

Substring - The Data Company

Von DevOps zu MLOps

Dataholics Webinar, 17.10.2023

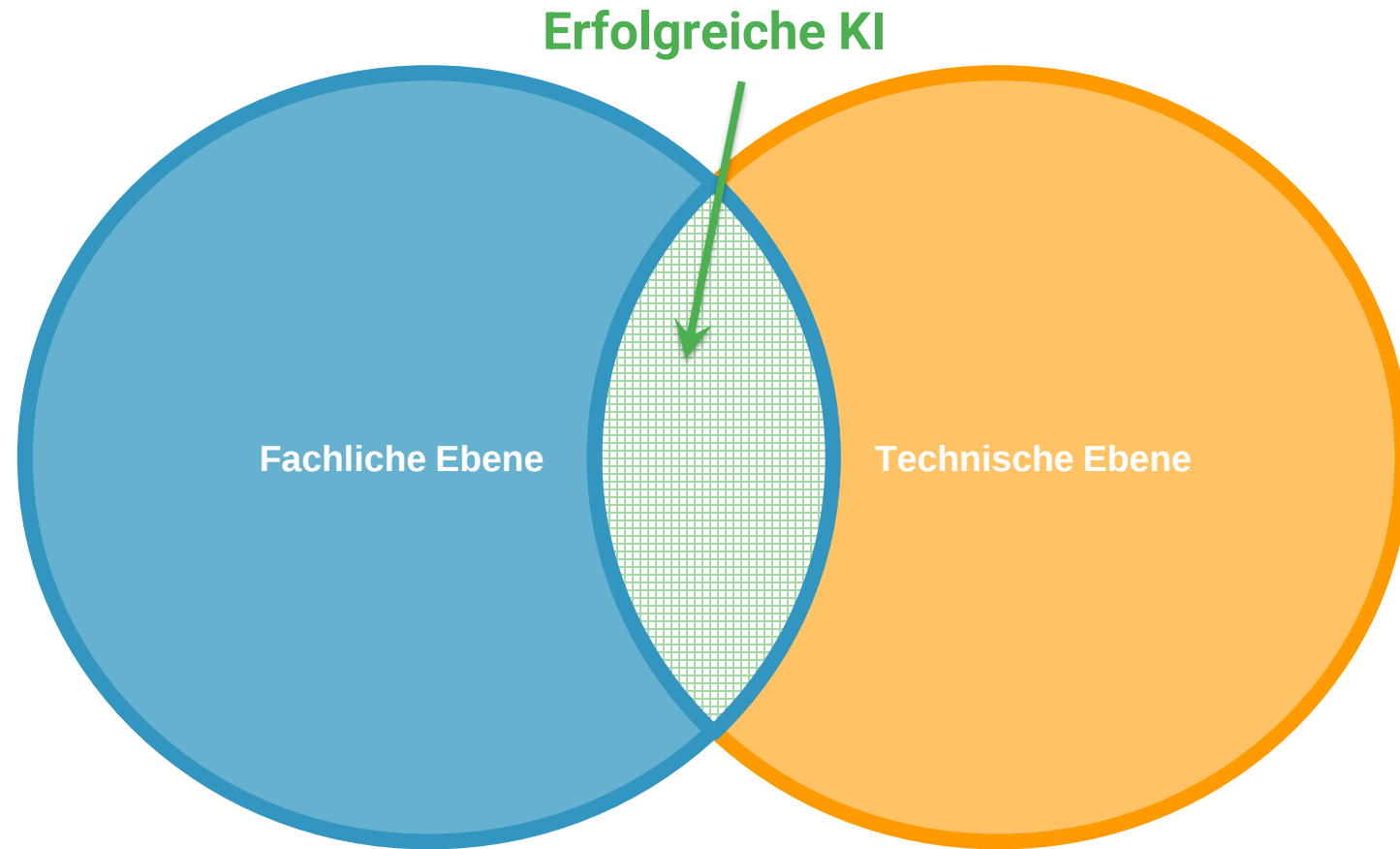


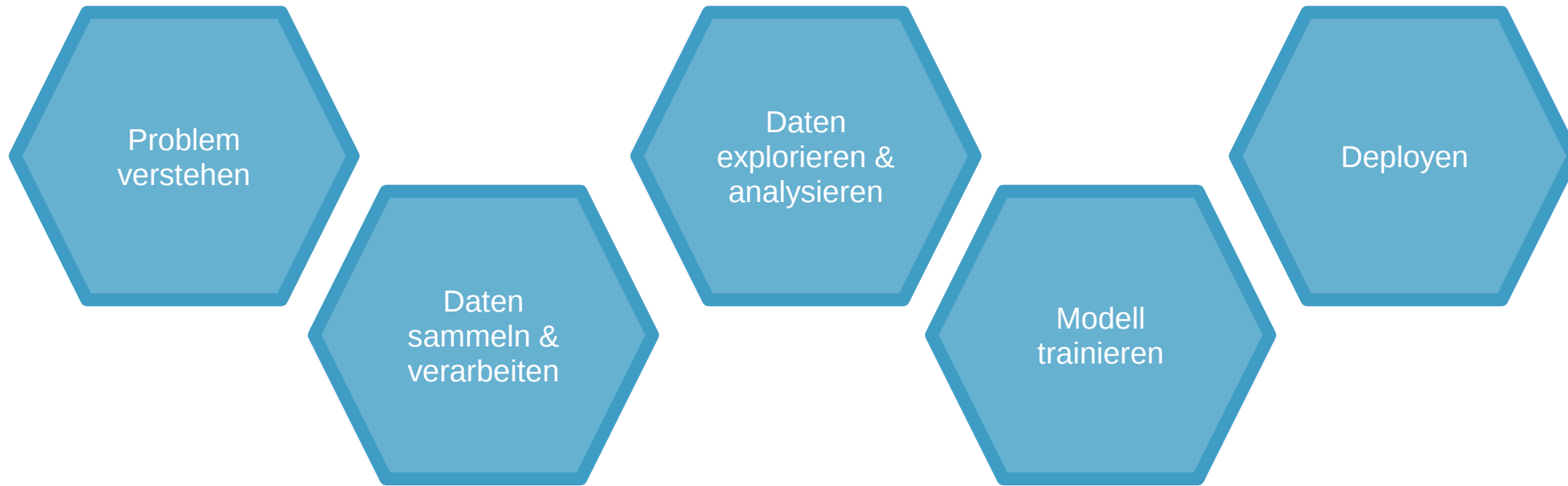
Nino Müller
Partner & Lead Technology
@ Substring
n.mueller@substring.ch

SO WHAT?

Was sind konkrete Anwendungsfälle von KI in Unternehmen ?

- LLM (Verbesserung von Kundeninteraktionen über textbasierte Dialogsysteme)
- Spracherkennung/-verarbeitung / Voice Assistants (Hands-free Interaktionen)
- Textanalyse (z.B. Spam-Detection oder Sentiment Analysis)
- Prediction (Predictive Maintenance in Manufacturing)
- Outlier detection (Erkennung von Abweichungen)
- Bilderkennung / Klassifikation (Identitätskontrolle)
- Deep Learning (Autonomes Fahren oder automatisches Trading von Währungen)
- Scoring (Risikoeinschätzung für eine Versicherung)

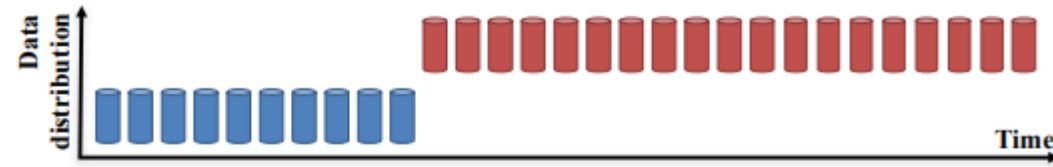




und dann...?

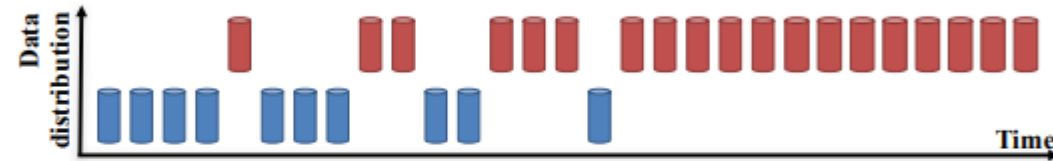
Sudden Drift:

A new concept occurs within a short time.



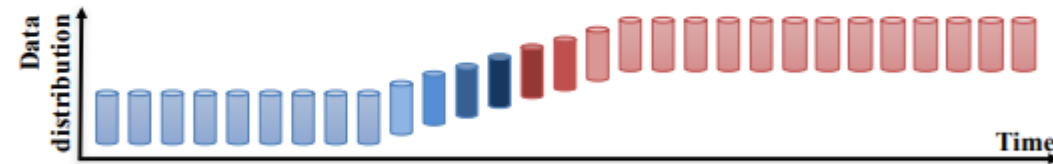
Gradual Drift:

A new concept gradually replaces an old one over a period of time.



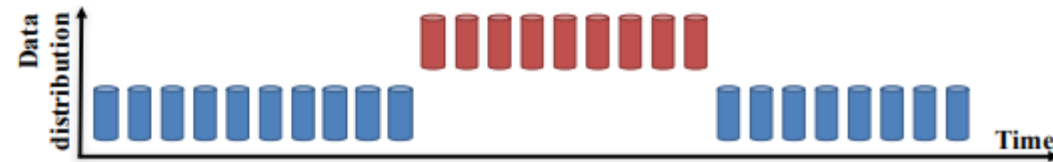
Incremental Drift:

An old concept incrementally changes to a new concept over a period of time.

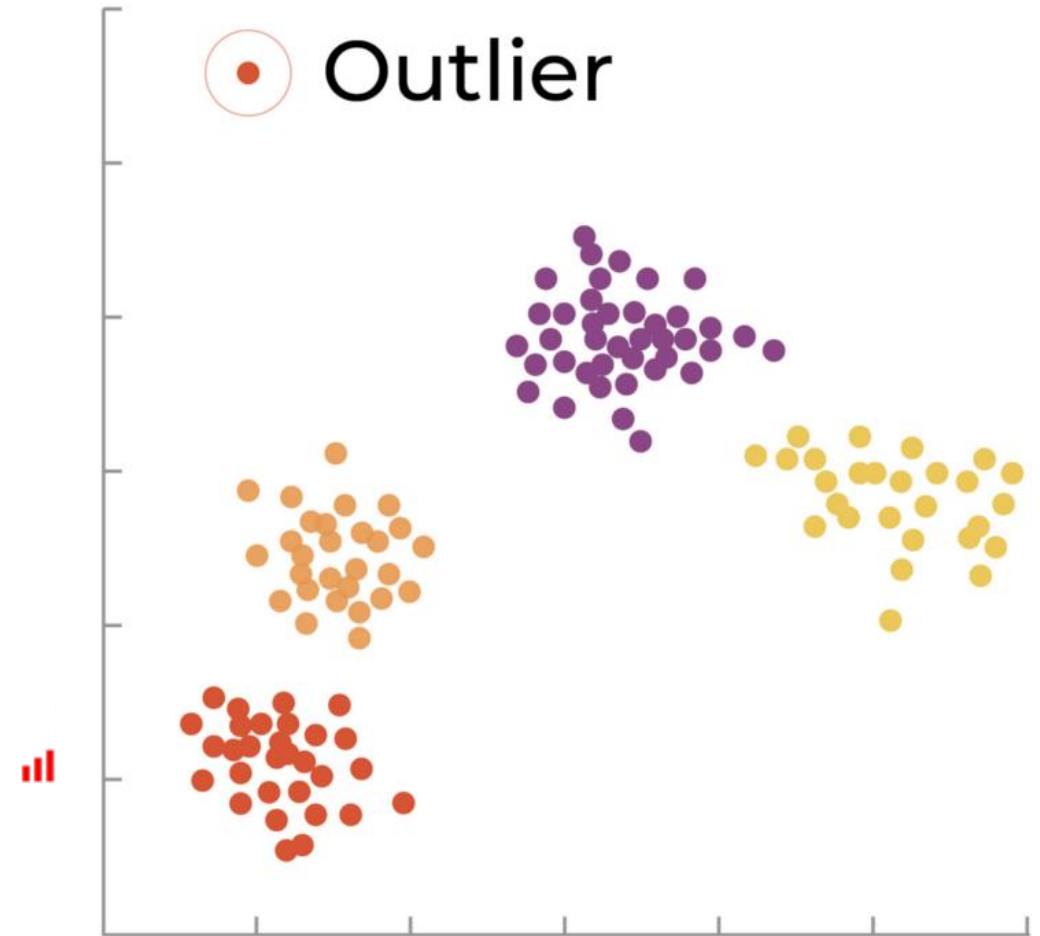
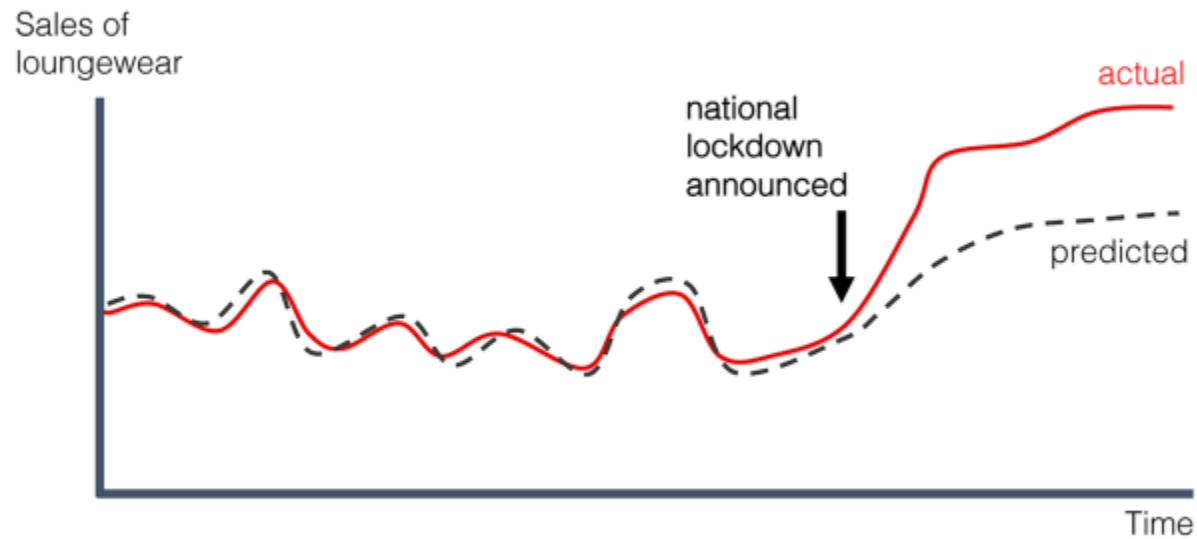


Reoccurring Concepts:

An old concept may reoccur after some time.

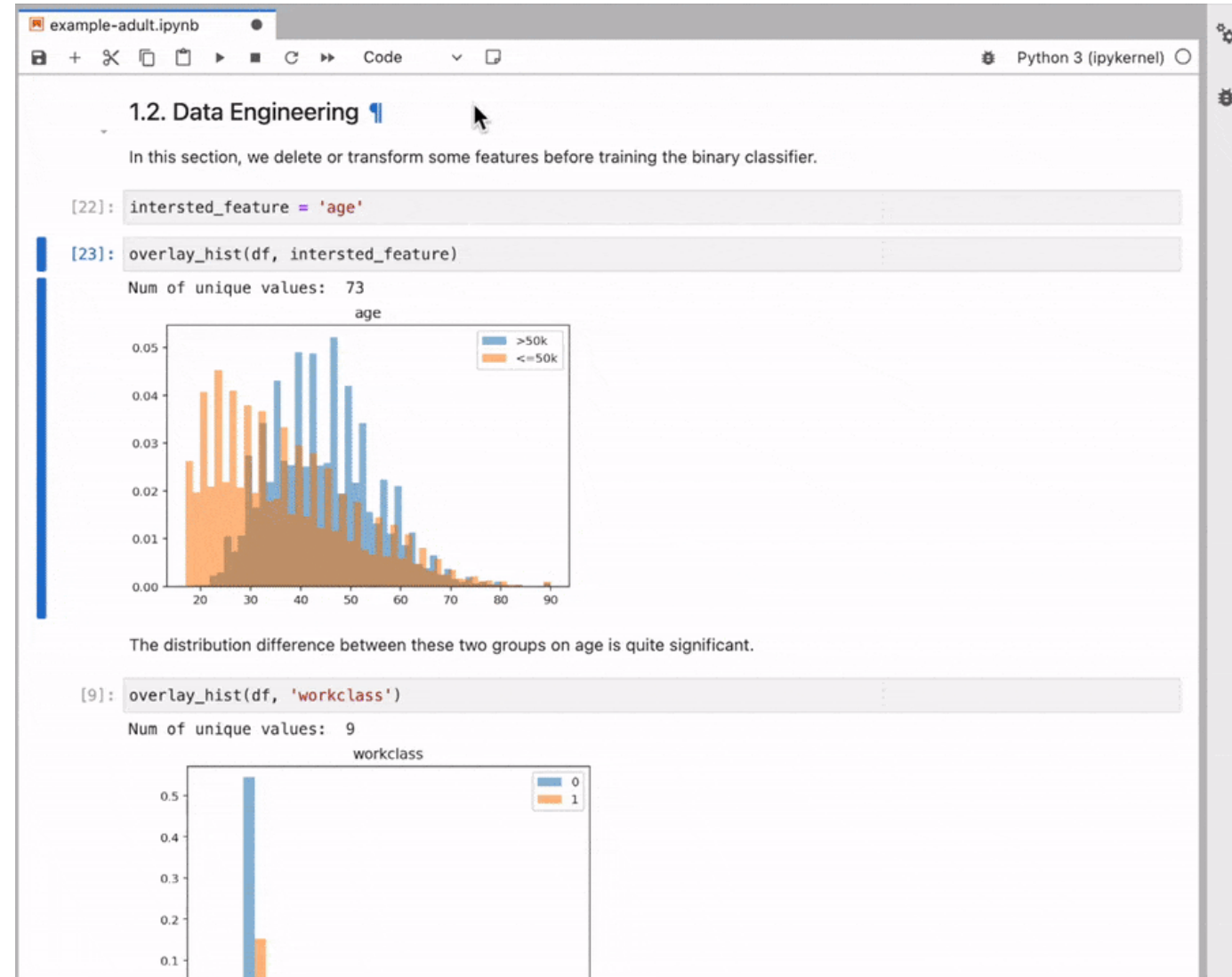


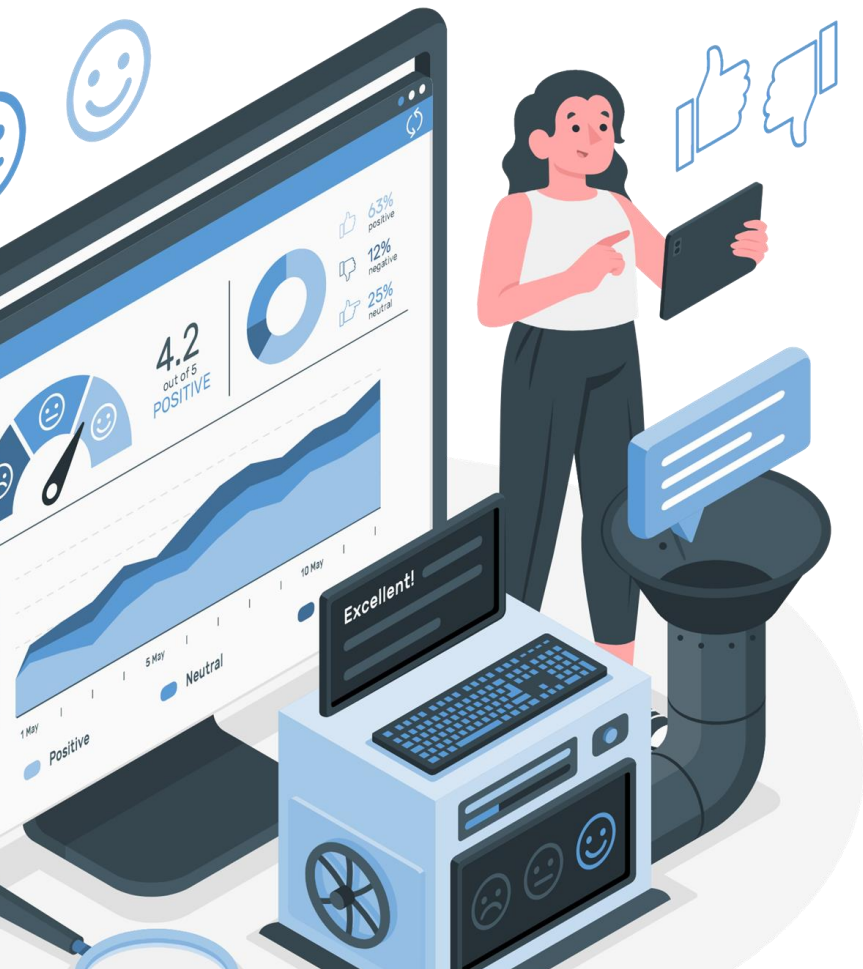
Model monitoring



Der Übeltäter

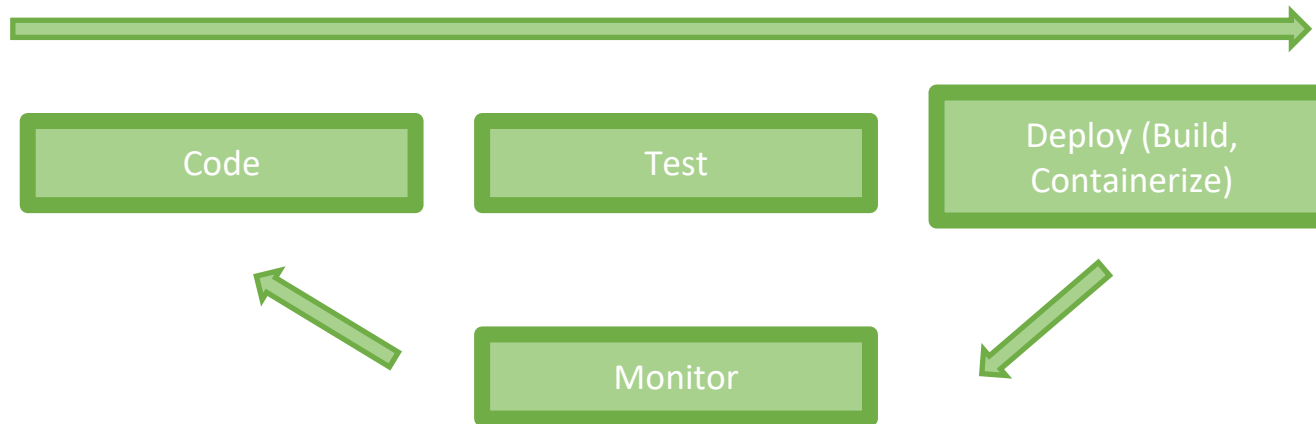
KI Prototyp



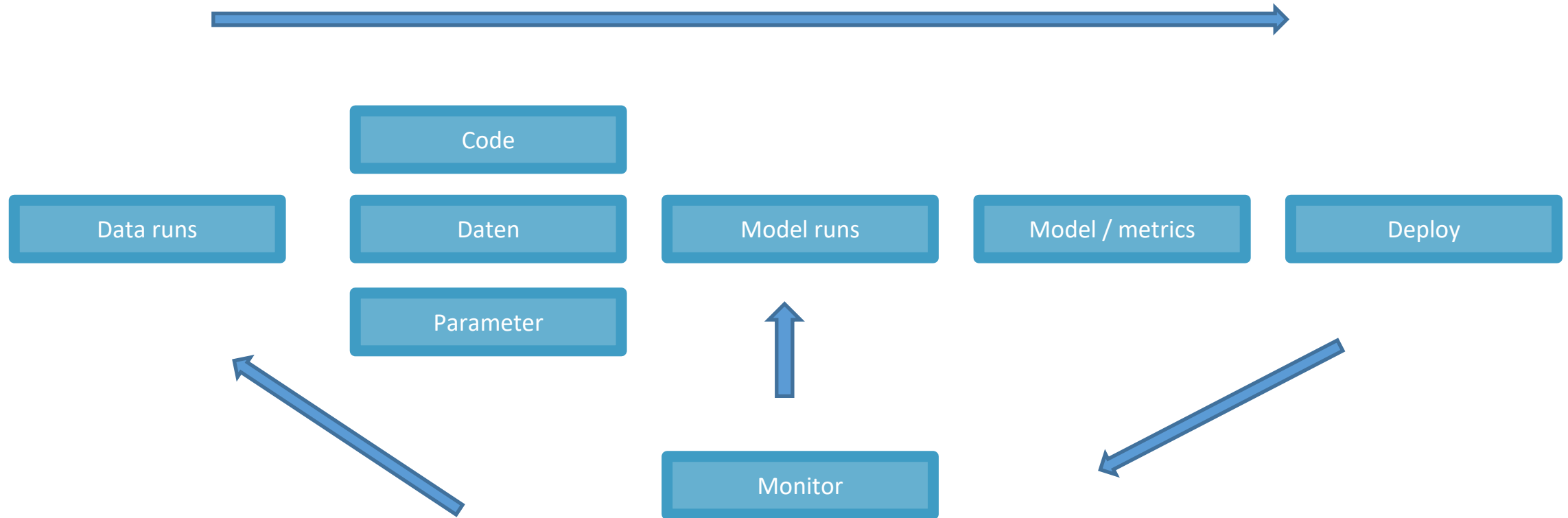


- **Nachvollziehbare Prozesse & Pipelines**
- **Reproduzierbare Modelle**
- **Kontinuierliches Deployment**
- **Kollaborative Prozesse**
- **Data-Driven**

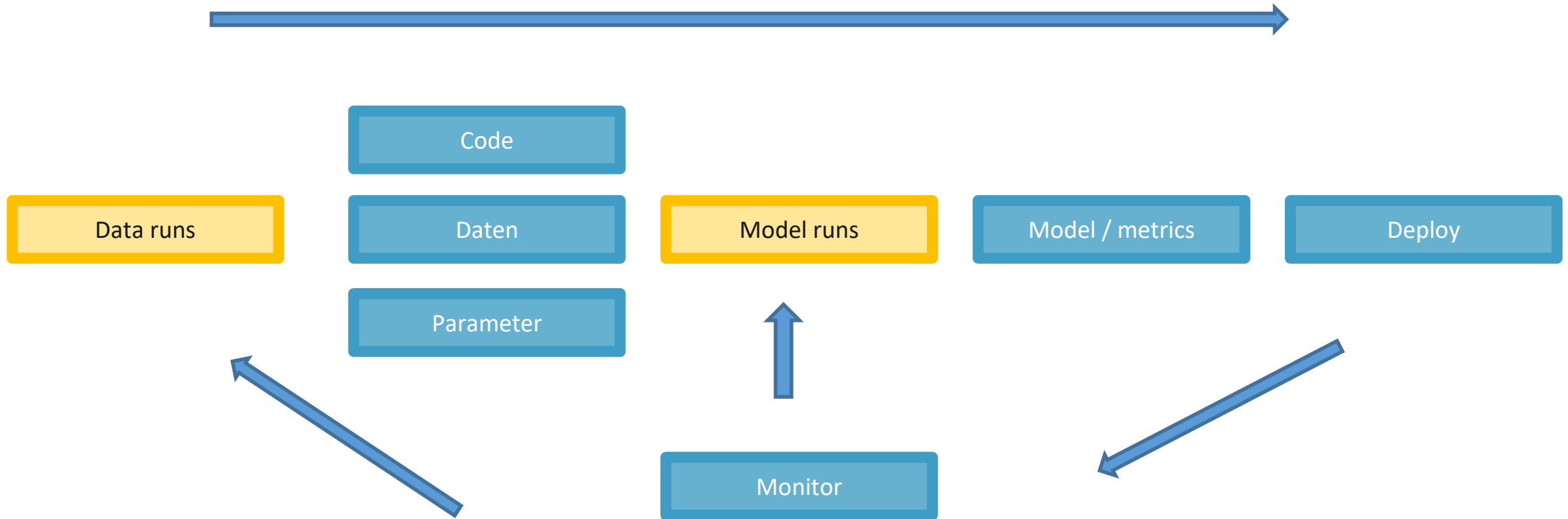
DevOps Prozess



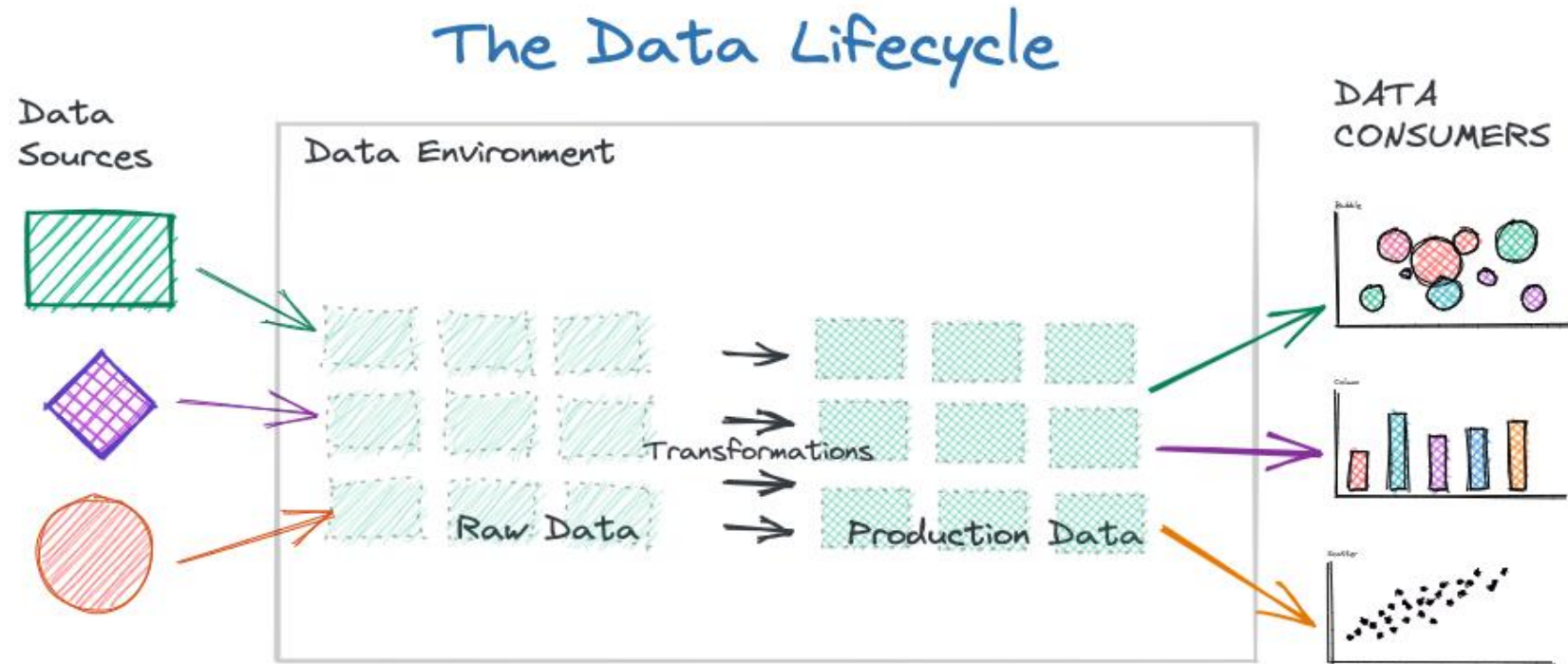
MLOps Prozess



MLOps Prozess



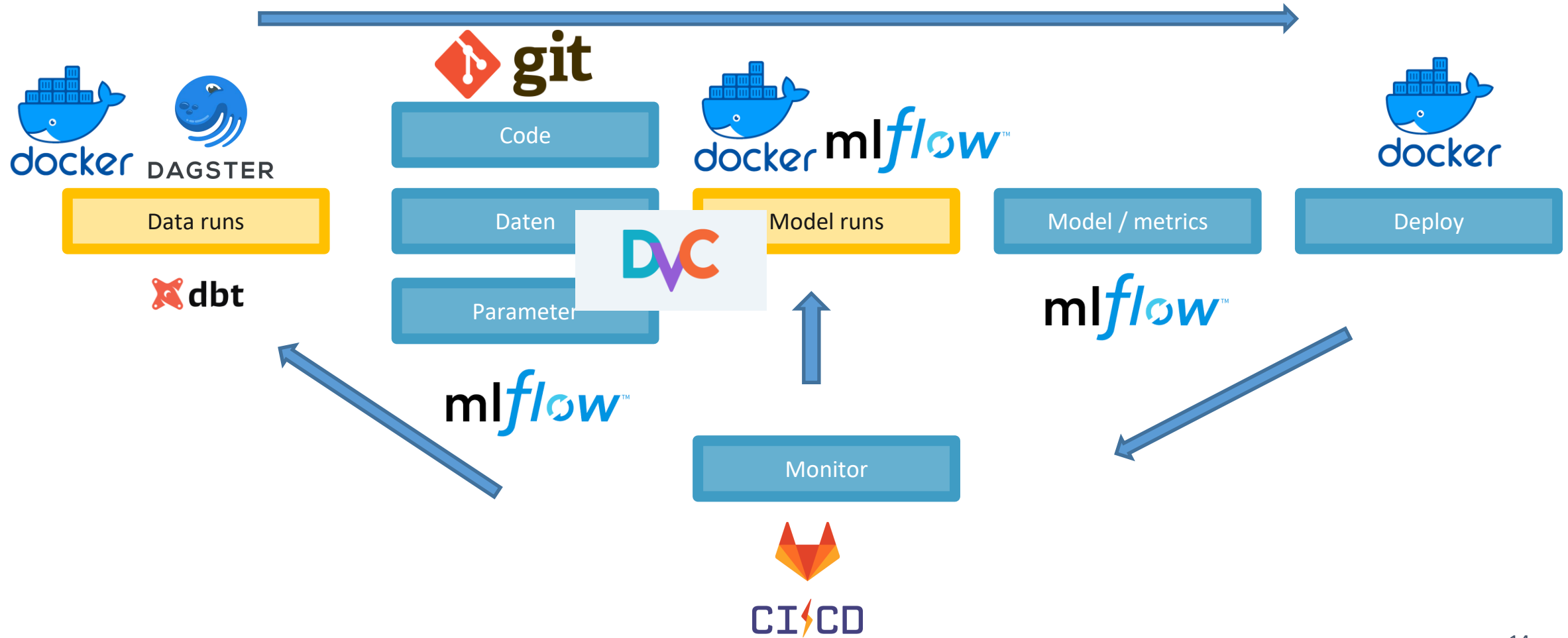
Was ist ein Data Run?



Was ist ein Model Run?

- **Input Parameter / Features**
- **Algorithmus**
- **Datengrundlage**
- **Hyperparameter**
- **Artifacts**
- **Metrics**

MLOps Prozess – OSS Stack

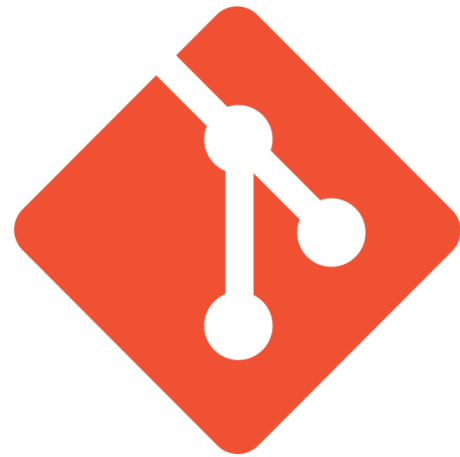


Data Runs

Was soll man speichern?

- **Data Runs Meta**
- **Features**
- **Datentransformationen**
- **Datenversion**
- **Data Annotations**

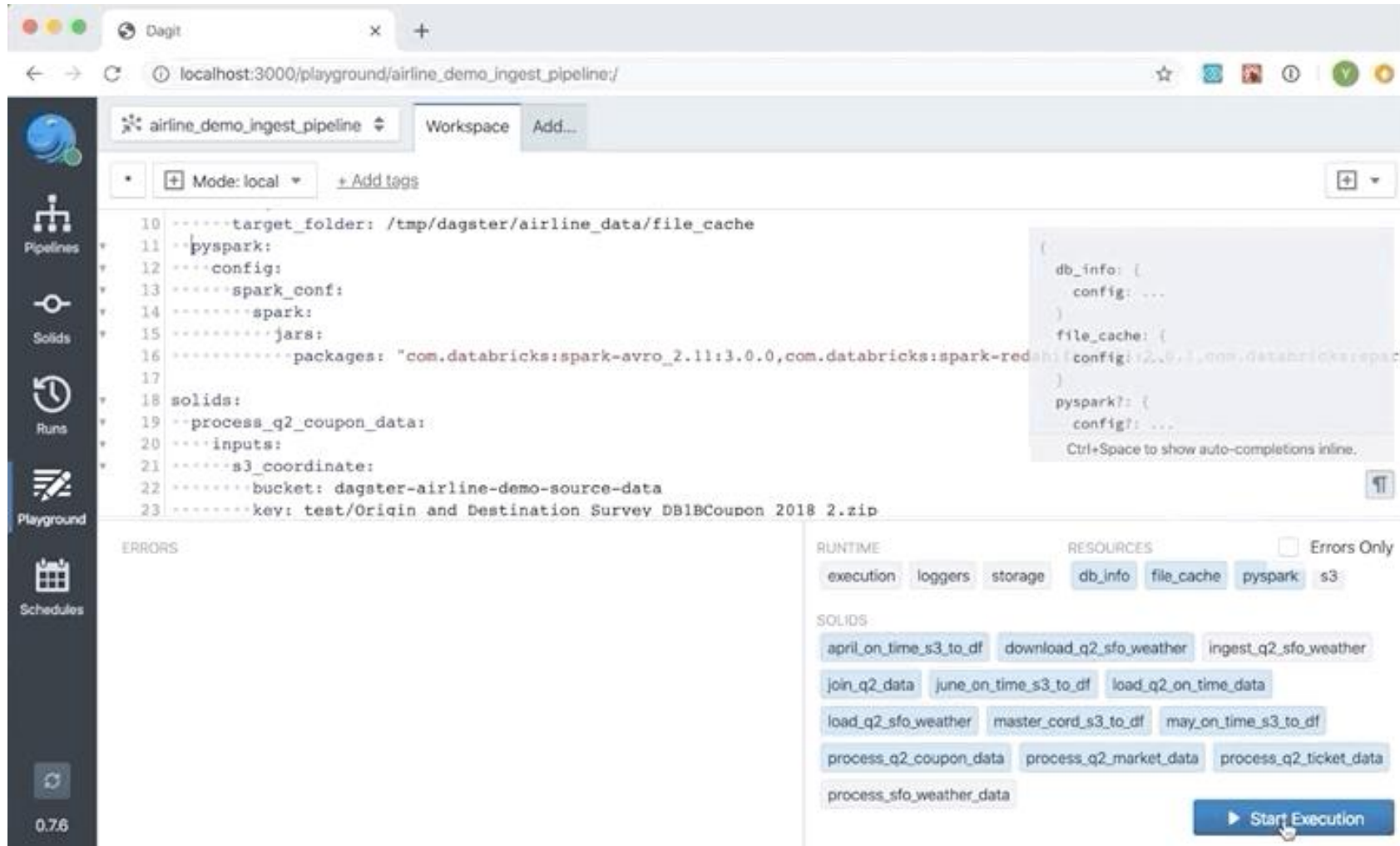
Versionierter Code



git



Dagster Pipelines



The screenshot shows the Dagster web interface in a browser window. The URL is `localhost:3000/playground/airline_demo_ingest_pipeline/`. The interface includes a sidebar with navigation icons for Pipelines, Solids, Runs, Playground (active), and Schedules. The main area displays the pipeline configuration for `airline_demo_ingest_pipeline` in a code editor. The configuration includes a `target_folder`, `pyspark` settings, and a `solids` section with various data processing tasks. A tooltip is visible over the `pyspark` configuration, showing details like `db_info`, `file_cache`, and `pyspark?`. Below the code editor, there are tabs for `ERRORS`, `RUNTIME`, and `RESOURCES`. The `RUNTIME` tab is active, showing a list of solids and a `Start Execution` button.

```

10 .....target_folder: /tmp/dagster/airline_data/file_cache
11 ..pyspark:
12 .....config:
13 .....spark_conf:
14 .....spark:
15 .....jars:
16 .....packages: "com.databricks:spark-avro_2.11:3.0.0,com.databricks:spark-redash:2.0.3,com.databricks:spark-
17 .....
18 solids:
19 ..process_q2_coupon_data:
20 .....inputs:
21 .....s3_coordinate:
22 .....bucket: dagster-airline-demo-source-data
23 .....key: test/Origin and Destination Survey DB1BCoupon 2018 2.zip
  
```

Tooltip content:

```

{
  db_info: {
    config: ...
  }
  file_cache: {
    config: {2.0.3,com.databricks:spark-
  }
  pyspark?: {
    config?: ...
  }
}
Ctrl+Space to show auto-completions inline.
  
```

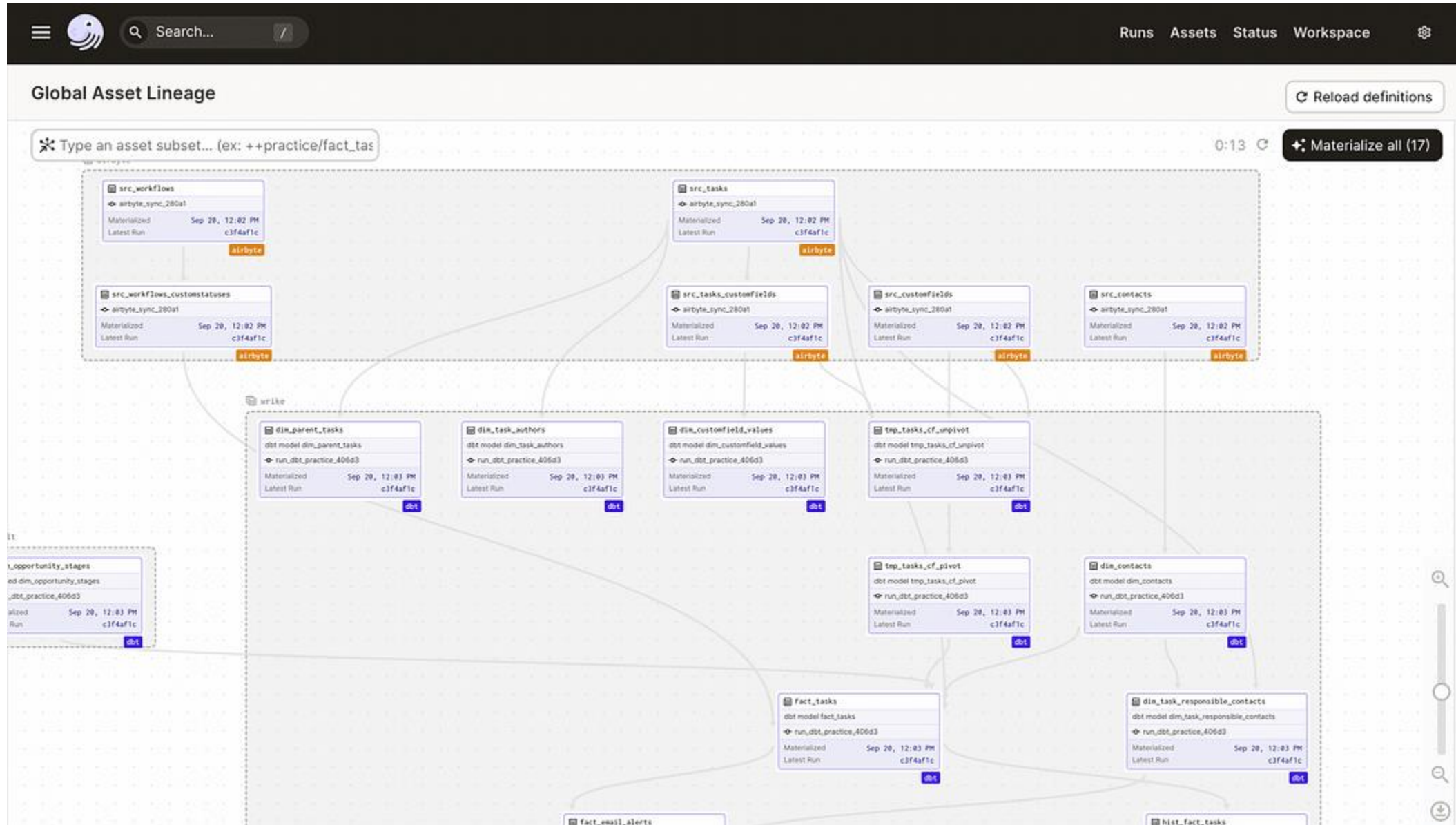
Runtime resources: `execution`, `loggers`, `storage`, `db_info`, `file_cache`, `pyspark`, `s3`.

Solids: `april_on_time_s3_to_df`, `download_q2_sfo_weather`, `ingest_q2_sfo_weather`, `join_q2_data`, `june_on_time_s3_to_df`, `load_q2_on_time_data`, `load_q2_sfo_weather`, `master_cord_s3_to_df`, `may_on_time_s3_to_df`, `process_q2_coupon_data`, `process_q2_market_data`, `process_q2_ticket_data`, `process_sfo_weather_data`.

[Start Execution](#)



DAGSTER

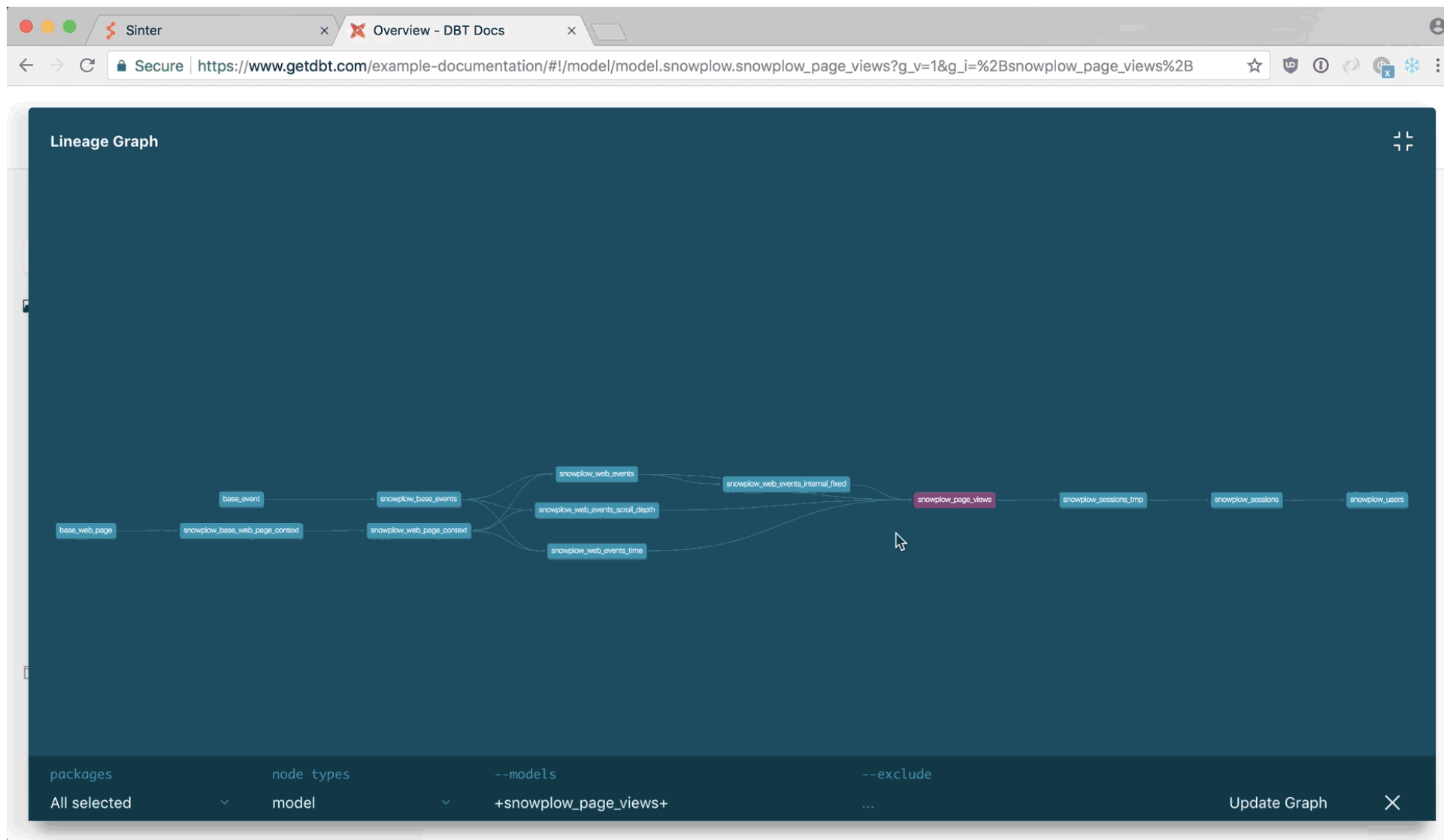


Data Transformation

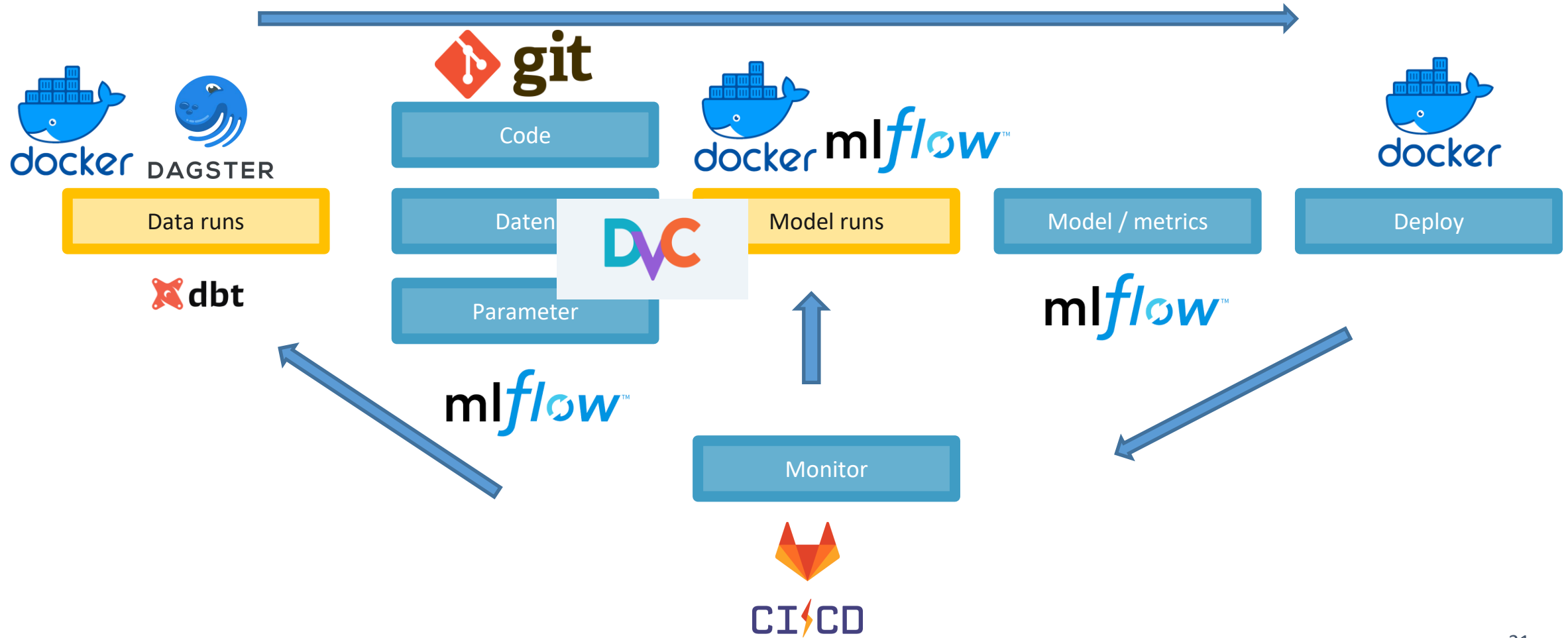


```
dbt-batch > src > dbt-project > models > example > top_nations.sql
1
2  /*
3  |   Find 3 nations with the most active customers
4  */
5
6  {{ config(materialized='table') }}
7
8  select n_name, count(*) as active_buyers
9  from nation n left join {{ ref('top_customers') }} c on n.n_nationkey = c.c_nationkey
10 group by n_name
11 order by active_buyers desc
12 limit 3
```

Data Transformation



MLOps Prozess – OSS Stack

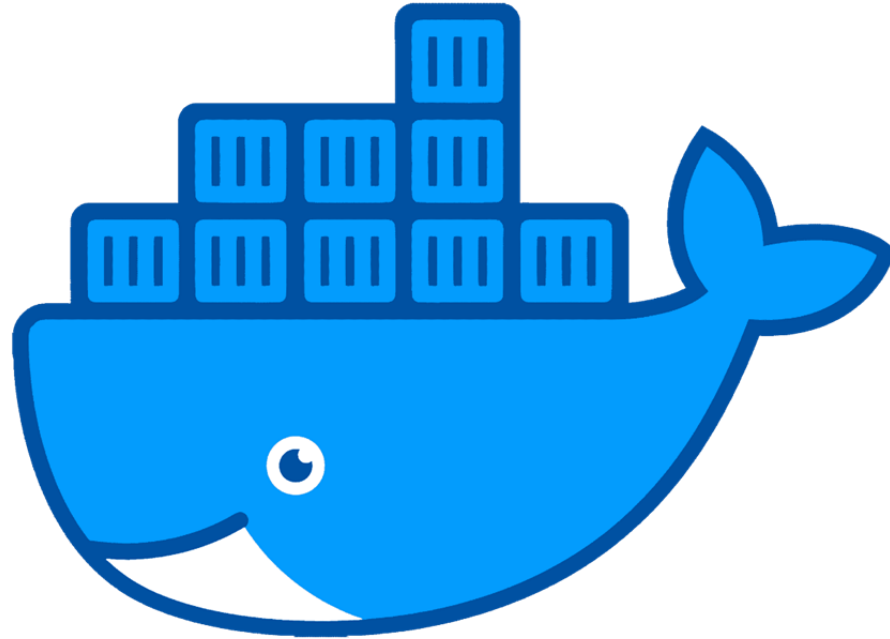


Model Runs

Was soll man speichern?

- **Code**
- **Data Annotations**
- **ML Run Meta**
- **Parameters**
- **Outputs**
- **Environment**

Environment



docker

Model Experiments

[GitHub](#) [Docs](#)

Experiments



Default



highlight



test-run



new1



new1

Experiment ID: 4

Artifact Location: ./mlruns/4

▼ Notes

None

Search Runs:

? State:

Active ▼

Search

Clear

Showing 2 matching runs

Compare

Delete

Download CSV



Columns

			Parameters >						Metrics		
☐	Start Time	Run Name	User	Source	Version	categorical_	colsample_f	early_stopp	accuracy	log_loss	train-multi_
☐	✓ 2020-02-22 14:53:25	-	harutaka	train.py	298a9c	auto	1.0	None	1	0.38	0.421
☐	✓ 2020-02-22 14:51:51	-	harutaka	train.py	298a9c	auto	1.0	None	1	0.38	0.421

```

from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
from sklearn.model_selection import train_test_split
from sklearn.linear_model import ElasticNet

import pandas as pd
import numpy as np

import mlflow

mlflow.set_tracking_uri("http://localhost:5000")
mlflow_experiment_id = 1

csv_url = "http://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality04.csv"
data = pd.read_csv(csv_url, sep=";")

alpha = 0.5
l1_ratio = 0.5

train, test = train_test_split(data)
train_x = train.drop(["quality"], axis=1)
test_x = test.drop(["quality"], axis=1)
train_y = train[["quality"]]
test_y = test[["quality"]]

with mlflow.start_run(experiment_id=mlflow_experiment_id):
    lr = ElasticNet(alpha=alpha, l1_ratio=l1_ratio, random_state=42)
    lr.fit(train_x, train_y)
    predicted_qualities = lr.predict(test_x)
    rmse = np.sqrt(mean_squared_error(test_y, predicted_qualities))
    mae = mean_absolute_error(test_y, predicted_qualities)
    r2 = r2_score(test_y, predicted_qualities)

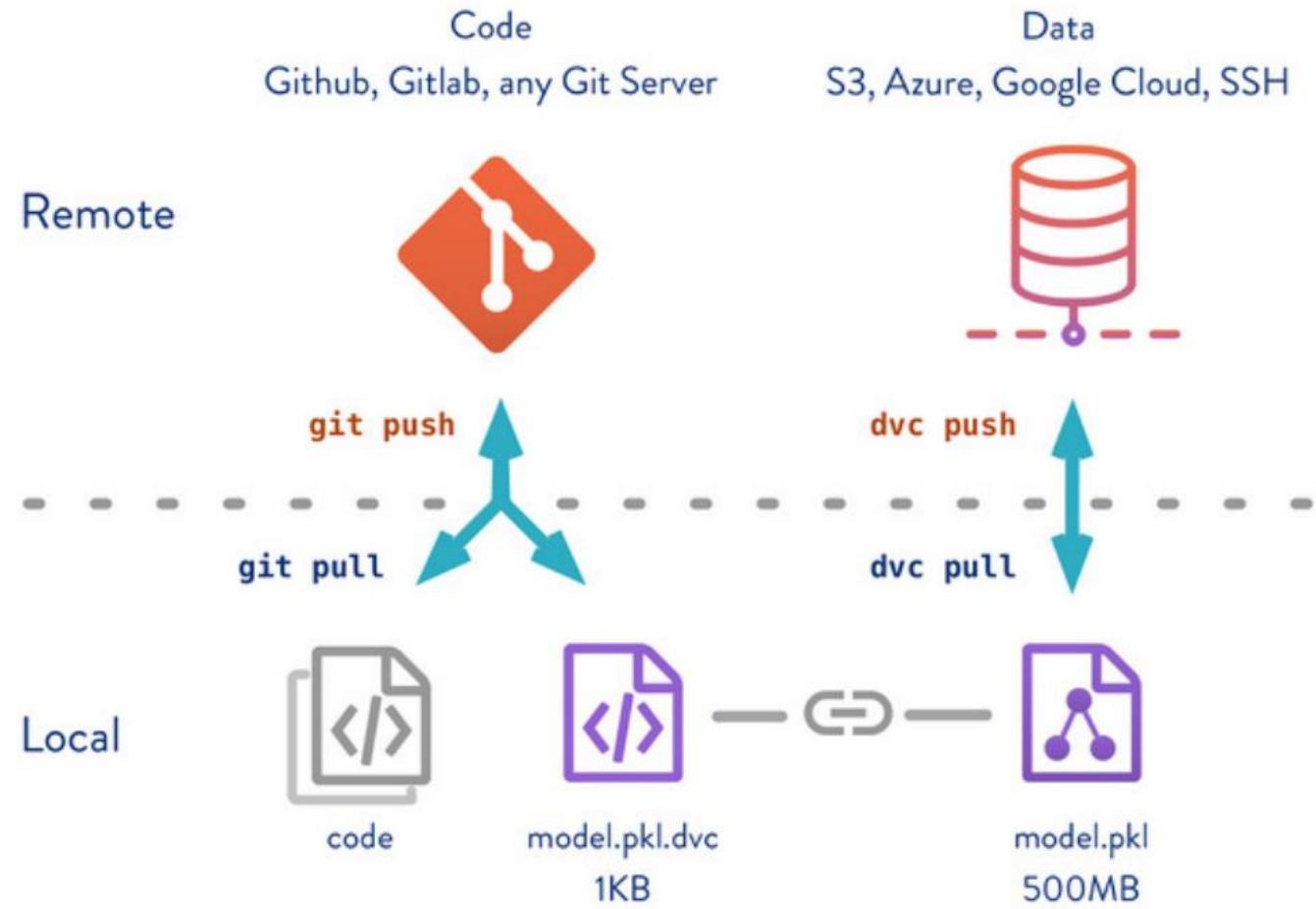
    print("Elasticnet model (alpha=%f, l1_ratio=%f):" % (alpha, l1_ratio))
    print("  RMSE: %s" % rmse)
    print("  MAE: %s" % mae)
    print("  R2: %s" % r2)

mlflow.log_param("alpha", alpha)
mlflow.log_param("l1_ratio", l1_ratio)
mlflow.log_metric("rmse", rmse)
mlflow.log_metric("r2", r2)
mlflow.log_metric("mae", mae)
mlflow.sklearn.log_model(lr, "model")

```



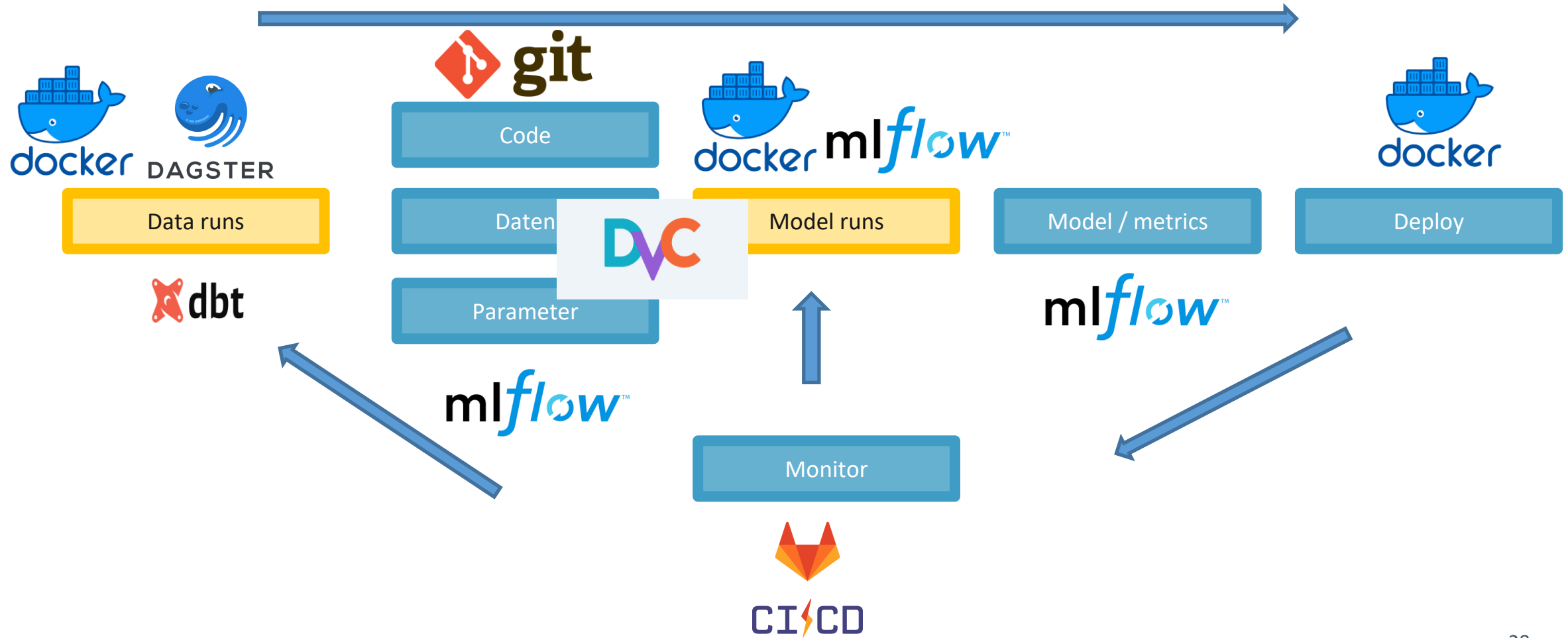
Data Version Control - Outputs



(not just) Data Version Control

Experiment	Created ↓	Message	CML	.../train/metrics.json	results	data	models	params.yaml		
				train	train			train		
				loss	pool-segment...	pool_data	model.pkl	arch	base_lr	epochs
☐ ☐ main :										
☐ ☐ BASELINE HEAD, main, 2...	Aug 08, 2023	Run dvc.yaml pipeline		0.02786	v0.1.0	19.0 MB	200.6 KB	shufflenet_v2_...	0.01	8
☐ ☐ 4b1e630 [bendy-sena]	Aug 10, 2023	Tune train.arch		0.02539	+	19.0 MB	64.0 MB	squeezenet1_1	0.01	8
☐ ☐ a091aaa [shaky-peba]	Aug 10, 2023	Tune train.arch		0.02072	+	19.0 MB	207.3 MB	resnet34	0.01	8
☐ ☐ d2bf3f8 [pseud-phon]	Aug 10, 2023	Tune train.arch		0.02425	+	19.0 MB	25.3 MB	alexnet	0.01	8
☐ ☐ 84f1337 :	Aug 07, 2023	Convert Notebook to dvc.y...		0.02906	+	19.0 MB	—	shufflenet_v2_...	0.01	8

MLOps Prozess – OSS Stack



Monitoring & Operations

Deployment, Testing & Models

- **Model Registry**
- **Pre-Train Tests (Data Quality Tests)**
- **Post-Train Tests (Model Tests)**
- **Metriken**

! schema.yml M X

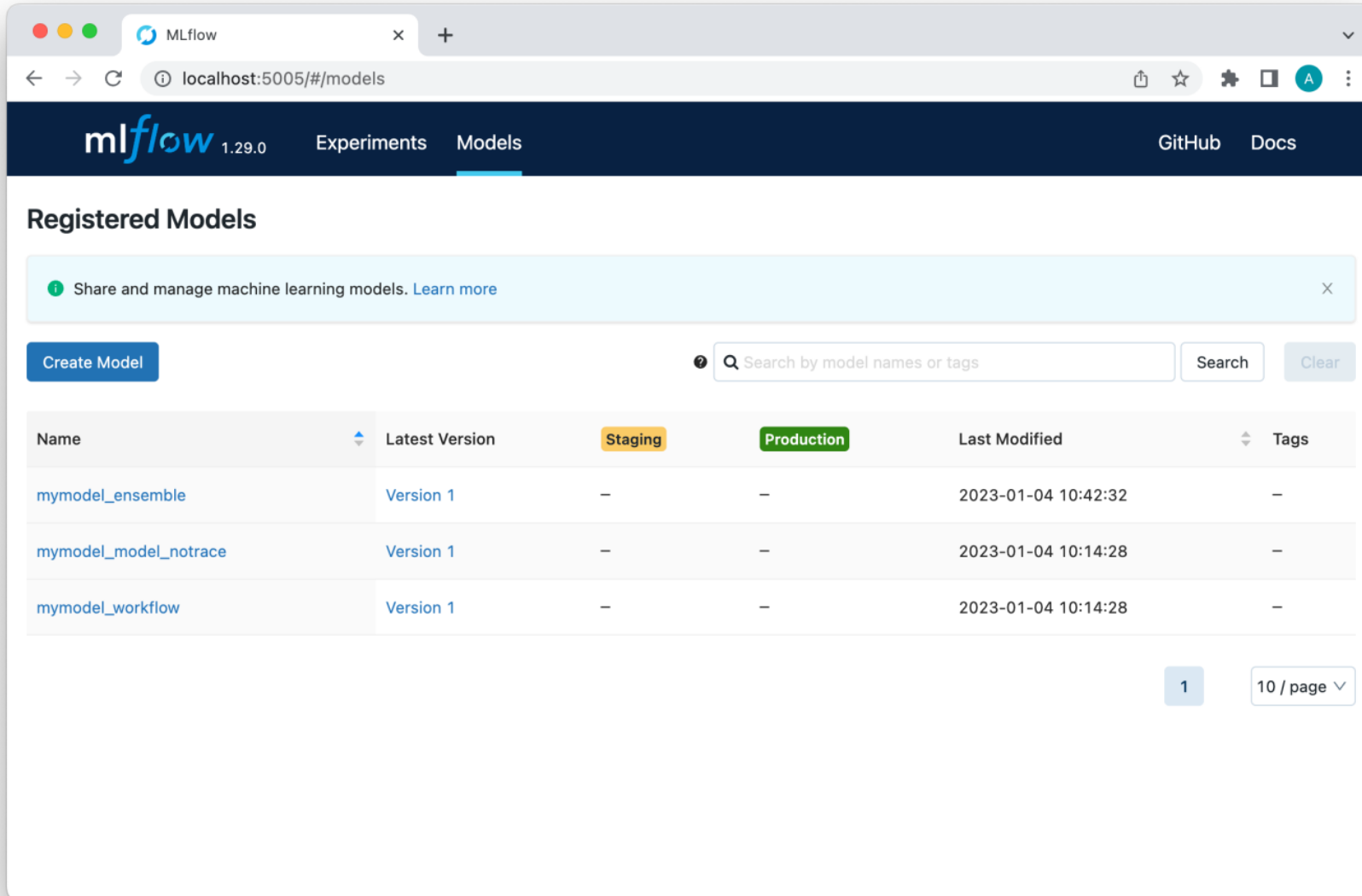
models > ! schema.yml

```
1  version: 2
2
3  sources:
4    - name: raw_covid19
5      schema: airbyte_schema
6      database: covid19_database
7      description: A replica of the covid19 demographics data from Google BigQuery Open data.
8      tables:
9        - name: stg_index
10          columns:
11            - name: _airbyte_index_hashid
12              description: Hash column based on the values in the record.
13              tests:
14                - unique
15                - not_null
16
17          - name: stg_demographics
18            columns:
19              - name: _airbyte_stg_demographics_hashid
20                description: Hash column based on the values in the record.
21                tests:
22                  - unique
23                  - not_null
24
25          - name: stg_economy
26            columns:
27              - name: _airbyte_stg_economy_hashid
28                description: Hash column based on the values in the record.
29                tests:
30                  - unique
31                  - not_null
32
```



```
> dbt test
11:12:23 Running with dbt=1.0.1
11:12:23 Found 1 model, 6 tests, 0 snapshots, 0 analyses, 165 macros, 0 operations, 0 seed files, 0 sources, 0 exposures, 0 metrics
11:12:23
11:12:23 Concurrency: 1 threads (target='dev')
11:12:23
11:12:23 1 of 6 START test not_null_user_email..... [RUN]
11:12:23 1 of 6 PASS not_null_user_email..... [PASS in 0.04s]
11:12:23 2 of 6 START test not_null_user_id..... [RUN]
11:12:23 2 of 6 PASS not_null_user_id..... [PASS in 0.01s]
11:12:23 3 of 6 START test not_null_user_username..... [RUN]
11:12:23 3 of 6 PASS not_null_user_username..... [PASS in 0.01s]
11:12:23 4 of 6 START test unique_user_email..... [RUN]
11:12:23 4 of 6 PASS unique_user_email..... [PASS in 0.01s]
11:12:23 5 of 6 START test unique_user_id..... [RUN]
11:12:23 5 of 6 PASS unique_user_id..... [PASS in 0.01s]
11:12:23 6 of 6 START test unique_user_username..... [RUN]
11:12:23 6 of 6 FAIL 1 unique_user_username..... [FAIL 1 in 0.01s]
11:12:23
11:12:23 Finished running 6 tests in 0.16s.
11:12:23
11:12:23 Completed with 1 error and 0 warnings:
11:12:23
11:12:23 Failure in test unique_user_username (models/example/schema.yml)
11:12:23   Got 1 result, configured to fail if != 0
11:12:23
11:12:23   compiled SQL at target/compiled/tpcds/models/example/schema.yml/unique_user_username.sql
11:12:23
```

Model Registry



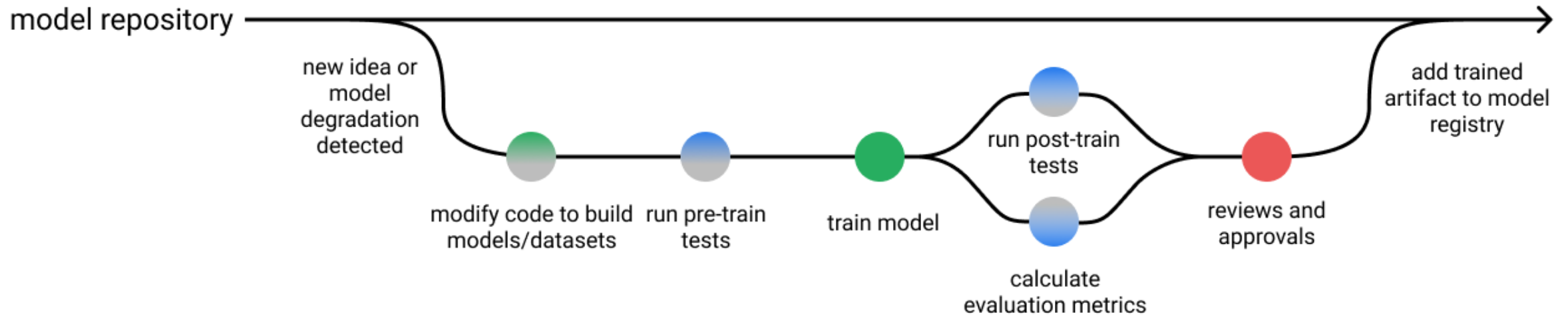
The screenshot shows the MLflow Model Registry web interface. The browser address bar indicates the URL is `localhost:5005/#/models`. The MLflow logo and version `1.29.0` are in the top left, with navigation links for `Experiments` and `Models`. The `Models` tab is active. A top navigation bar includes links to `GitHub` and `Docs`.

Below the navigation bar, the section is titled `Registered Models`. A light blue banner contains the text: `Share and manage machine learning models. Learn more`. A `Create Model` button is on the left. A search bar with the placeholder `Search by model names or tags` and buttons for `Search` and `Clear` is on the right.

Name	Latest Version	Staging	Production	Last Modified	Tags
mymodel_ensemble	Version 1	—	—	2023-01-04 10:42:32	—
mymodel_model_notrace	Version 1	—	—	2023-01-04 10:14:28	—
mymodel_workflow	Version 1	—	—	2023-01-04 10:14:28	—

At the bottom right, there is a pagination control showing `1` and `10 / page` with a dropdown arrow.





Substring - The Data Company

This wasn't very data-driven of you!