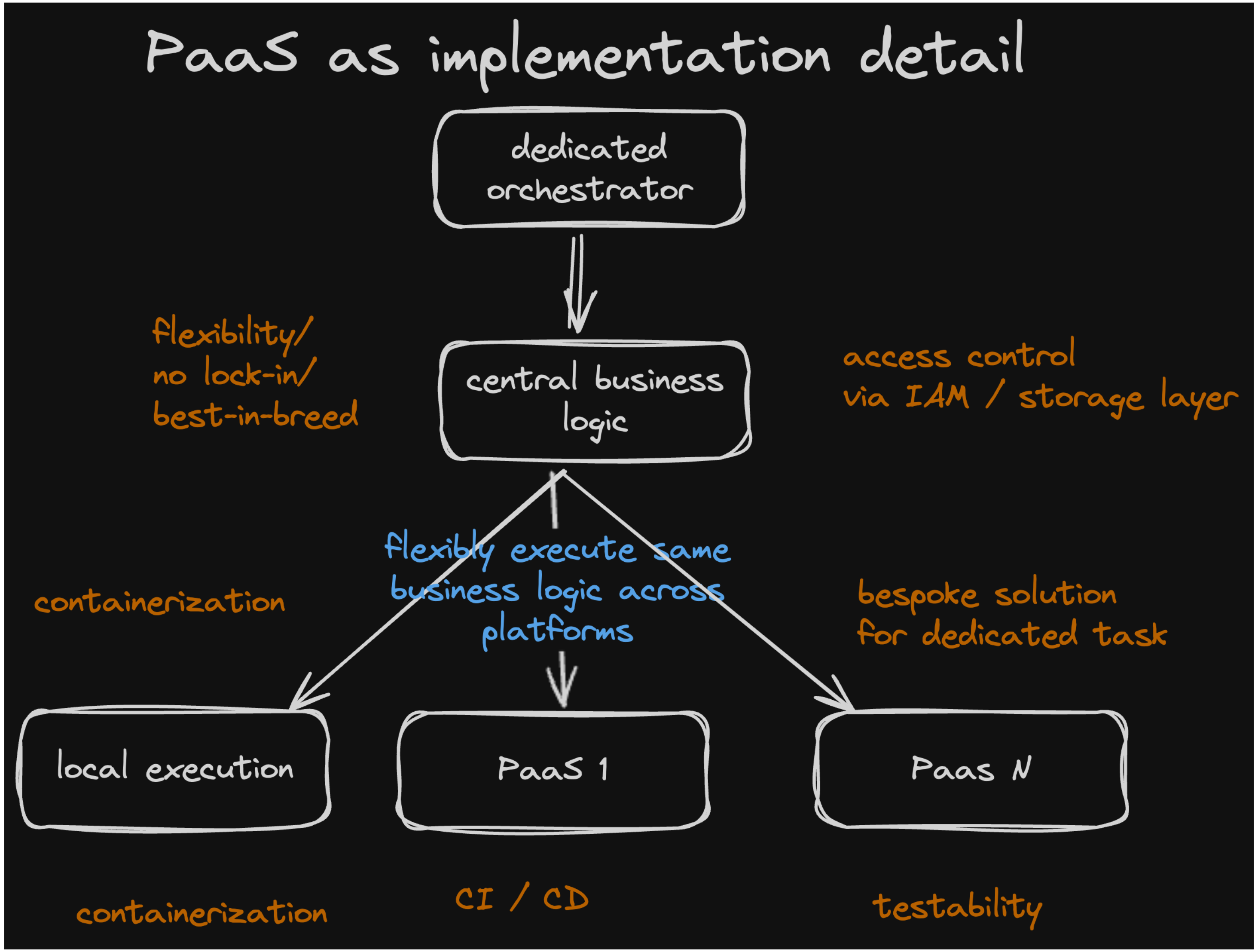
- Price efficient
- Many tuning knobs available (& required)
- OSS Spark managed (scaled)

Challenges

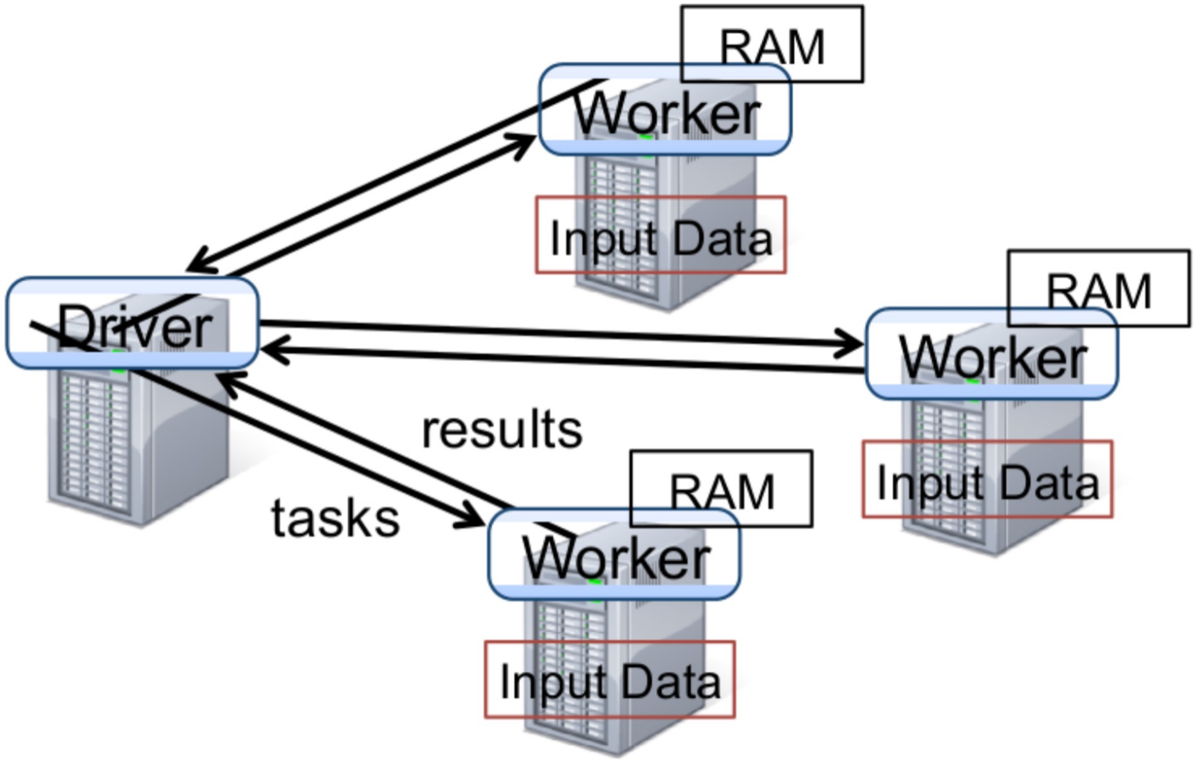
- Runaway expenses (usage-based pricing)
- Missing software engineering best practices (notebooks)
- Developer productivity reduced
- Vendor lock-in

Vision

- 0-cost switch
- Software engineering practices
- Cost reduction
- Reduce lock-in

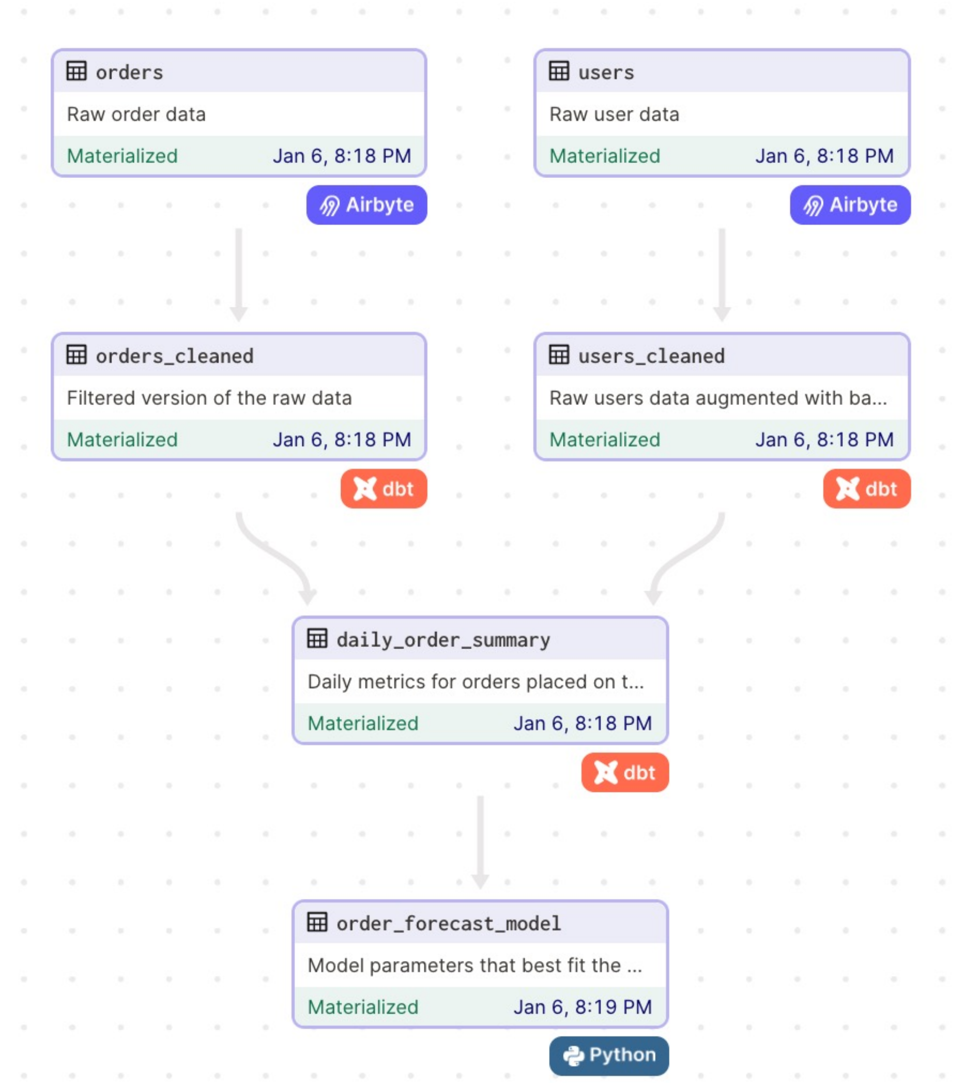

PaaS as implementation detail
dedicated orchestrator
central business logic
flexibility/ no lock-in/ best-in-breed
access control via IAM / storage layer
containerization
flexibly execute same business logic across platforms
bespoke solution for dedicated task
local execution
PaaS 1
PaaS N
containerization
CI / CD
testability

Spark at a glance



Dagster introduction

dagster



orders
Raw order data
Materialized Jan 6, 8:18 PM

users
Raw user data
Materialized Jan 6, 8:18 PM

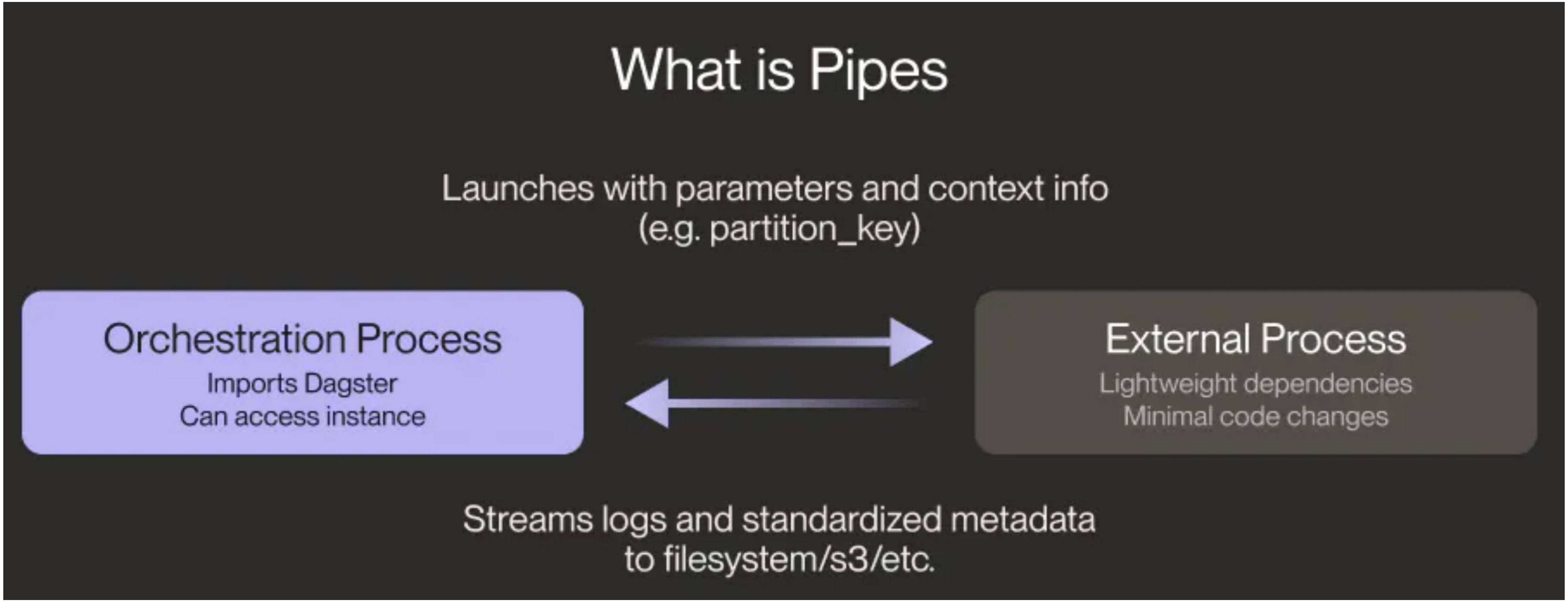
orders_cleaned
Filtered version of the raw data
Materialized Jan 6, 8:18 PM

users_cleaned
Raw users data augmented with ba...
Materialized Jan 6, 8:18 PM

daily_order_summary
Daily metrics for orders placed on t...
Materialized Jan 6, 8:18 PM

order_forecast_model
Model parameters that best fit the ...
Materialized Jan 6, 8:19 PM

Dagster-pipes



What is Pipes

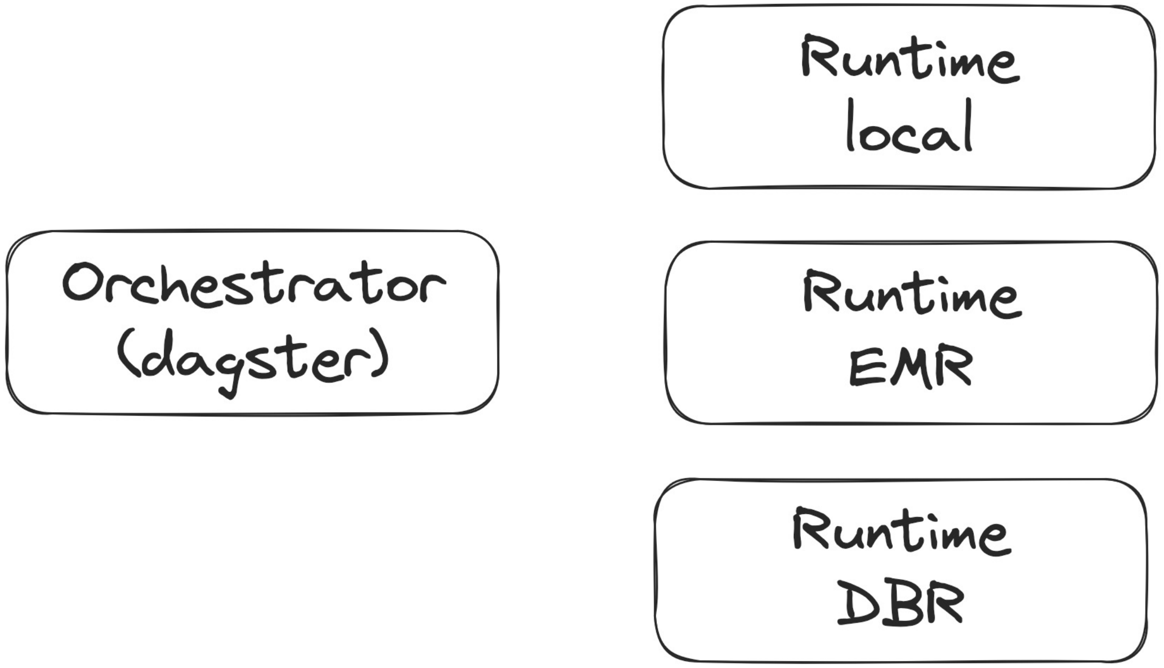
Launches with parameters and context info
(e.g. partition_key)

Orchestration Process
Imports Dagster
Can access instance

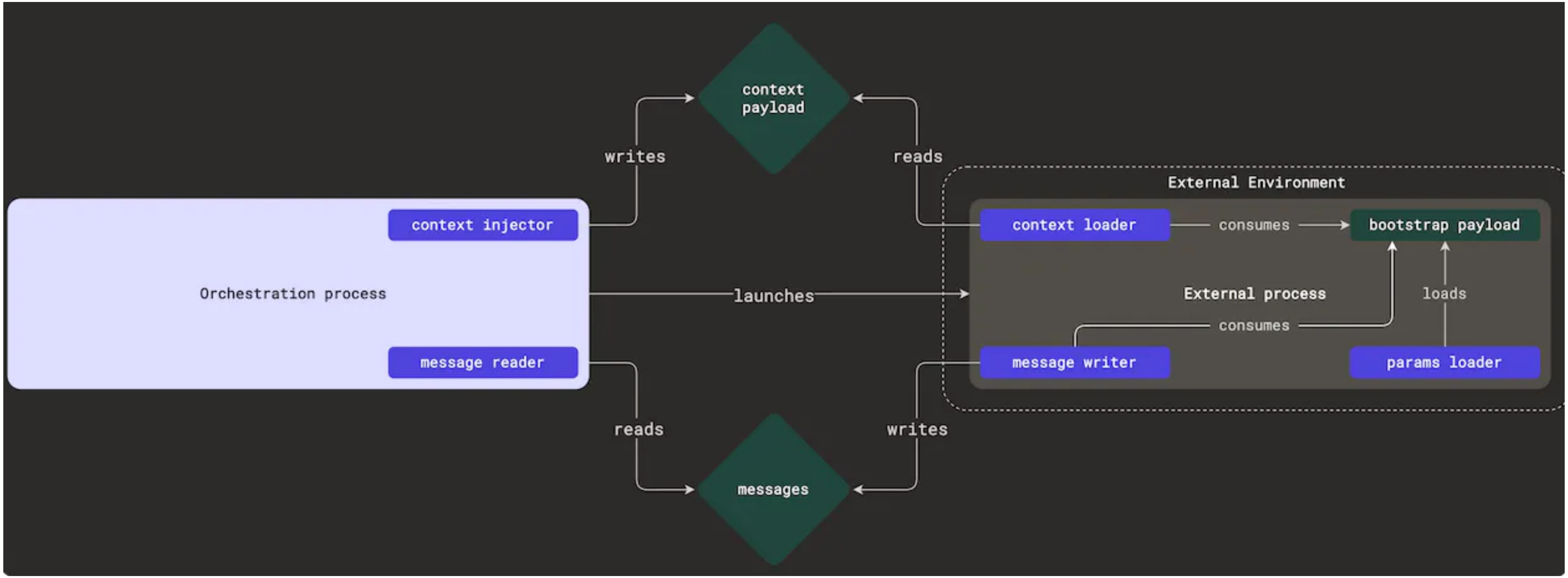
External Process
Lightweight dependencies
Minimal code changes

Streams logs and standardized metadata
to filesystem/s3/etc.

High-level Architecture

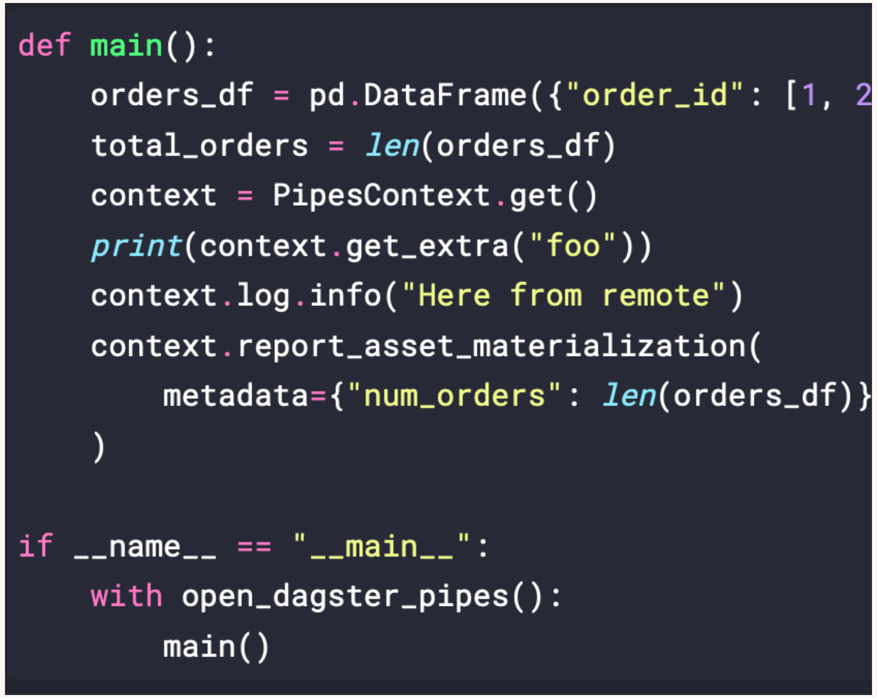


Dagster-pipes - Architecture



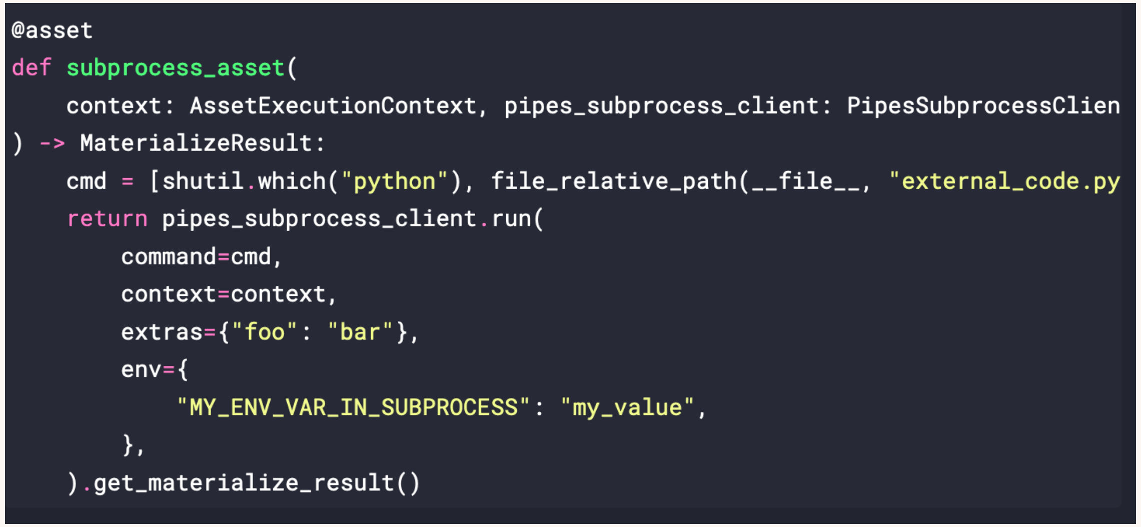
Dagster-pipes Sample

External code (with metadata)



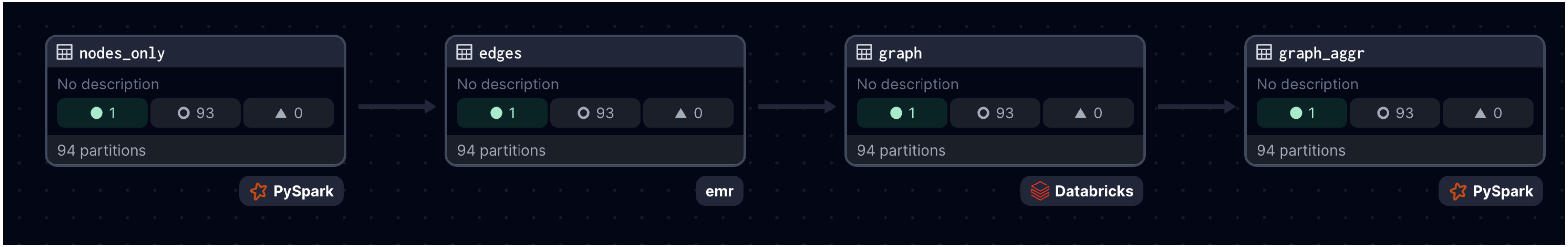
```python
def main():
    orders_df = pd.DataFrame({"order_id": [1, 2
    total_orders = len(orders_df)
    context = PipesContext.get()
    print(context.get_extra("foo"))
    context.log.info("Here from remote")
    context.report_asset_materialization(
        metadata={"num_orders": len(orders_df)}
    )

if __name__ == "__main__":
    with open_dagster_pipes():
        main()
```

Internal asset shim orchestrating the execution of external script

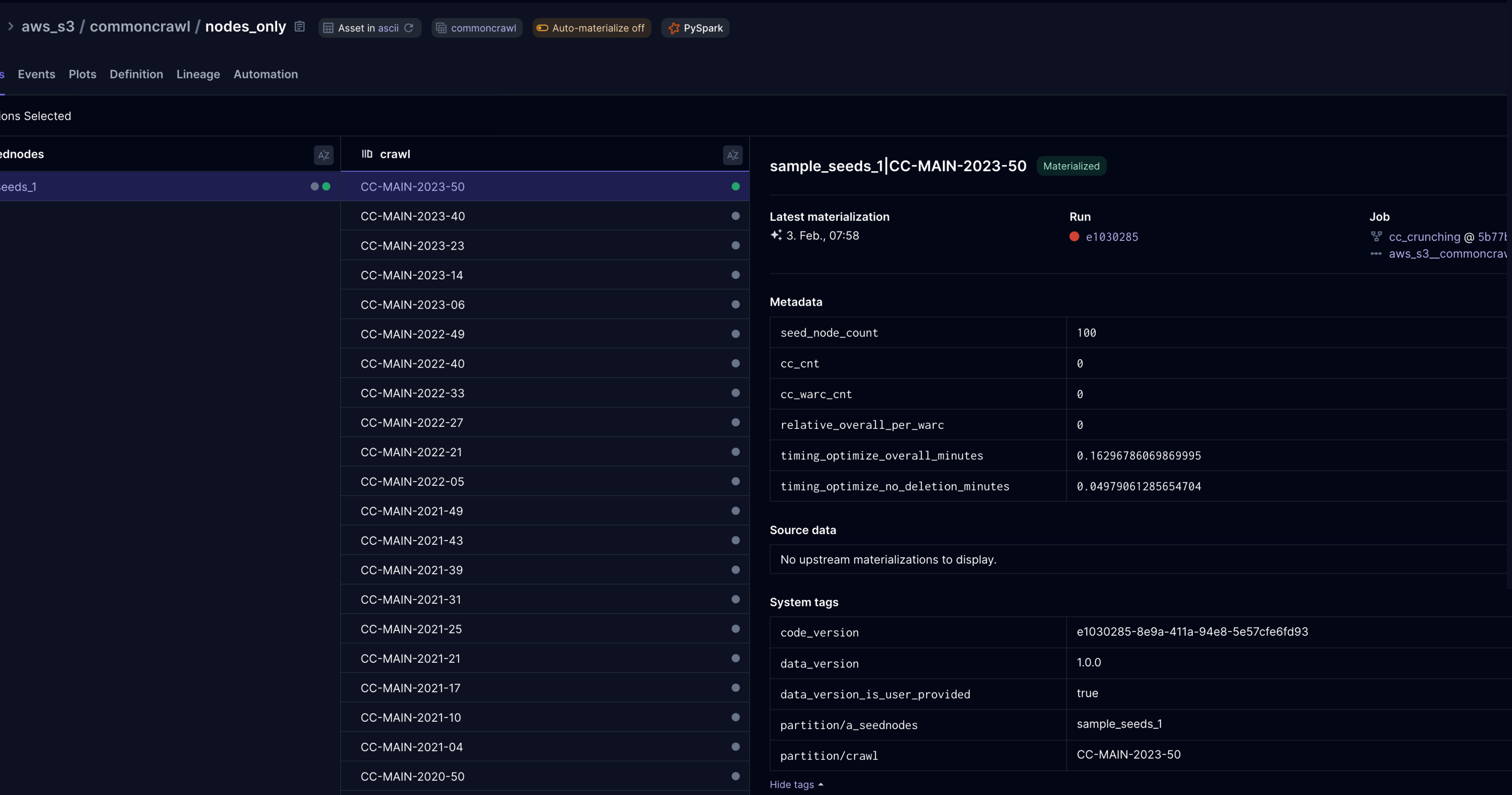


```python
@asset
def subprocess_asset(
    context: AssetExecutionContext, pipes_subprocess_client: PipesSubprocessClien
) -> MaterializeResult:
    cmd = [shutil.which("python"), file_relative_path(__file__, "external_code.py
    return pipes_subprocess_client.run(
        command=cmd,
        context=context,
        extras={"foo": "bar"},
        env={
            "MY_ENV_VAR_IN_SUBPROCESS": "my_value",
        },
    ).get_materialize_result()
```

Results


nodes_only
No description
● 1
○ 93
▲ 0
94 partitions
PySpark
edges
No description
● 1
○ 93
▲ 0
94 partitions
emr
graph
No description
● 1
○ 93
▲ 0
94 partitions
Databricks
graph_aggr
No description
● 1
○ 93
▲ 0
94 partitions
PySpark

Partitioned UI



aws_s3 / commoncrawl / nodes_only

ons Selected

ednodes	crawl
eeds_1	CC-MAIN-2023-50
	CC-MAIN-2023-40
	CC-MAIN-2023-23
	CC-MAIN-2023-14
	CC-MAIN-2023-06
	CC-MAIN-2022-49
	CC-MAIN-2022-40
	CC-MAIN-2022-33
	CC-MAIN-2022-27
	CC-MAIN-2022-21
	CC-MAIN-2022-05
	CC-MAIN-2021-49
	CC-MAIN-2021-43
	CC-MAIN-2021-39
	CC-MAIN-2021-31
	CC-MAIN-2021-25
	CC-MAIN-2021-21
	CC-MAIN-2021-17
	CC-MAIN-2021-10
	CC-MAIN-2021-04
	CC-MAIN-2020-50

sample_seeds_1|CC-MAIN-2023-50 Materialized

Latest materialization	Run	Job
3. Feb., 07:58	e1030285	cc_crunching @ 5b77t aws_s3__commoncraw

Metadata

seed_node_count	100
cc_cnt	0
cc_warc_cnt	0
relative_overall_per_warc	0
timing_optimize_overall_minutes	0.16296786069869995
timing_optimize_no_deletion_minutes	0.04979061285654704

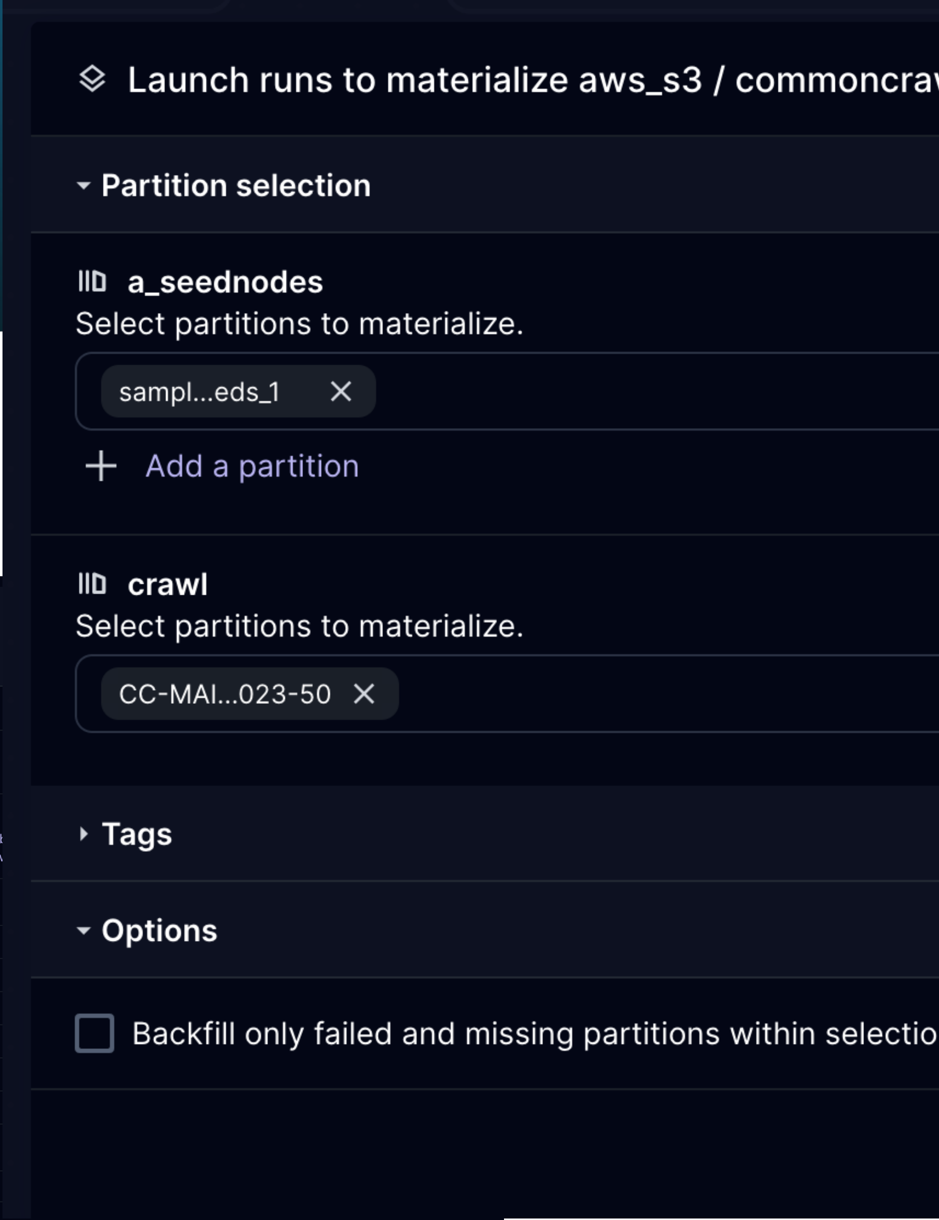
Source data

No upstream materializations to display.

System tags

code_version	e1030285-8e9a-411a-94e8-5e57cfe6fd93
data_version	1.0.0
data_version_is_user_provided	true
partition/a_seednodes	sample_seeds_1
partition/crawl	CC-MAIN-2023-50

Hide tags



Launch runs to materialize aws_s3 / commoncraw

Partition selection

a_seednodes

Select partitions to materialize.

sampl...eds_1 ✕

crawl

Select partitions to materialize.

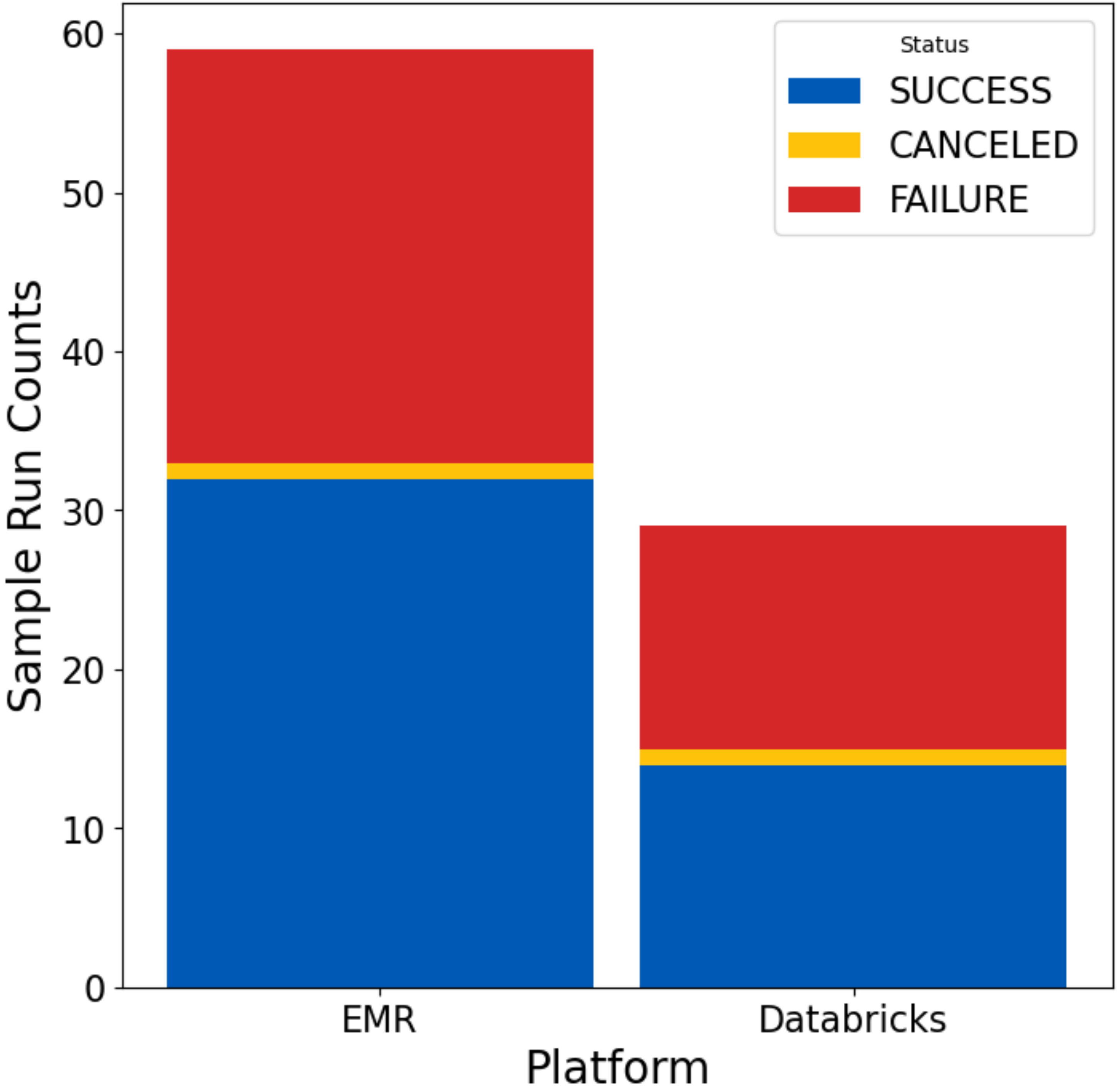
CC-MAI...023-50 ✕

Tags

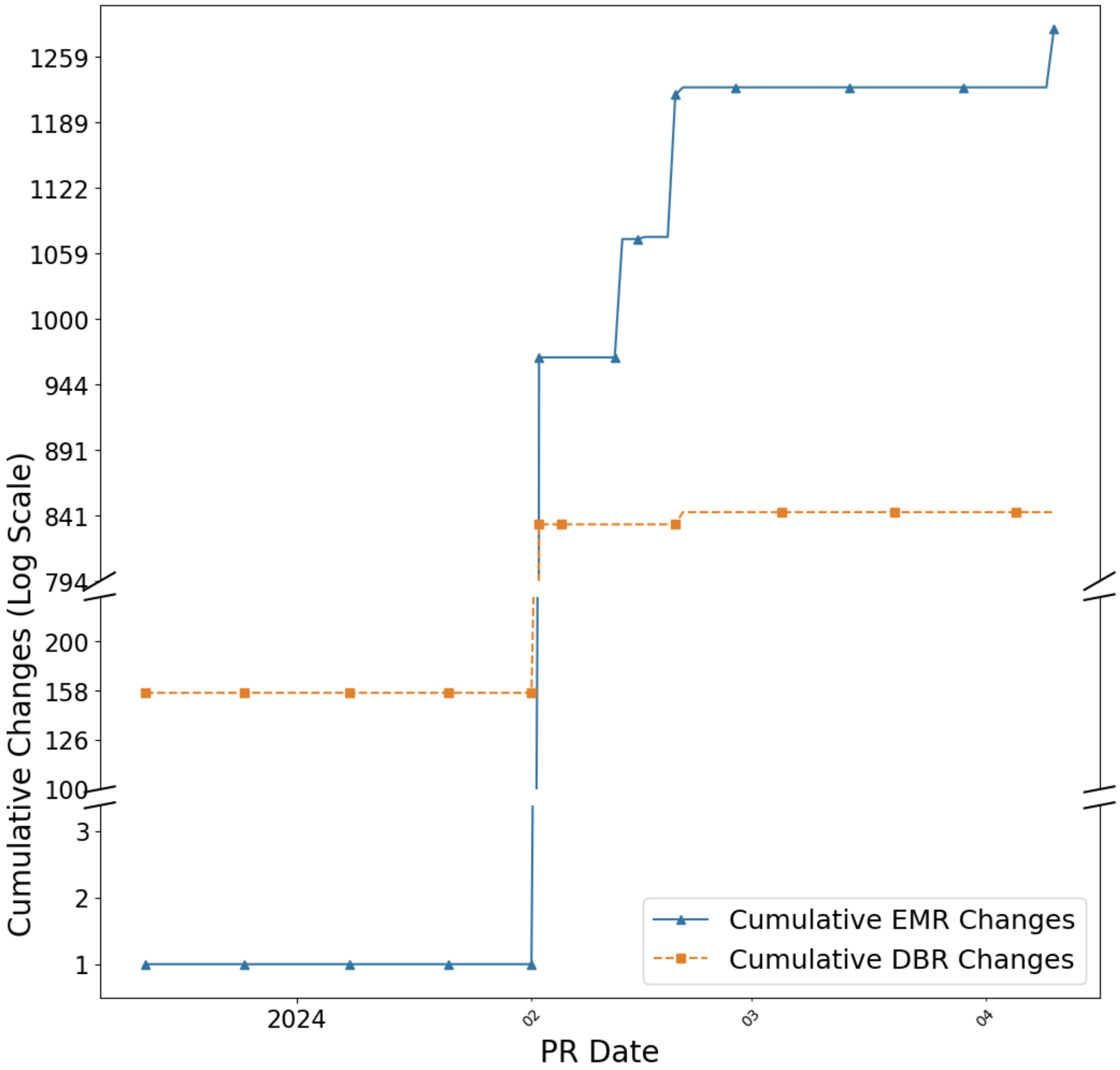
Options

□ Backfill only failed and missing partitions within selectio

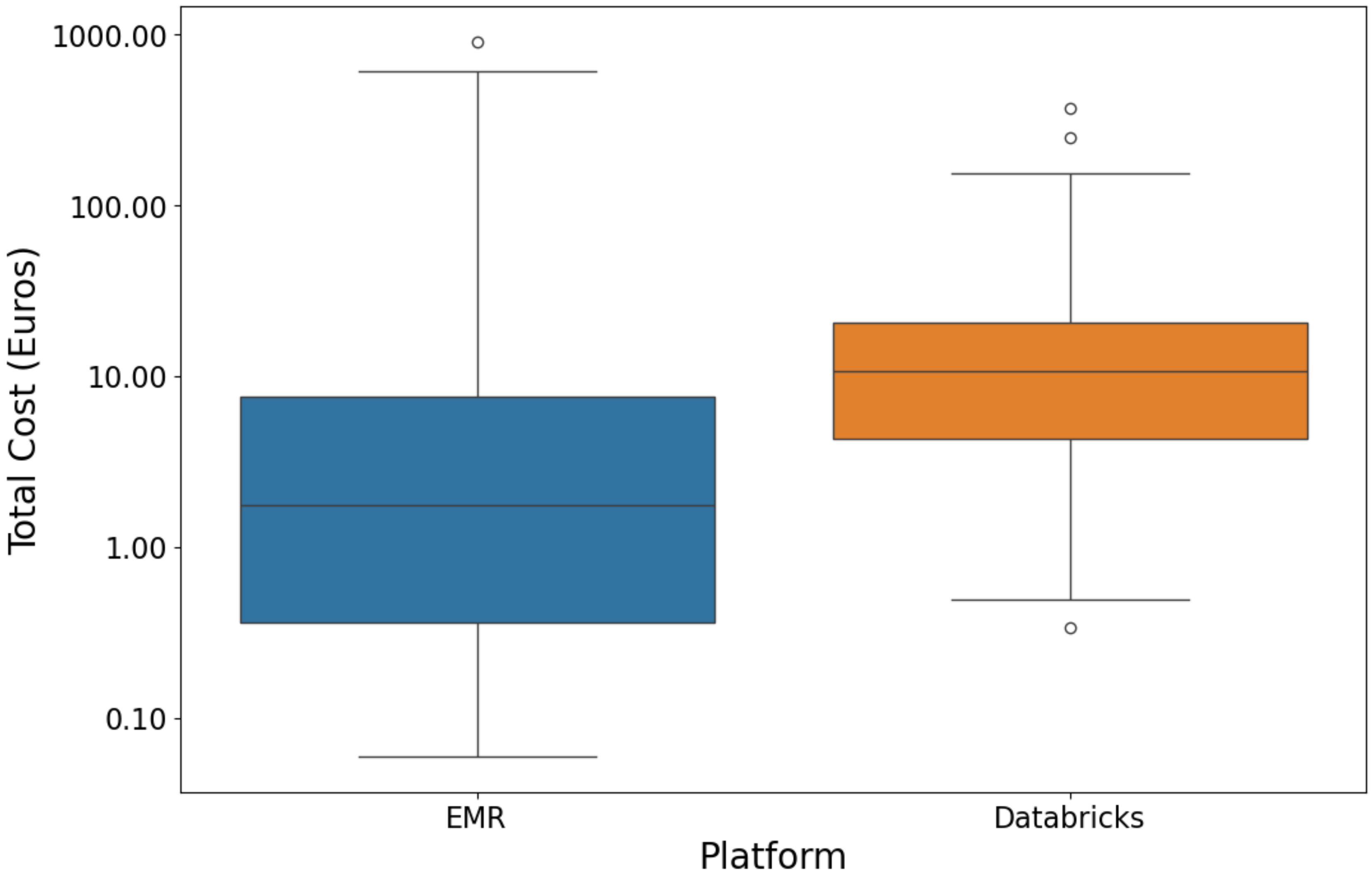
Implementation time of DBR is lower



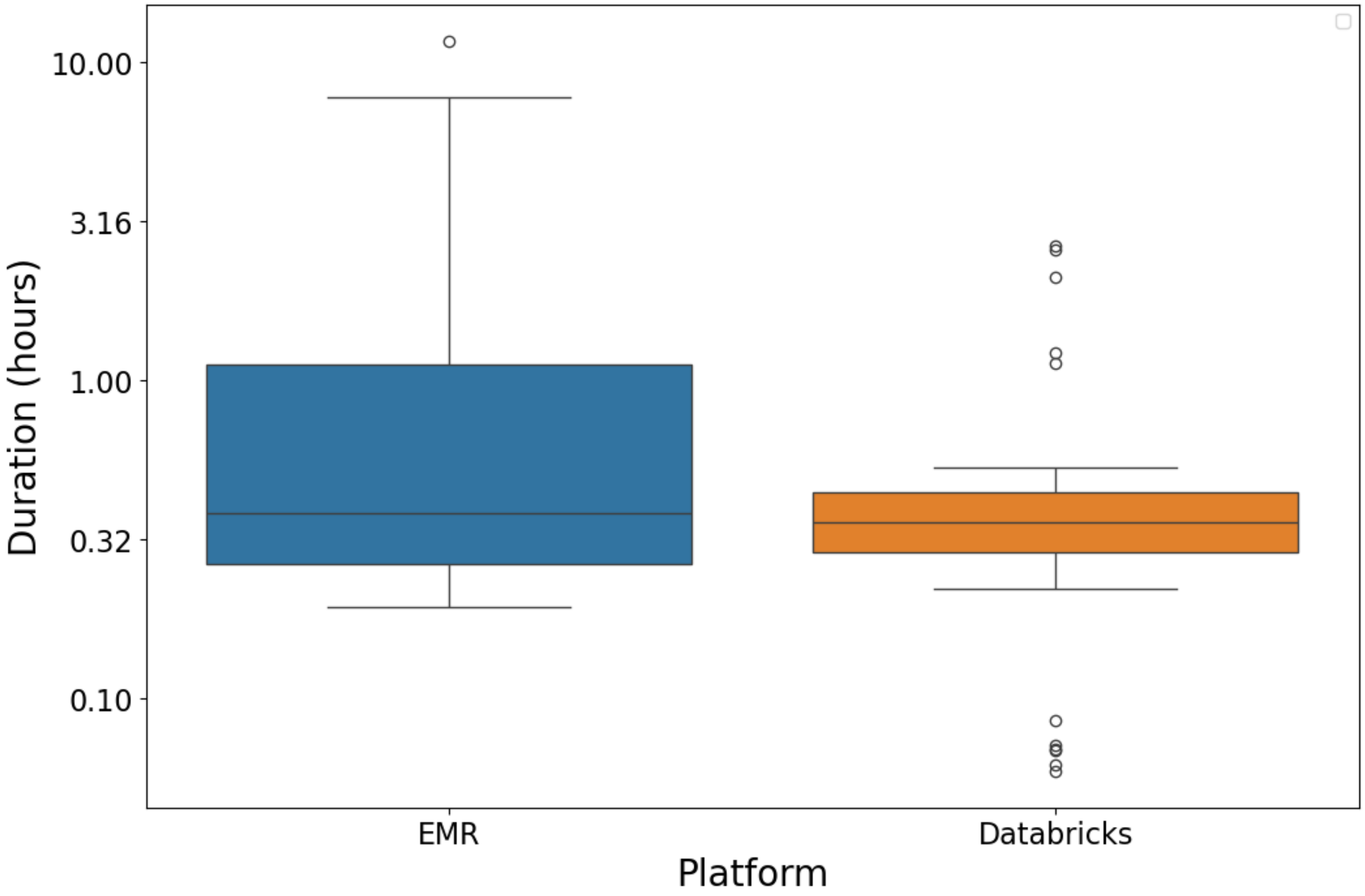
Implementation complexity of DBR is lower
more & more frequent commits for EMR integration



Median cost of DBR is higher than EMR



Variability of execution time of DBR is lower



Implementation learnings

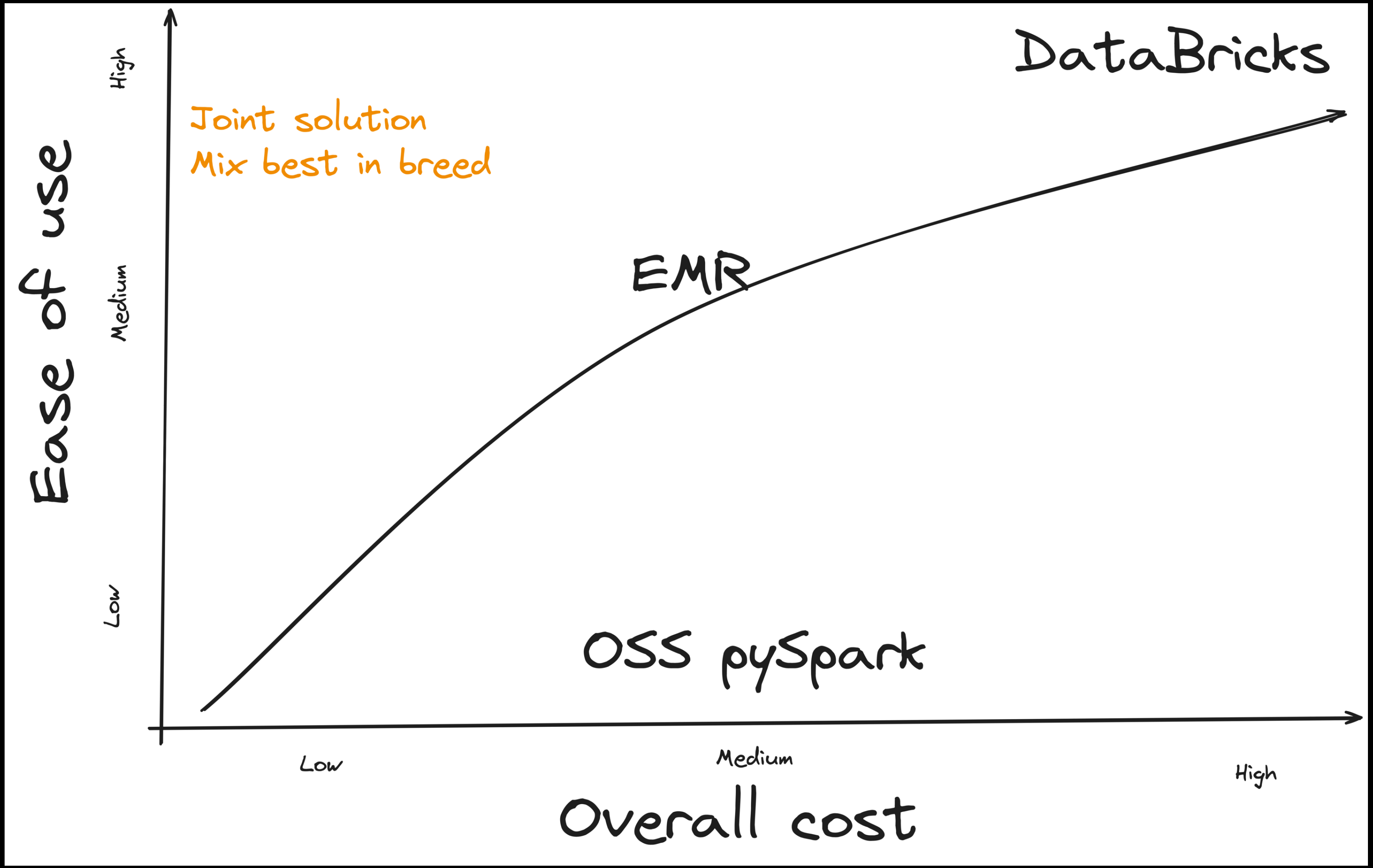
- Complexity of AWS EMR: Many low level details about AWS, spot instances, networking required (master on spot instance => 💥💥)
- Abstracting the PaaS requires deep understanding of their APIs

Tips

- maximizeResourceAllocation
- LZO
- Delta zorder on partition
- spark.databricks.delta.vacuum.parallelDelete.enabled=true

Summary

- Money saved – 43%
- Bring back software engineering best practices for data
- Flexibility
  - Data PaaS as a commodity
  - Take back control
  - Best in breed



COST EFFICIENCY FOR SPARK

Georg Heiler / Hernan Picatto

georgheiler.com/2024/05/02/cost-efficient-alternative-to-databricks-lock-in