WORKSHOP DAY ONSITE
OCTOBER 4, 2023

CONFERENCE DAY ONSITE
OCTOBER 5, 2023

WORKSHOP DAY ONLINE
OCTOBER 6, 2023

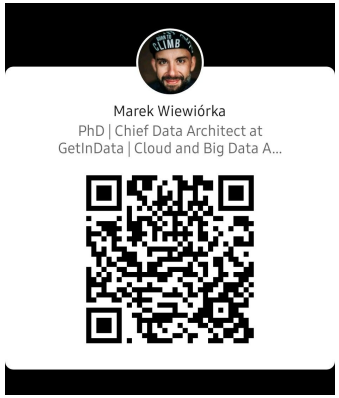
The MLOps platform that makes you productive, everywhere!

Marek Wiewiórka
GetInData | Part of Xebia

SUMMIT.DATAMASS.IO

About me

getindata
Part of Xebia



- Chief Data Architect @**GetInData | Part of Xebia**
- Research Assistant at the Warsaw University of Technology
- An Open source contributor to Snowflake Terraform Provider, SeQuiLa and Kedro plugins
- Personally a keen long distance runner and gravel bikes enthusiast

GetInData - At a Glance

getindata
Part of Xebia

- Experts in **Big Data, Cloud, Analytics and ML/AI solutions**
- Team of 120+ consultants, **~60% senior level**
- Experience in: **media, e-commerce, retail, fintech, banking, and telco**
- We work with **digital natives where data is core business** (Spotify, Truecaller, Acast, Volt), as well as with traditional enterprises where data is used for improvements
- **A go-to partner** for companies that need tailored and highly scalable data processing and analytics platforms that give competitive advantage and **unlock the full business potential of data**.

SOLUTION AREAS

MLOps & Modern Data Platforms

Data & ML engineering project accelerators
Read: Kedro plugins, DP Framework

Stream processing & real-time analytics

Technologies

dbt, snowflake, databricks, Looker, Airbyte, Kedro, jupyter, Flink, Apache Airflow, mlflow

Selected USE CASES

1. Volt.io (Fintech)

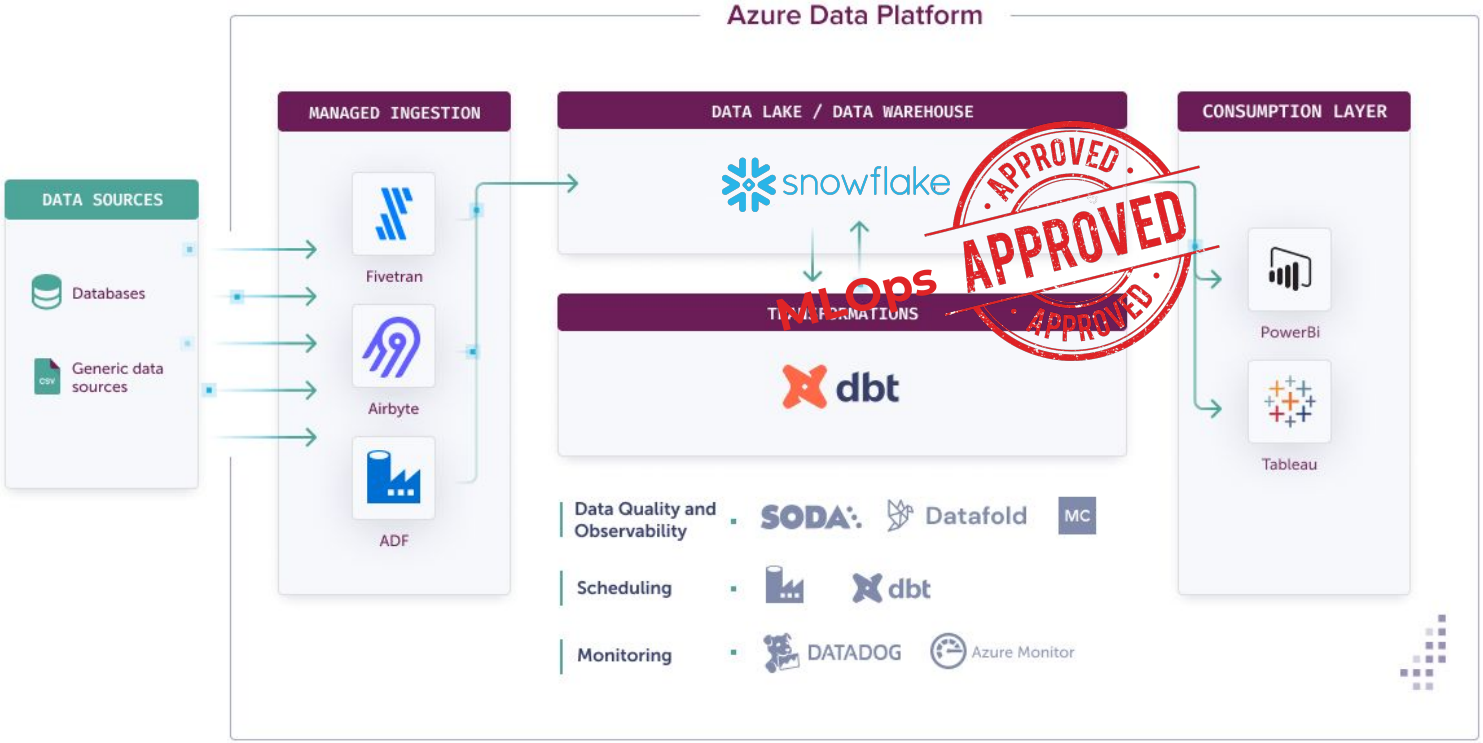
- **Snowflake-based Modern Data Platform**
- Just **4 months** to build from scratch to insights
- Strong focus on platform security
- The right mix of open-source and cloud-managed technologies

2. Other customer references

ING, PLAY, iZettle, truecaller, Kcell, VEOLIA, Acast

From MDP[1] to MDP (MLOps-enabled Data Platform)

getindata
Part of Xebia



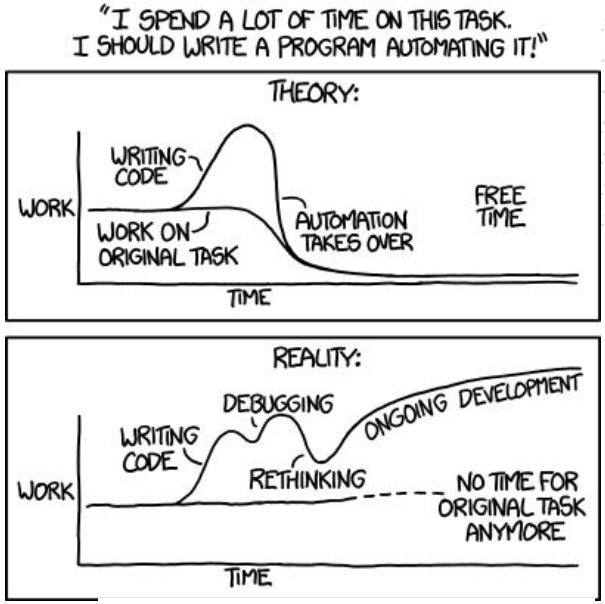
[1]MDP - Modern Data Platform

What MLOps is (*not only*) about ?

getindata
Part of Xebia

- Application of the DevOps principles to ML world
- Managing ML model lifecycle
- Tools and platforms
- *Automation* and processes
- Infrastructure as Code

The ultimate goal is: **PRODUCTIVITY**



Source: xkcd by Randall Munroe. Automation takes a life of its own.

GID MLOps “Productivity Manifesto”

getindata
Part of Xebia

- Machine Learning and data science should be ***first-class*** citizens of Data Platforms
- Lightweight and KISS
- ***Open*** standards and cloud ***agnosticism***
- Short development ***feedback loop*** (incl. local dev)
- ***Fast*** new ML projects bootstrapping and ***standardization***
- Execution environment ***independent*** training pipelines
- Easy ***collaboration***
- … MLOps capabilities provisioned ***in days not months***

ML projects in layers

Data Scientist

Experimentation + EDA

Machine Learning frameworks

Example technologies:

ML projects in layers

Data Scientist

Example technologies:

Experimentation + EDA

Machine Learning frameworks

?

Execution environment

Data

MLOps / ML Engineer

Building blocks of the GID MLOps

getindata
Part of Xebia

Data Scientist

Experimentation + EDA

Machine Learning frameworks

Portable MLOps framework

Experiment tracking and collaboration

IaC and automation

Kedro

mlflow

Terraform

Cloud Integrations (incl. GID Kedro plugins)

Execution environment

Data

MLOps / ML Engineer

Example technologies:

jupyter

learn

XGBoost

GID MLOps Platform

Building blocks of the GID MLOps

getindata
Part of Xebia

Data Scientist

Experimentation + EDA

Machine Learning frameworks

Portable MLOps framework

Kedro

Experiment tracking and collaboration

mlflow

IaC and automation

Terraform

Cloud Integrations (incl. GID Kedro plugins)

Execution environment

Data

MLOps / ML Engineer

Example technologies:

jupyter

learn

XGBoost

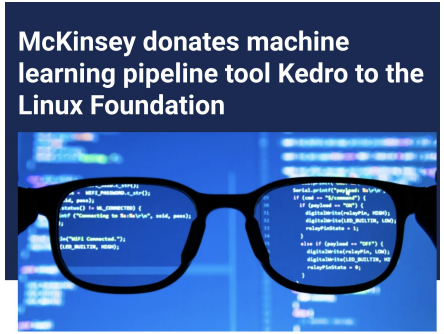
GID MLOps Platform

What is Kedro?

getindata
Part of Xebia

Kedro

= Software Engineering Principles + Data Science



Kedro *is an open-source Python framework for creating reproducible, maintainable and modular data science code.*

What features does Kedro have?

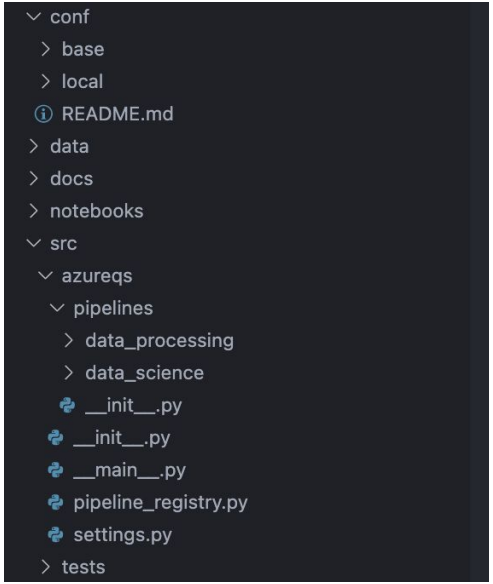
getindata
Part of Xebia

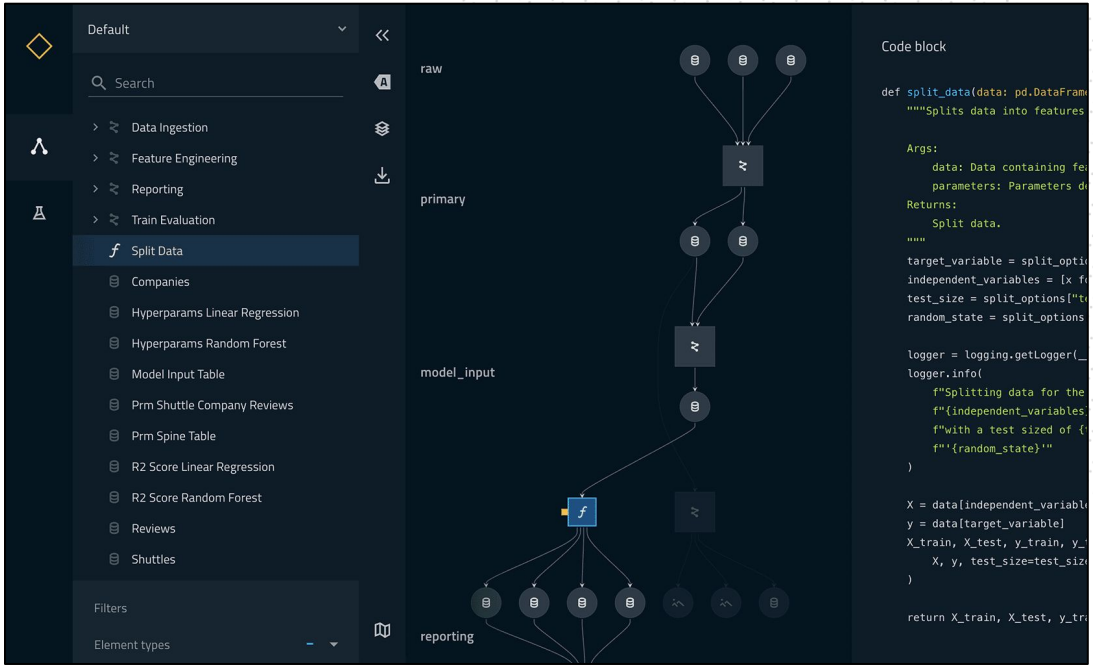
```
conf
  base
  local
  README.md
data
docs
notebooks
src
  azureqs
    pipelines
      data_processing
      data_science
      __init__.py
    __init__.py
    __main__.py
    pipeline_registry.py
    settings.py
  tests
```

Well defined project structure + project starters

What features does Kedro have?

getindata
Part of Xebia





Well defined project structure + project starters

Nodes & pipelines abstractions

Kedro pipeline - data science & engineering

getindata
Part of Xebia

```python
def create_pipeline(**kwargs) -> Pipeline:
    return pipeline(
        [
            node(
                func=preprocess_companies,
                inputs="companies",
                outputs="preprocessed_companies",
                name="preprocess_companies_node",
            ),
            node(
                func=preprocess_reviews,
                inputs="reviews",
                outputs="preprocessed_reviews",
                name="preprocess_reviews_node",
            ),
            node(
                func=create_model_input_table,
                inputs=["preprocessed_reviews", "prepr
                outputs="model_input_table",
                name="create_model_input_table_node",
            ),
        ]
    )
```

```python
def create_pipeline(**kwargs) -> Pipeline:
    return pipeline(
        [
            node(
                func=split_data,
                inputs=["model_input_table", "params:model_options"],
                outputs=["X_train", "X_test", "y_train", "y_test"],
                name="split_data_node",
            ),
            node(
                func=train_model,
                inputs=["X_train", "y_train"],
                outputs="regressor",
                name="train_model_node",
            ),
            node(
                func=evaluate_model,
                inputs=["regressor", "X_test", "y_test"],
                outputs=None,
                name="evaluate_model_node",
            ),
        ]
    )
```

Kedro node

getindata
Part of Xebia

```python
def create_pipeline(**kwargs) -> Pipeline:
    return pipeline(
        [
            node(
                func=preprocess_companies,
                inputs="companies",
                outputs="preprocessed_companies",
                name="preprocess_companies_node",
            ),
            node(
                func=preprocess_reviews,
                inputs="reviews",
                outputs="preprocessed_reviews",
                name="preprocess_reviews_node",
            ),
            node(
                func=create_model_input_table,
                inputs=["preprocessed_reviews", "
                outputs="model_input_table",
                name="create_model_input_table_node",
            ),
        ]
    )
```

```python
def create_model_input_table(
    reviews: pd.DataFrame, companies: pd.DataFrame, ratings: pd.DataFrame
) -> pd.DataFrame:
    """Combines all data to create a model input table.

    Args:
        reviews: Preprocessed data for reviews.
        companies: Preprocessed data for companies.
        ratings: Raw data for ratings.
    Returns:
        Model input table.

    """
    reviews_with_ratings = reviews.merge(ratings, left_on="id", right_on="rating_id")
    model_input_table = reviews_with_ratings.merge(
        companies, left_on="company_id", right_on="id"
    )
    model_input_table = model_input_table.dropna()
    return model_input_table
```

What about parameters?

getindata
Part of Xebia

```python
def create_pipeline(**kwargs) -> Pipeline:
    return pipeline(
        [
            node(
                func=split_data,
                inputs=["model_input_table", "params:model_options"],
                outputs=["X_train", "X_test", "y_train", "y_test"],
                name="split_data_node",
            ),
            node(
                func=train_model,
                inputs=["X_train", "y_train"],
                outputs="regressor",
                name="train_model_node",
            ),
            node(
                func=evaluate_model,
                inputs=["regressor", "X_test", "y_test"],
                outputs=None,
                name="evaluate_model_node",
            ),
        ]
    )
```

What about parameters?

getindata
Part of Xebia

```python
def create_pipeline(**kwargs) -> Pipeline:
    return pipeline(
        [
            node(
                func=split_data,
                inputs=["model_input_table", "
                outputs=["X_train", "X_test",
                name="split_data_node",
            ),
            node(
                func=train_model,
                inputs=["X_train", "y_train"],
                outputs="regressor",
                name="train_model_node",
            ),
            node(
                func=evaluate_model,
                inputs=["regressor", "X_test",
                outputs=None,
                name="evaluate_model_node",
            ),
        ]
    )
```

- conf
  - base
    - parameters
      - data_processing.yml
      - data_science.yml
    - azureml.yml
    - catalog.yml
    - logging.yml
    - parameters.yml
  - local
- data
- docs

```yaml
model_options:
  test_size: 0.2
  random_state: 3
  features:
    - engines
    - passenger_capacity
    - crew
    - d_check_complete
    - moon_clearance_complete
    - iata_approved
    - company_rating
    - review_scores_rating
```

What about data - Kedro data catalog!

getindata
Part of Xebia

```python
def create_pipeline(**kwargs) -> Pipeline:
    return pipeline(
        [
            node(
                func=preprocess_companies,
                inputs="companies",
                outputs="preprocessed_companies",
                name="preprocess_companies_node",
            ),
            node(
                func=preprocess_reviews,
                inputs="reviews",
                outputs="preprocessed_reviews",
                name="preprocess_reviews_node",
            ),
            node(
                func=create_model_input_table,
                inputs=["preprocessed_reviews", "p
                outputs="model_input_table",
                name="create_model_input_table_nod
            ),
        ]
    )
```

- conf
  - base
    - parameters
    - azure_ml.yml
    - catalog.yml
    - logging.yml
    - parameters.yml
  - local
- data
- docs
- notebooks
- src

```yaml
companies:
  type: pandas.CSVDataSet
  filepath: data/01_raw/companies.csv

reviews:
  type: pandas.ParquetDataSet
  filepath: data/01_raw/reviews.parquet

pictures:
  type: pillow.ImageDataSet
  filepath: data/01_raw/images/*.jpg
```

Configuration and code separation with envs!

getindata
Part of Xebia

Local catalog.yml

```yaml
companies:
  type: pandas.CSVDataSet
  filepath: data/01_raw/companies.csv

reviews:
  type: pandas.ParquetDataSet
  filepath: data/01_raw/reviews.parquet

pictures:
  type: pillow.ImageDataSet
  filepath: data/01_raw/images/*.jpg
```

Cloud catalog.yml

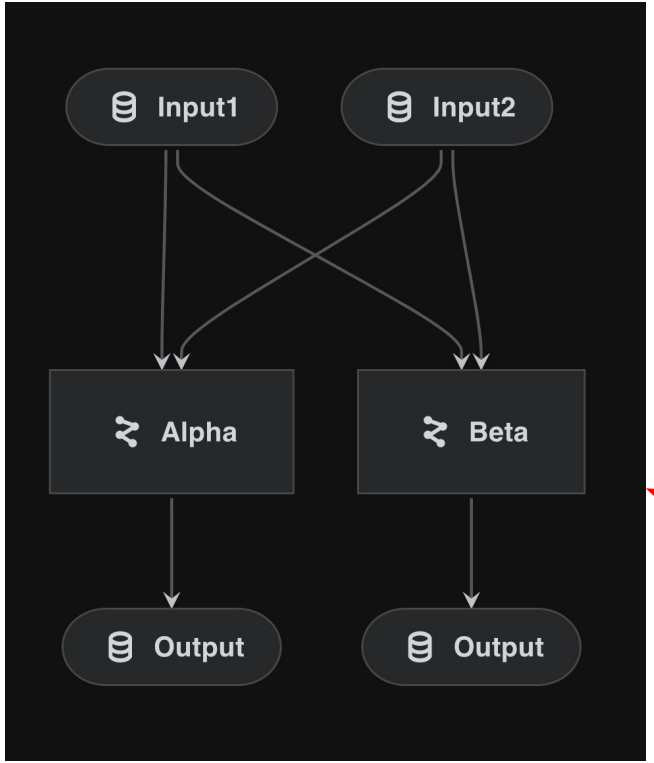
```yaml
companies:
  type: pandas.CSVDataSet
  filepath: abfs://my_blob_container/data/01_raw/companies.csv

reviews:
  type: pandas.SQLQueryDataSet
  sql: "select * from reviews;"
  credentials: db_credentials

pictures:
  type: kedro_azureml.AzureMLFileDataSet
  dataset: my_dataset_from_azureml
  filepath: data/01_raw/images/*.jpg
```

Data Catalog

What other features does Kedro have?

getindata
Part of Xebia

Modular and dynamic pipelines!



Extensibility & Integrations

Kedro can be integrated with multiple industry leading solutions, including: Apache Spark, Pandas, Dask, Matplotlib, Plotly, fsspec, Apache Airflow, Jupyter Notebook and Docker.

Stay tuned for our blog post soon!

Are we done yet?

getindata
Part of Xebia

Building blocks of the GID MLOps

getindata
Part of Xebia

Data Scientist

Experimentation + EDA

Machine Learning frameworks

Portable MLOps framework

Experiment tracking and collaboration

IaC and automation

Kedro

mlflow

Terraform

Cloud Integrations (incl. GID Kedro plugins)

Execution environment

Data

MLOps / ML Engineer

Example technologies:

jupyter

learn

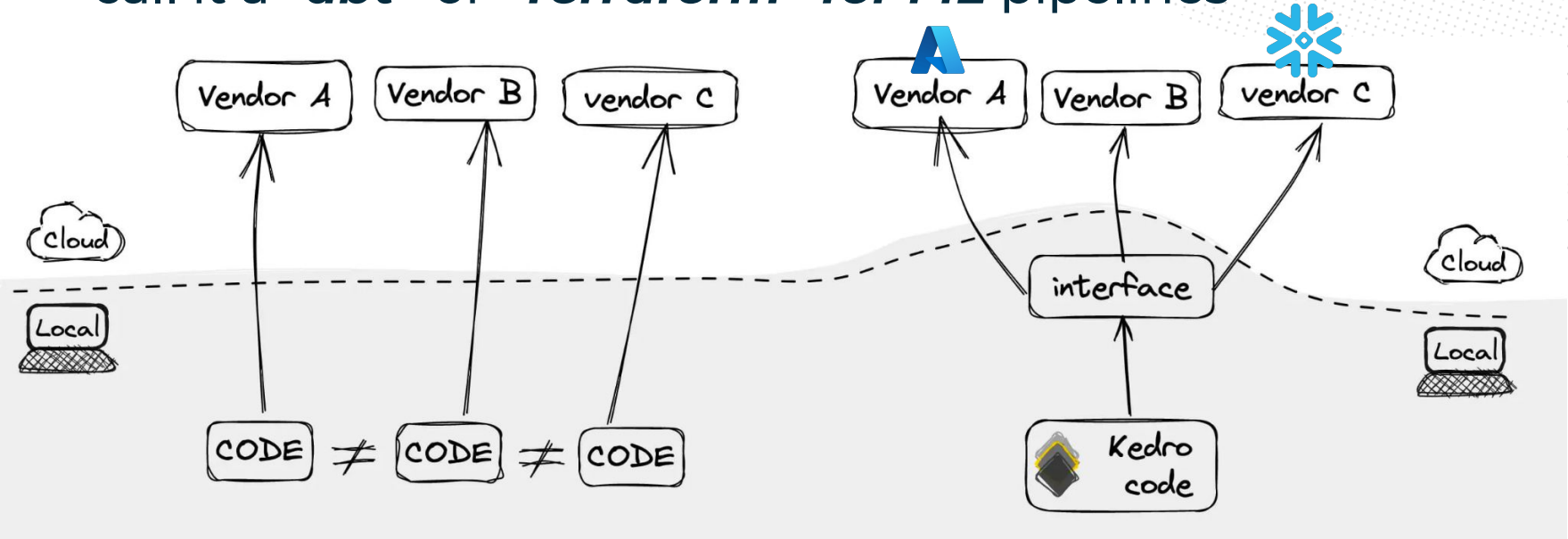
XGBoost

GID MLOps Platform

Kedro plugins to run your ML pipelines, everywhere!

getindata
Part of Xebia

- Kedro is claimed to be a “React” for ML … but we prefer to call it a “***dbt***” or “***Terraform” for ML*** pipelines



Source: [Xebia blog]

Write once - run (almost) everywhere

getindata
Part of Xebia

Kedro

Kedro Vertex AI (GCP)
github.com/getindata/kedro-vertexai

Kedro Sagemaker (AWS)
github.com/getindata/kedro-sagemaker

Kedro Kubeflow (Kubernetes)
github.com/getindata/kedro-kubeflow

Kedro AzureML (Azure)
github.com/getindata/kedro-azureml

Kedro Snowflake (all clouds)
github.com/getindata/kedro-snowflake

In progress: ***Kedro DBX***

Read more on our blog: Running Kedro... everywhere? Machine Learning Pipelines on Kubeflow, Vertex AI, Azure and Airflow

Write once - run (almost) everywhere

getindata
Part of Xebia

Kedro

Kedro Vertex AI (GCP)
github.com/getindata/kedro-vertexai

Kedro Sagemaker (AWS)
github.com/getindata/kedro-sagemaker

Kedro Kubeflow (Kubernetes)
github.com/getindata/kedro-kubeflow

Kedro AzureML (Azure)
github.com/getindata/kedro-azureml

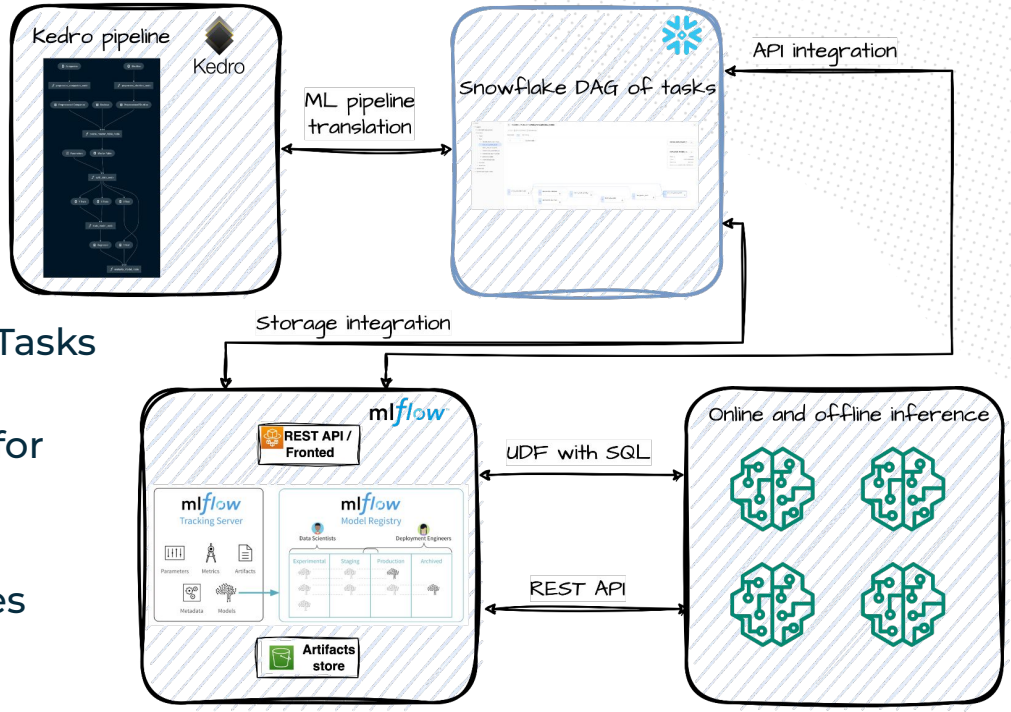
Kedro Snowflake (all clouds)
github.com/getindata/kedro-snowflake

In progress: ***Kedro DBX***

Read more on our blog: Running Kedro... everywhere? Machine Learning Pipelines on Kubeflow, Vertex AI, Azure and Airflow

Kedro-Snowflake - putting it all together...

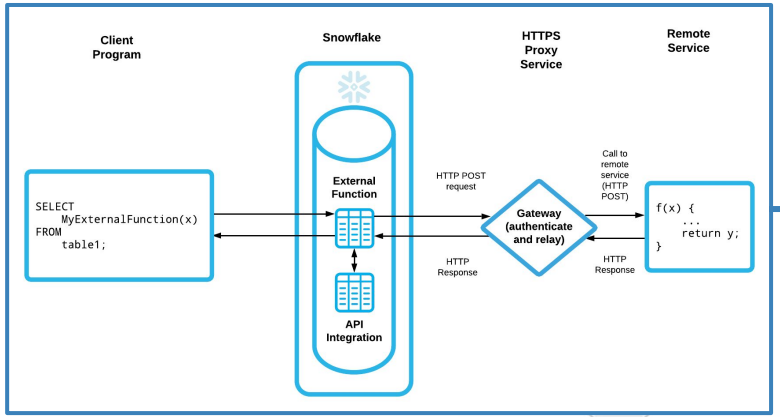
getindata
Part of Xebia



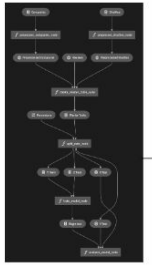
- Native ***Snowpark ML*** and Tasks integration
- ***MLflow Snowflake*** plugin for deployment as *UDF*
- MLflow **Sagemaker** - *REST*
- Set of ***Terraform*** of modules
- Built-in Kedro ***starter***

MLOps Platform for Snowflake - GCP

getindata
Part of Xebia



- Snowflake **external functions** and **API Gateway**



CI/CD

Package,upload to stage & register as UDF

mlflow-snowflake

Get model storage URI

Download model artifact

@ml_model_stage (internal)

Native ML pipeline

kedro-snowflake

Upload ML artifacts

@mlflow_stage (external)

External functions (log_parameter, log_metric)

API Gateway

GCP integration

mlflow

MLflow Fronted

MLflow Artifacts

MLflow Backend

MLOps Platform for Snowflake- GCP

getindata
Part of Xebia

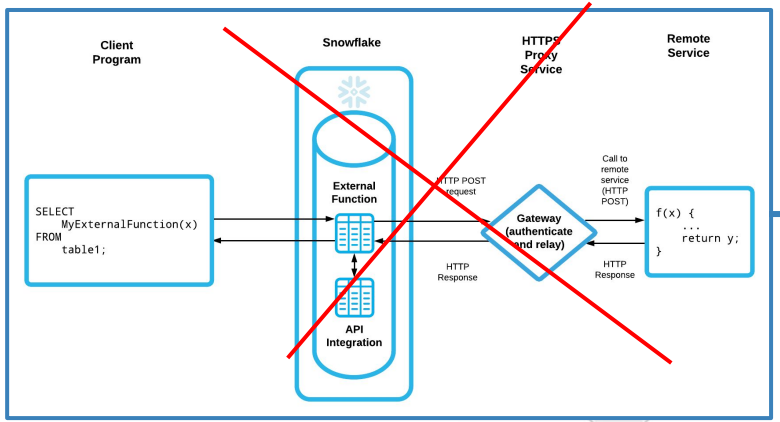
- Glue code for requests/responses to MLflow API
- PR to the Snowflake provider

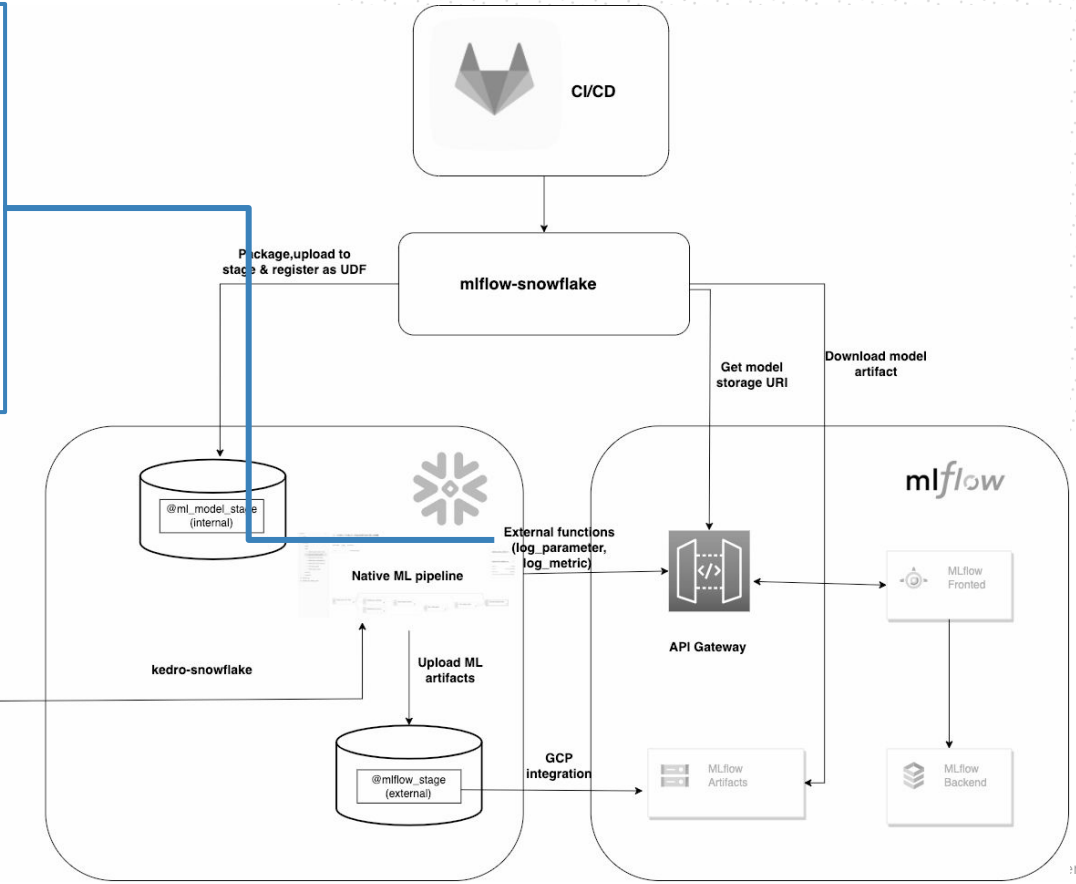
```
resource "snowflake_external_function" "mlflow_run_create" {
  name     = upper("mlflow_run_create")
  database = var.database_name
  schema   = var.schema_name
  arg {
    name = "experiment_id"
    type = "varchar"
  }
  return_type               = "OBJECT"
  return_behavior           = "VOLATILE"
  api_integration           = snowflake_api_integration.mlflow_gcp.name
  request_translator = "${var.database_name}.${var.schema_name}.${snowflake_function.mlflow_run_create_req.name}"
  response_translator = "${var.database_name}.${var.schema_name}.${snowflake_function.mlflow_generic_res.name}"
  url_of_proxy_and_resource = "${var.api_gateway_url}/api/2.0/mlflow/runs/create"
}
```

```
resource "snowflake_function" "mlflow_run_create_req" {
  name     = upper("mlflow_run_create_req")
  database = snowflake_database.db.name
  schema   = snowflake_schema.schema.name
  arguments {
    name = "event"
    type = "OBJECT"
  }
  comment             = "Request translator for MLflow create run"
  return_type         = "OBJECT"
  language            = "javascript"
  statement           = <<EOH
      let exeprimentId = EVENT.body.data[0][1]
      let timestamp = new Date().getTime();
      return { "body": { "experiment_id": exeprimentId, start_time: timestamp }}
  EOH
}
```

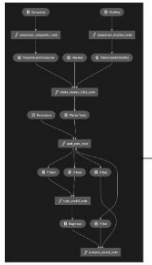
MLOps Platform for Snowflake - AWS/Azure

getindata
Part of Xebia





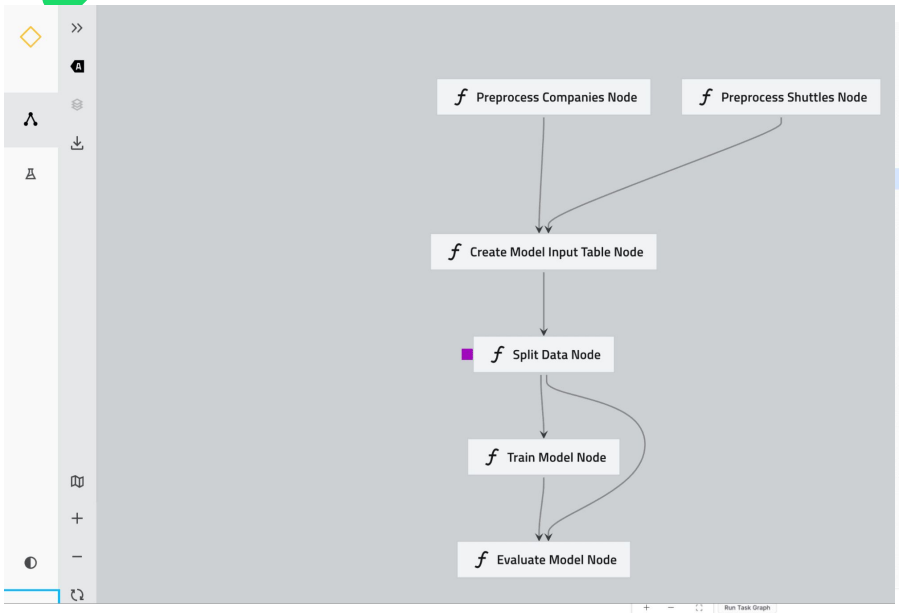
Kedro

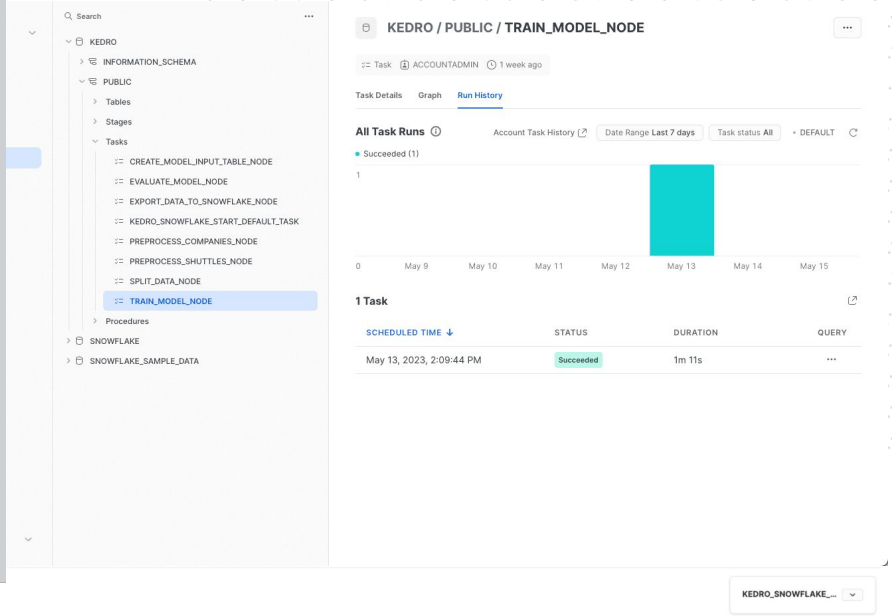


- Snowflake external **integrations** and **network** rules
- **No** need for API Gateway

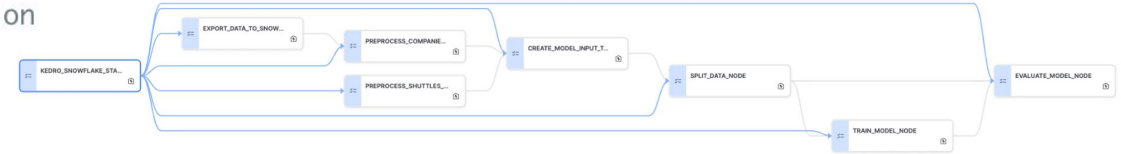
MLOps Platform for Snowflake

getindata
Part of Xebia






```
kedro snowflake run --wait-for-completion
```



MLOps Platform for Snowflake

getindata
Part of Xebia

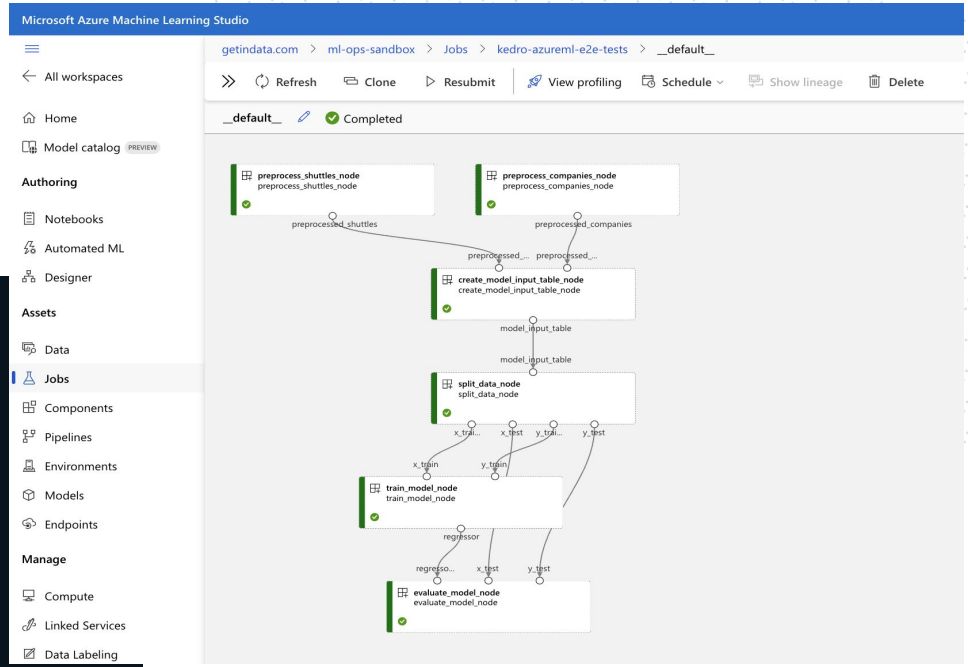
- Support for native Snowflake Tables and Stages in Kedro Data catalog

```yaml
companies_snowflake:
  type: kedro_datasets.snowflake.SnowparkTableDataSet
  table_name: companies_snowflights_starter
  credentials: snowflake
  save_args:
    mode: overwrite


preprocessed_shuttles:
  type: kedro_snowflake.datasets.native.SnowflakeStageFileDataSet
  stage: "@KEDRO_SNOWFLAKE_TEMP_DATA_STAGE" # <-- Snowflake stage to store data in
  filepath: data/02_intermediate/preprocessed_shuttles.csv # <-- file path within the stage
  credentials: snowflake # <-- credentials to connect to Snowflake (the same as for SnowparkTableDataSet)
  dataset: # <-- dataset key defines the dataset type to use
    type: pandas.CSVDataSet # <-- specify any params for the nested dataset here
```

MLOps Platform for Azure

getindata
Part of Xebia

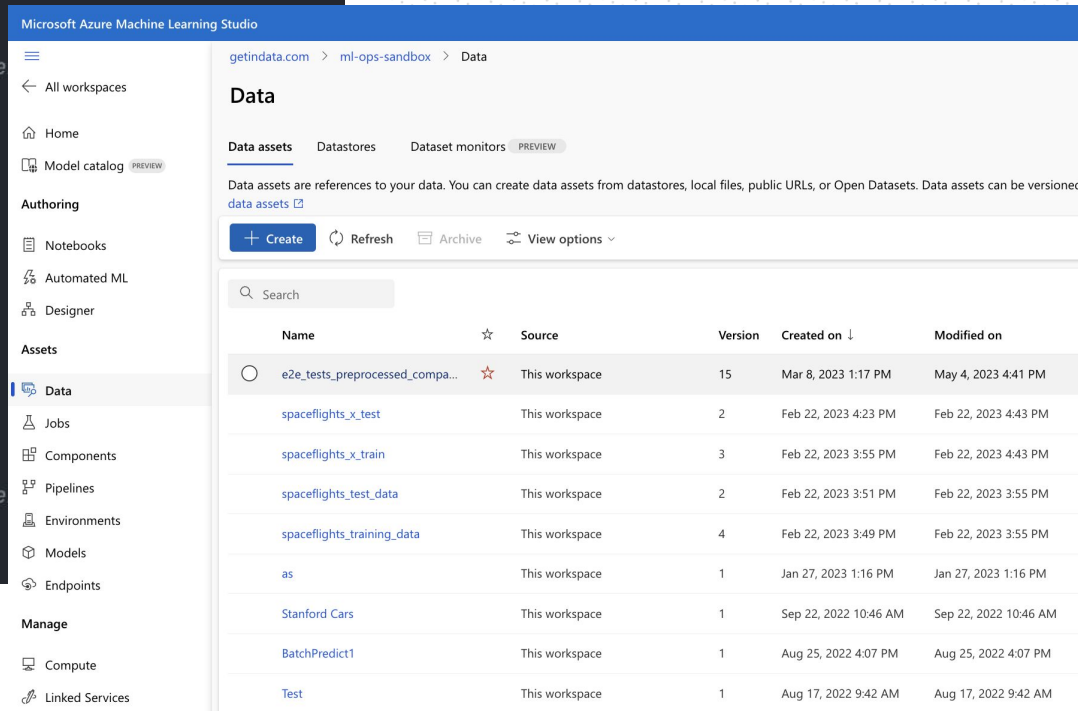
- ***Kedro*** AzureML plugin
- Azure ML builtin“**MLflow”**
- Set of ***Terraform*** of modules



Kedro-AzureML - datasets

getindata
Part of Xebia

```yaml
my_pandas_dataframe_dataset:
  type: kedro_azureml.datasets.AzureMLPandasDataSet
  azureml_dataset: my_new_azureml_dataset

  # if version is not provided, the latest dataset ve
  azureml_dataset_load_args:
    version: 1


processed_images:
  type: kedro_azureml.datasets.AzureMLFileDataSet
  dataset: pillow.ImageDataSet
  filename_suffix: '.png'
  azureml_dataset: processed_images
  azureml_dataset_save_args:
    create_new_version: true

  # if version is not provided, the latest dataset ve
  azureml_dataset_load_args:
    version: 1
```



MLOps Azure DevOps pipeline

getindata
Part of Xebia

Azure DevOps marcin-getindata / gidmlopsv7 / Pipelines / MLOps demo pipeline / 20230510.1

gidmlopsv7

- Overview
- Boards
- Repos
- Pipelines
- Pipelines
- Environments
- Releases
- Library
- Task groups
- Deployment groups

Jobs in run #20230510.1

MLOps demo pipeline

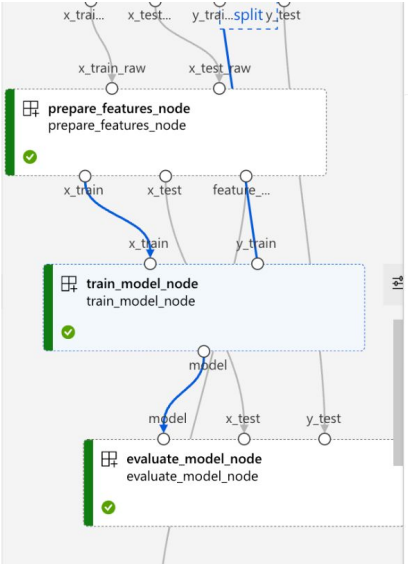
Build & run

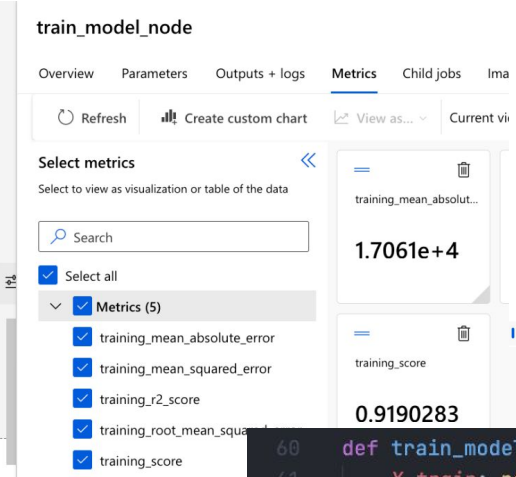
MLOps Pipeline	42s
Initialize job	1s
Checkout MLOps by GetInData ...	1s
Login to ACR	<1s
Pull latest image	1s
Build	39s
Push	
Run Kedro pipeline in Azure ML	
Post-job: Checkout MLOps by GetI...	

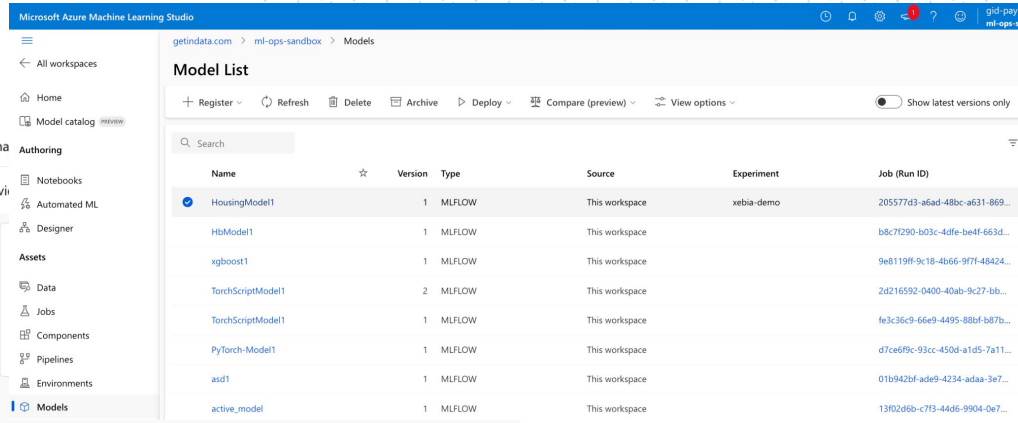
Build

```
488        ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 78.2/78.2 KB 238.3
489  Collecting pytoolconfig[global]>=1.2.2
490    Downloading pytoolconfig-1.2.4-py3-none-any.whl (16 kB)
491    Downloading pytoolconfig-1.2.2-py3-none-any.whl (16 kB)
492  Collecting appdirs>=1.4.4
493    Downloading appdirs-1.4.4-py2.py3-none-any.whl (9.6 kB)
494  Collecting idna<4,>=2.5
495    Downloading idna-3.4-py3-none-any.whl (61 kB)
496        ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 61.5/61.5 KB 224.5
497  Collecting PySocks!=1.5.7,>=1.5.6
498    Downloading PySocks-1.7.1-py3-none-any.whl (16 kB)
499  Collecting jeepney>=0.6
500    Downloading jeepney-0.8.0-py3-none-any.whl (48 kB)
501        ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 48.4/48.4 KB 203.5
502  Collecting greenlet!=0.4.17
503    Downloading greenlet-2.0.2-cp39-cp39-manylinux_2_17_x86_64.ma
504        ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 610.9/610.9 KB 293.6
505  Collecting backports.weakref
506    Downloading backports.weakref-1.0.post1-py2.py3-none-any.whl
507  Collecting pyproject_hooks
```

Azure ML “MLflow”








```python
def train_model(
    X_train: pd.DataFrame, y_train: pd.Series, random_state: int, model_params: dict
):
    """Train the model on the training data."""
    mlflow.sklearn.autolog(
        log_input_examples=True, log_model_signatures=True, log_models=True
    )
    model = RandomForestRegressor(random_state=random_state, **model_params)
    model.fit(X_train, y_train)
    return model
```

Model deployment - Azure DevOps

getindata
Part of Xebia

Azure DevOps marcin-getindata / gidmlopsv7 / Pipelines / MLOps demo pipeline

gidmlopsv7

- Overview
- Boards
- Repos
- Pipelines
- Pipelines
- Environments
- Releases
- Library
- Task groups
- Deployment groups
- Test Plans
- Artifacts

Project settings

MLOps demo pipeline

Runs | Branches | Analytics

Description

#20230510.1 • Update azure-pipelines.yml
Manually triggered for MZ master bc329b3c

Run pipeline

Select parameters below and manually run the pipeline

Branch/tag
master
Select the branch, commit, or tag

☐ Run training pipeline

☑ Deploy model to Online Endpoint

Name of the model to deploy
HousingModel:1

Name of the endpoint to deploy to
mlops-getindata-demo-endpoint

Description of the endpoint
Demo endpoint for mlops-getindata-demo

Name of the deployment
mlops-getindata-demo-deployment

Instance type for the deployment
Standard_DS2_v2

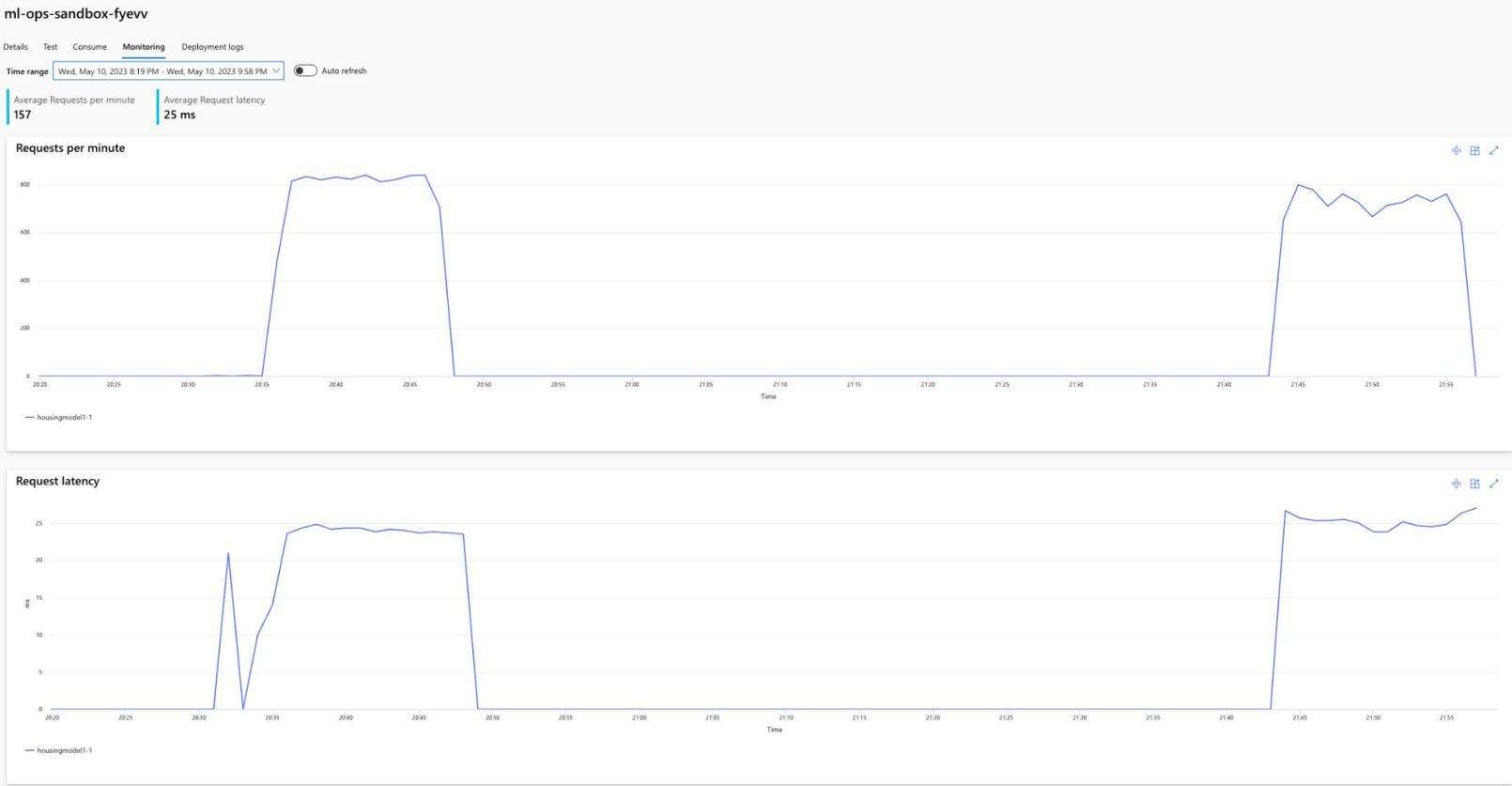
Authentication method for the endpoint (Key or AMLToken)
Key

Traffic allocation for the deployment
100

☐ Enable system diagnostics

Cancel | Run

Model Monitoring



3 Take-home messages

getindata
Part of Xebia

- Kedro is one of the best MLOps frameworks to make data scientists more ***productive*** out-of-the-box
- GetinData contributions to Kedro enable users to extend their Data Platforms with MLOps capabilities ***seamlessly***
- ***Kedro*** together with ***MLflow*** and ***Terraform*** are the main building blocks of ***our MLOps platform***

getindata
Part of Xebia

MLOps Lab

Core TEAM

Michał Bryś
Machine Learning Architect

Marek Wiewiórka
Chief Data Architect

Marcin Zabłocki
MLOps Architect

Artur Dobrogowski
MLOps Engineer

Take DATA Pill and join DATA Matrix

getindata
Part of Xebia

- Best content from **Big Data, Cloud, AI/ML** and more
- simple, **condensed formula**
- sent **weekly** every Friday morning

Subscribe now: