PyData

Why Not Dagster?

PyData Vilnius #1

2023 -11-21

Tomas Peluritis

Data Engineer @ WIX

AKA Uncle Data

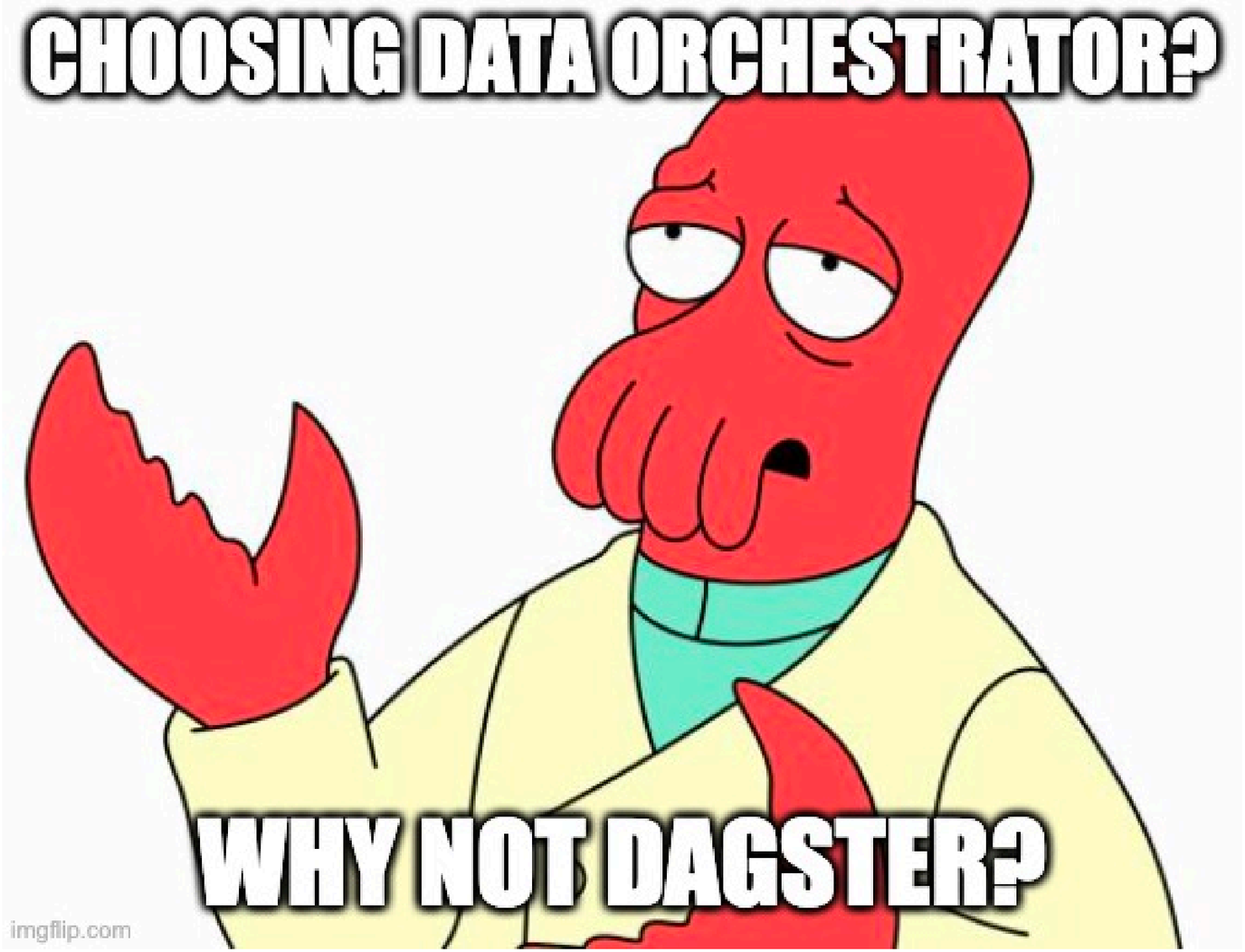
https://www.linkedin.com/in/tomaspeluritis/

https://podcasters.spotify.com/pod/show/duomenu-dede

https://uncledata.substack.com

“... Choose something that will hurt you”

by Dr. Venkat Subramaniam


CHOOSING DATA ORCHESTRATOR?
WHY NOT DAGSTER?
imgflip.com

What data orchestrator are you using in production?

You can see how people vote. Learn more

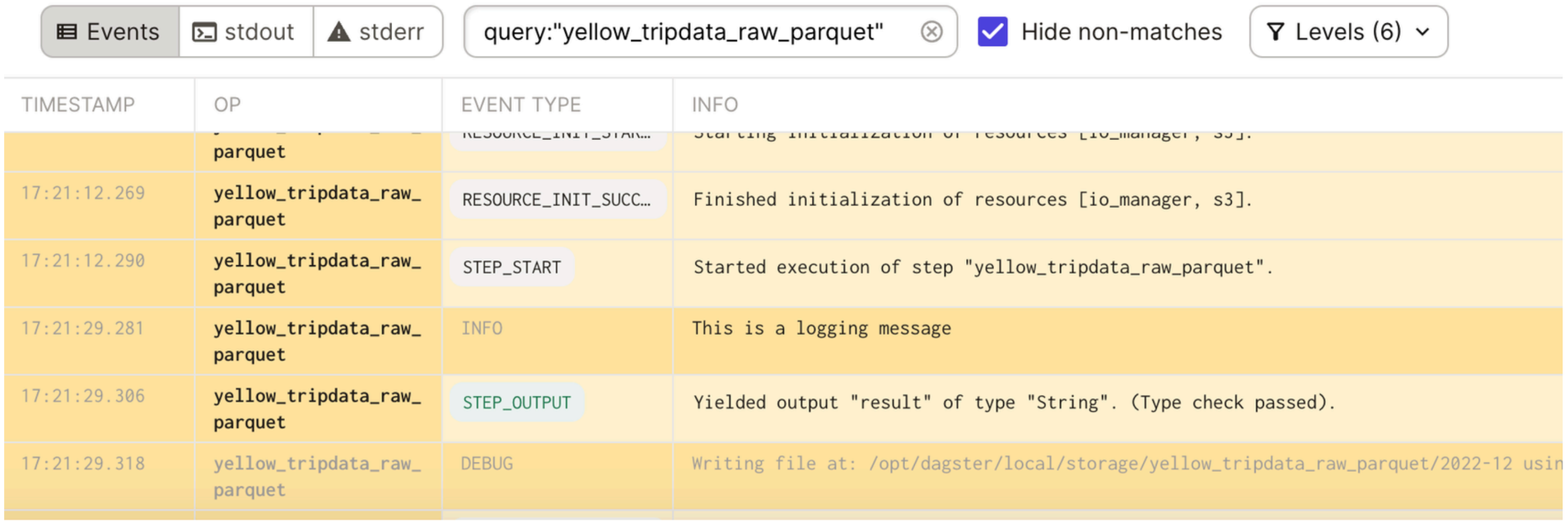
Airflow	71%
Dagster	4%
Prefect	4%
Other (elaborate in comments)	22%

106 votes • 2w left • Hide results

Development practices

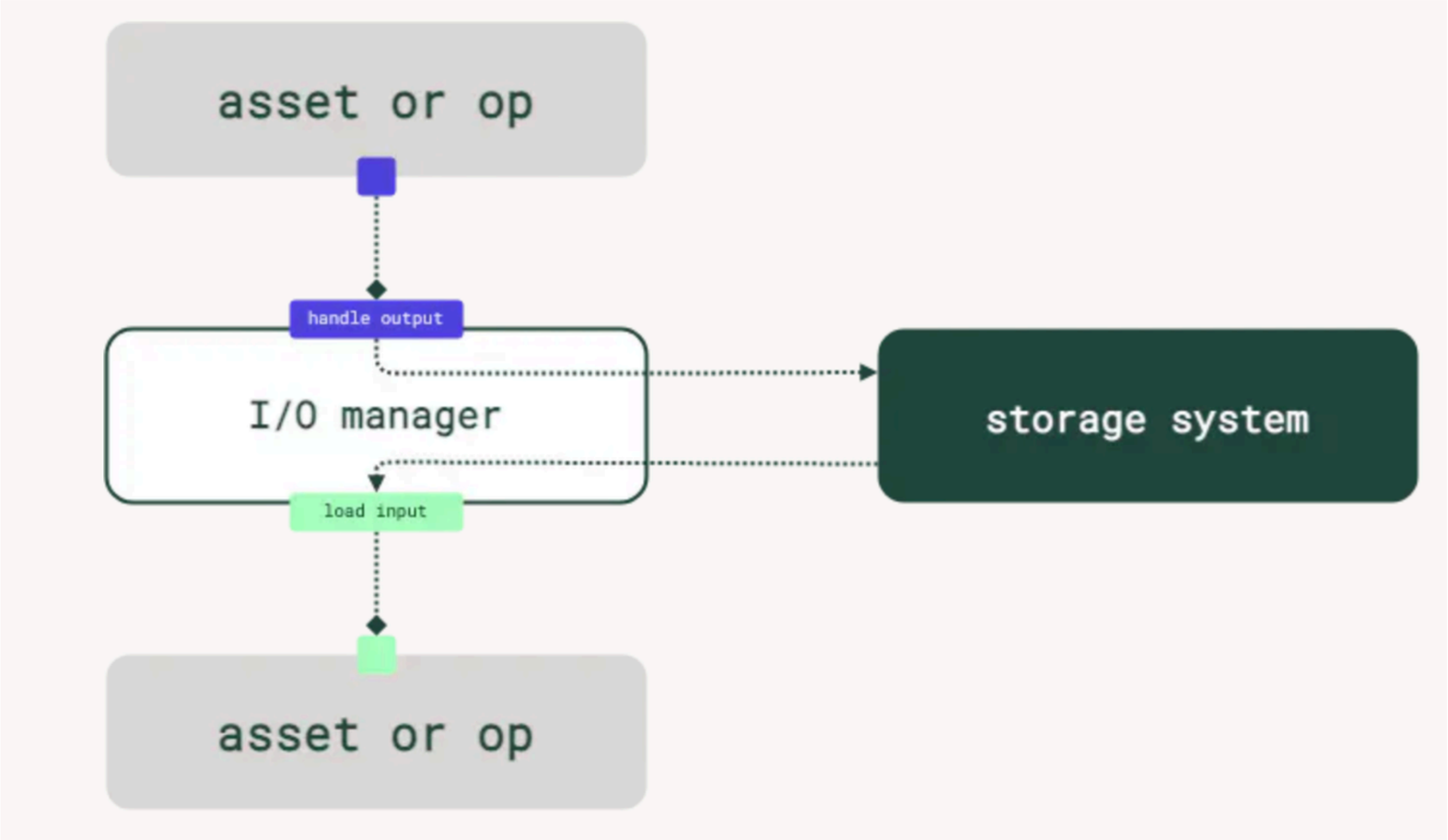
I ♥ PRINT

Use logging!


Events
stdout
stderr
query:"yellow_tripdata_raw_parquet"
Hide non-matches
Levels (6)
TIMESTAMP
OP
EVENT TYPE
INFO
parquet
17:21:12.269
yellow_tripdata_raw_parquet
RESOURCE_INIT_SUCC…
Finished initialization of resources [io_manager, s3].
17:21:12.290
yellow_tripdata_raw_parquet
STEP_START
Started execution of step "yellow_tripdata_raw_parquet".
17:21:29.281
yellow_tripdata_raw_parquet
INFO
This is a logging message
17:21:29.306
yellow_tripdata_raw_parquet
STEP_OUTPUT
Yielded output "result" of type "String". (Type check passed).
17:21:29.318
yellow_tripdata_raw_parquet
DEBUG
Writing file at: /opt/dagster/local/storage/yellow_tripdata_raw_parquet/2022-12 usin


I have a complex object I need to serialize
Panik
Python has multiple libraries that can help me
Kalm
Now I have to deserialize it back and ensure data integrity across different systems.
Panik
imgflip.com

More Flexibility



Mess in the repo

Build as a module

```
.
└── dagster_university
    ├── README.md
    ├── dagster_cloud.yaml
    ├── dagster_university
    │   ├── __init__.py
    │   ├── assets
    │   │   ├── __init__.py
    │   │   ├── constants.py
    │   │   ├── metrics.py
    │   │   └── trips.py
    │   ├── jobs
    │   │   └── __init__.py
    │   ├── partitions
    │   │   └── __init__.py
    │   ├── resources
    │   │   └── __init__.py
    │   ├── schedules
    │   │   └── __init__.py
    │   └── sensors
    │       └── __init__.py
    ├── dagster_university_tests
    │   ├── __init__.py
    │   └── test_assets.py
    ├── data
    │   ├── outputs
    │   ├── raw
    │   ├── requests
    │   │   └── README.md
    │   └── staging
    ├── pyproject.toml
    ├── setup.cfg
    └── setup.py
```


YOU GET AN ISOLATED ENVIRONMENT
EVERYONE GETS AN ISOLATED ENVIRONMENT
imgflip.com

Creating isolated environments

```
load_from:
  - python_file:
      relative_path: path/to/dataengineering_spark_team.py
      location_name: dataengineering_spark_team_py_38_virtual_env
      executable_path: venvs/path/to/de_team/bin/python
  - python_file:
      relative_path: path/to/team_code_location.py
      location_name: ml_team_py_36_virtual_env
      executable_path: venvs/path/to/ml_tensorflow/bin/python
  - grpc_server:
      host: MY_GRPC_SERVER
      port: 4266
      location_name: "my_grpc_server"
```


Unit testing your pipeline
Not Unit testing your pipeline
DATA ENGINEERS
imgflip.com
JAKE-CLARK.TUMBLR

Testing seems more natural

```python
from dagster import asset, materialize_to_memory


@asset
def data_source():
    return get_data_from_source()


@asset
def structured_data(data_source):
    return extract_structured_data(data_source)


# An example unit test using materialize_to_memory
def test_data_assets():
    result = materialize_to_memory([data_source, structured_data])
    assert result.success
    # Materialized objects can be accessed in terms of the underlying op
    materialized_data = result.output_for_node("structured_data")
```

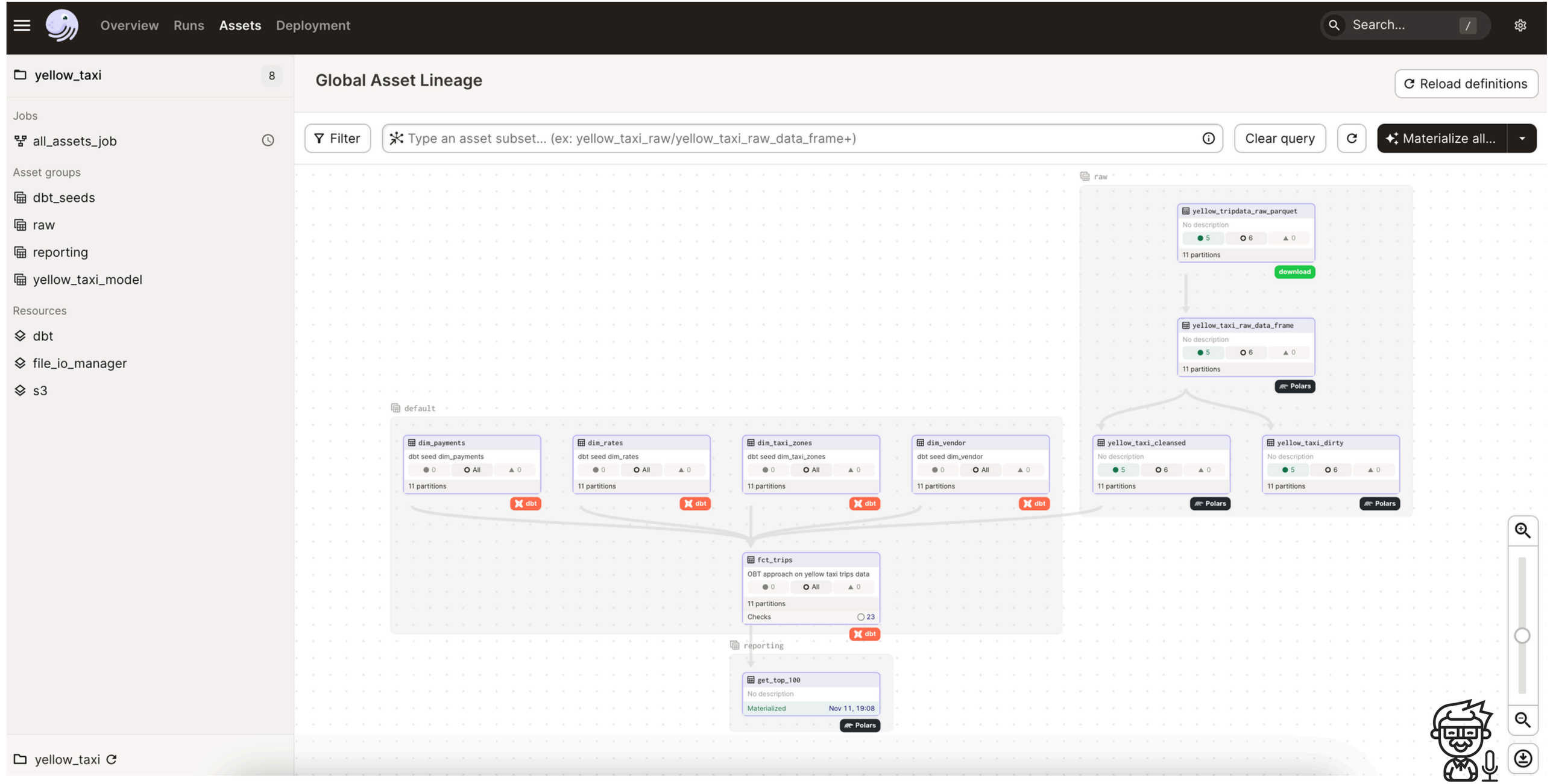
“... Choose something that will hurt you”

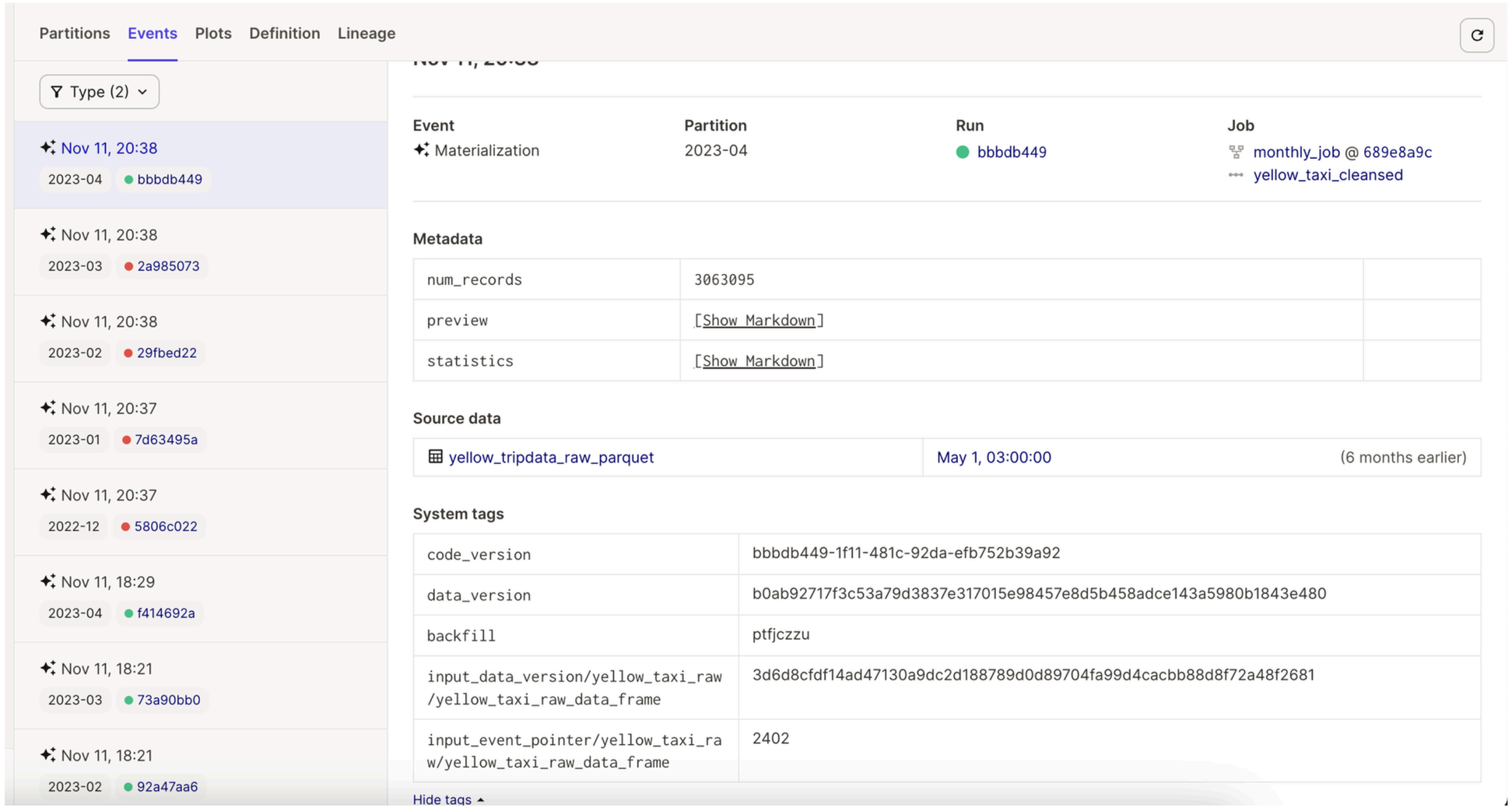
by Dr. Venkat Subramaniam

Paradigm Shift

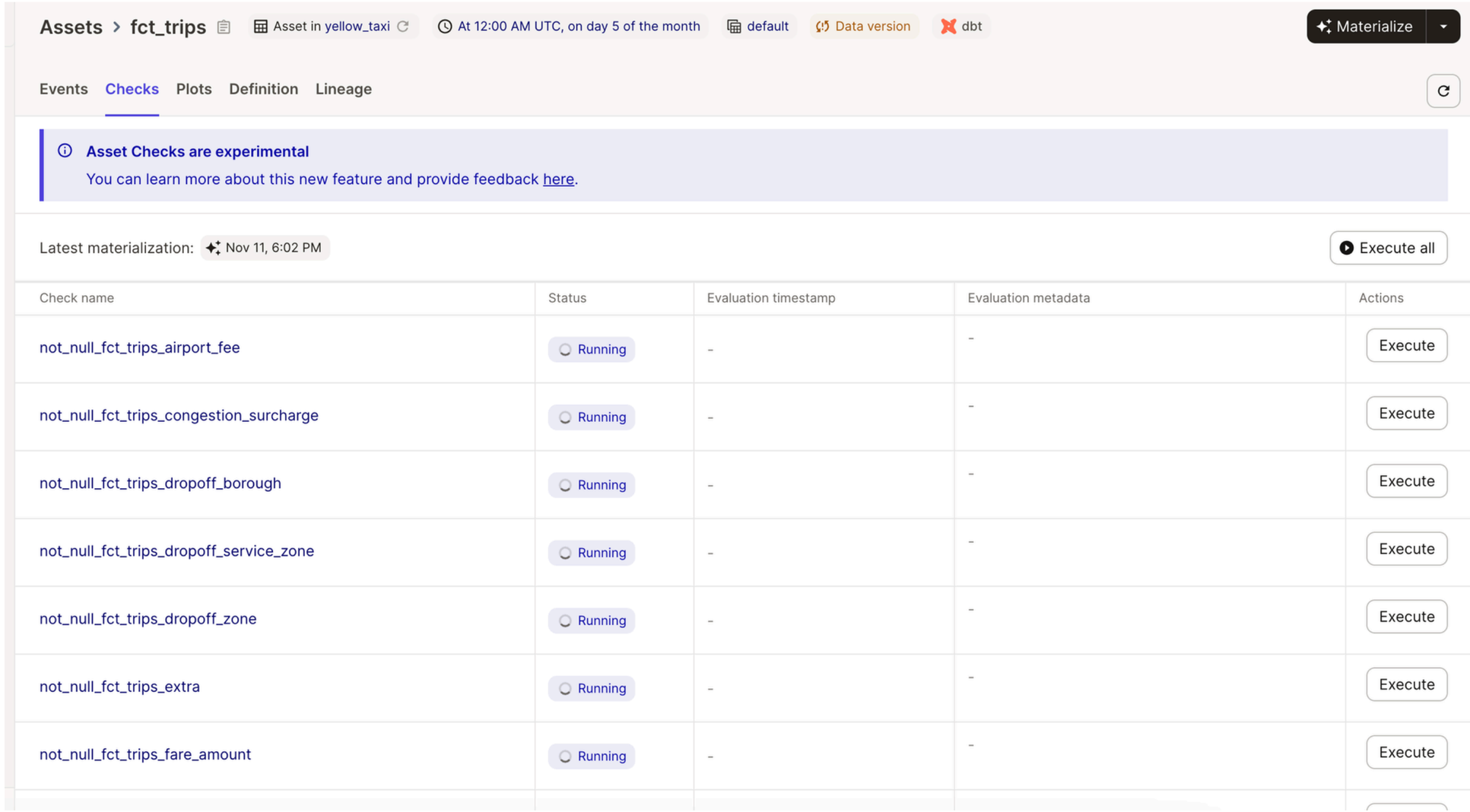
Asset focused approach


WHERE DEMO?
imgflip.com

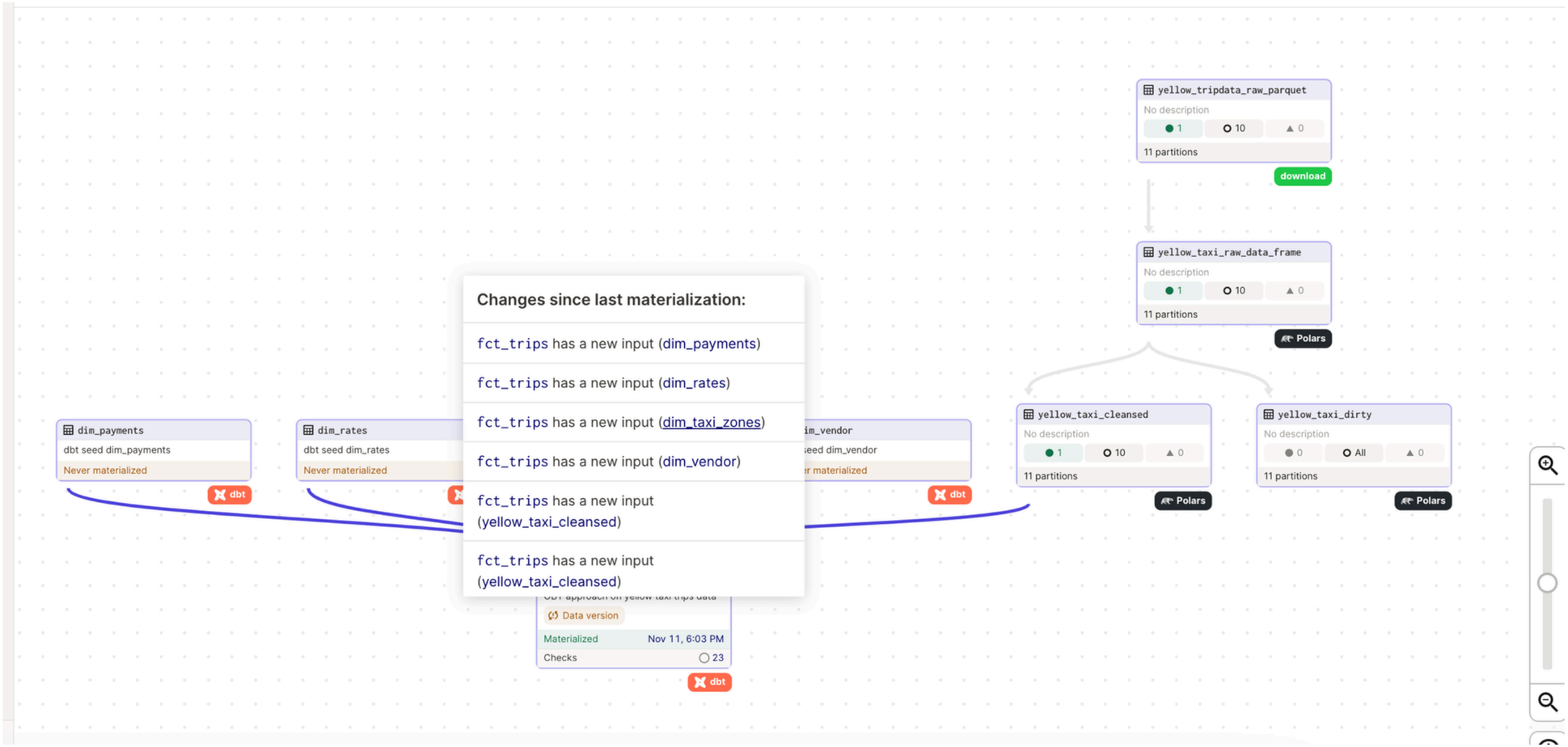

Overview
Runs
Assets
Deployment
Search...
/
yellow_taxi
8
Jobs
all_assets_job
Asset groups
dbt_seeds
raw
reporting
yellow_taxi_model
Resources
dbt
file_io_manager
s3
yellow_taxi
Global Asset Lineage
Reload definitions
Filter
Type an asset subset... (ex: yellow_taxi_raw/yellow_taxi_raw_data_frame+)
Clear query
Materialize all...
raw
yellow_tripdata_raw_parquet
No description
11 partitions
download
yellow_taxi_raw_data_frame
No description
11 partitions
Polars
default
dim_payments
dbt seed dim_payments
11 partitions
dbt
dim_rates
dbt seed dim_rates
11 partitions
dbt
dim_taxi_zones
dbt seed dim_taxi_zones
11 partitions
dbt
dim_vendor
dbt seed dim_vendor
11 partitions
dbt
yellow_taxi_cleansed
No description
11 partitions
Polars
yellow_taxi_dirty
No description
11 partitions
Polars
fct_trips
OBT approach on yellow taxi trips data
11 partitions
Checks
23
dbt
reporting
get_top_100
No description
Materialized
Nov 11, 19:08
Polars


Partitions
Events
Plots
Definition
Lineage
Type (2)
Nov 11, 20:38
2023-04
bbbdb449
Nov 11, 20:38
2023-03
2a985073
Nov 11, 20:38
2023-02
29fbed22
Nov 11, 20:37
2023-01
7d63495a
Nov 11, 20:37
2022-12
5806c022
Nov 11, 18:29
2023-04
f414692a
Nov 11, 18:21
2023-03
73a90bb0
Nov 11, 18:21
2023-02
92a47aa6
Event
Materialization
Partition
2023-04
Run
bbbdb449
Job
monthly_job @ 689e8a9c
yellow_taxi_cleansed
Metadata
num_records
3063095
preview
[Show Markdown]
statistics
[Show Markdown]
Source data
yellow_tripdata_raw_parquet
May 1, 03:00:00
(6 months earlier)
System tags
code_version
bbbdb449-1f11-481c-92da-efb752b39a92
data_version
b0ab92717f3c53a79d3837e317015e98457e8d5b458adce143a5980b1843e480
backfill
ptfjczzu
input_data_version/yellow_taxi_raw/yellow_taxi_raw_data_frame
3d6d8cfdf14ad47130a9dc2d188789d0d89704fa99d4cacbb88d8f72a48f2681
input_event_pointer/yellow_taxi_raw/yellow_taxi_raw_data_frame
2402
Hide tags

Asset Checks



Data change notifications



But....

- **Not many tutorials**
- **Hard to grasp some of the definitions**
- **Learning curve is way steeper**
- **All other things I haven’t found out yet**

Dagster University

https://courses.dagster.io/courses/dagster-essentials


QUESTIONS