Azure Machine Learning & Databricks : friends or foe ?

Wenesday, the 1st of june, 2022

Speaker : Paul PETON

- Manager Analytics & Data Science @AVANADE Nantes
  - *We are hiring*
- Microsoft MVP Data Platform & AI since 2018
- Specialised in the industrialisation of ML models on Azure

- Blog : https://methodidacte.org
- LinkedIn : https://www.linkedin.com/in/paul-peton-datascience/
- Twitter : @paulpeton

I can do everything with only one tool

In French :

"touche à tout, bon à rien"

In English :

"all-purpose, no-purpose"

https://freesvg.org/utopic-unicorn

Agenda : the right tool at the right place

- « Ops » for Machine Learning
- DEMO : a taxi story…
- Azure ML + Databricks
- MLOps stack
- Azure ML + Synapse Analytics ?
- #finops : do it cheaper

A definition of « Ops » for Machine Learning

Automatisation

Continuous X

Versioning

Testing

Reproductibilité

Monitoring

“**End to End**“: from raw data to predictive use of the model

Test developments as much as possible: unit tests, non-regression tests, etc.

Deploy versioned developments in a continuous and automated way

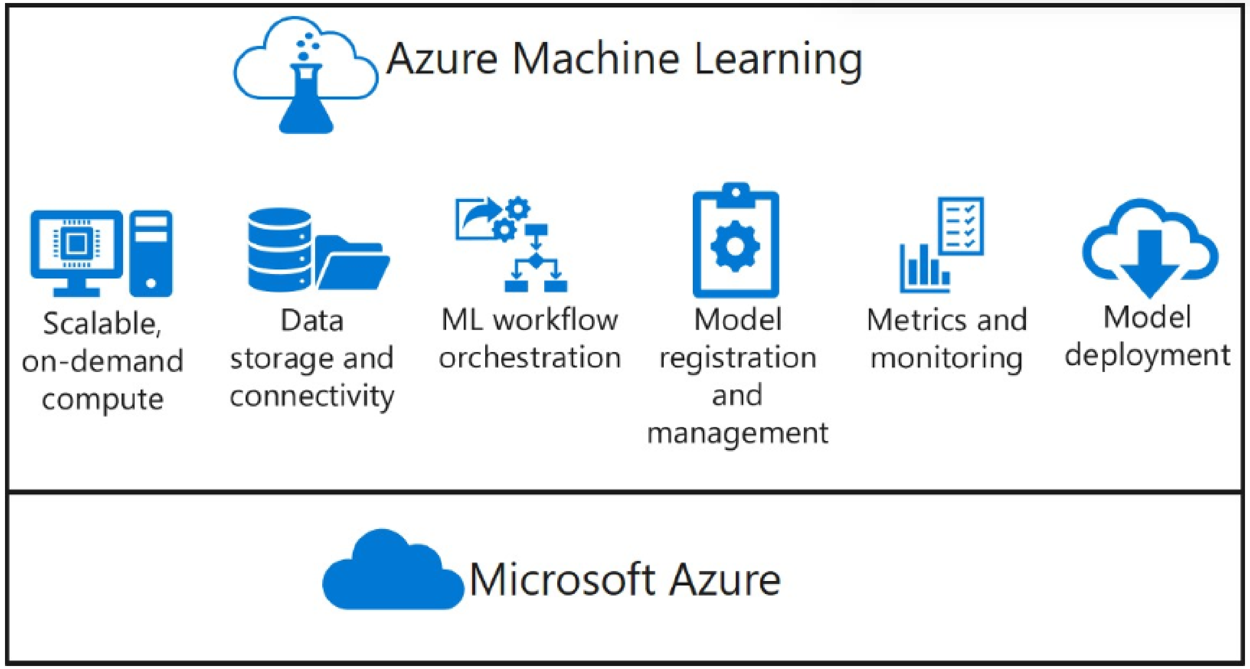
Monitor:

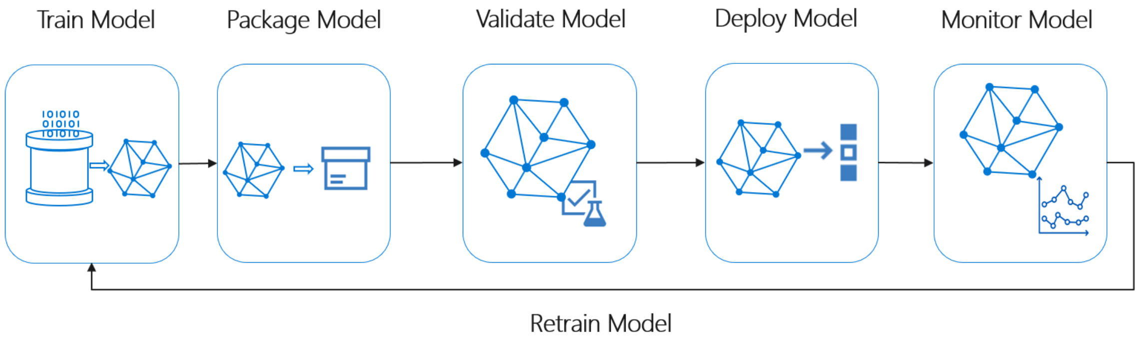
- storage / calculation / exposure resources
- serving: response time AND response "quality

React to drifts: data drift, concept drift, etc.

Azure Machine Learning in a few words

- Portal for
  - Citizen Data Scientists
    - no code, autoML, data labelling
  - Data Engineers
    - code + logs
- MLOps tool
  - From dev to production
  - Store, compute, versioning, expose





Azure Databricks in a few words

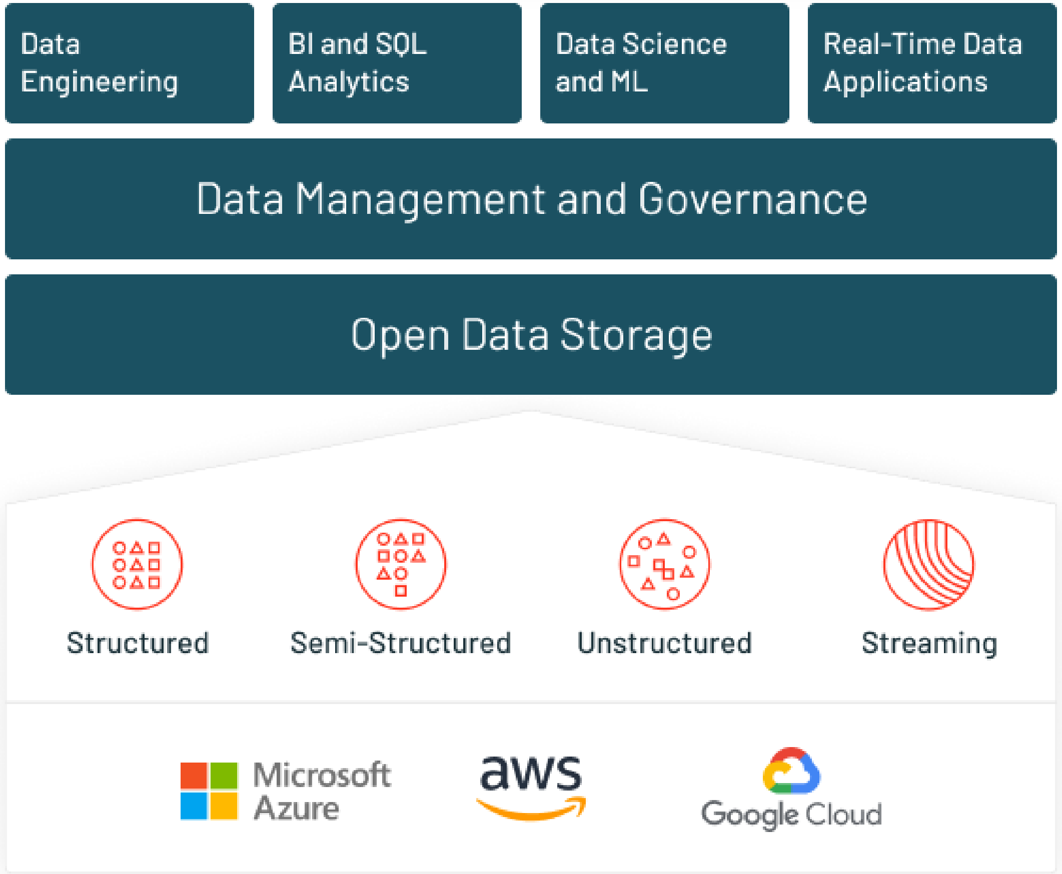
databricks

- Available on the three major public clouds : AWS, Azure, GCP
- Product vision : « *unified analytics platform* »
- Define the « **Lakehouse** » approach
- Integrate the Open Source tool **MLFlow**
- Partnership with the tool **dbt**

mlflow

dbt_

- Open Source languages (Scala, Python, R) and SQL
- Ability to '**scale out**' with the power of Spark, a distributed computing engine



A taxi story…

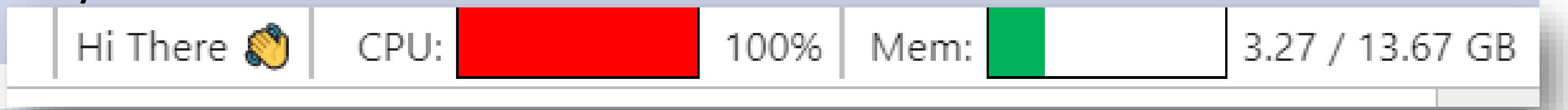
- Yellow Taxi dataset
- Data from 2018 to march 2022
  - Web site
  - Monthly Parquet files
- Stored in a Azure Data Lake gen2
- 252 094 475 rows
- Objective :
  - Load the dataset in memory
  - Train a model of “fare_amount”
  - Register the model
  - Prediction in batch or real-time

First try : notebook in Azure ML

Define a Dataset in Azure ML

Compute instance : Standard_DS3_v2 4 cores, 14 GB RAM, 28 GB disk

Jupyter : memory error => restart Kernel



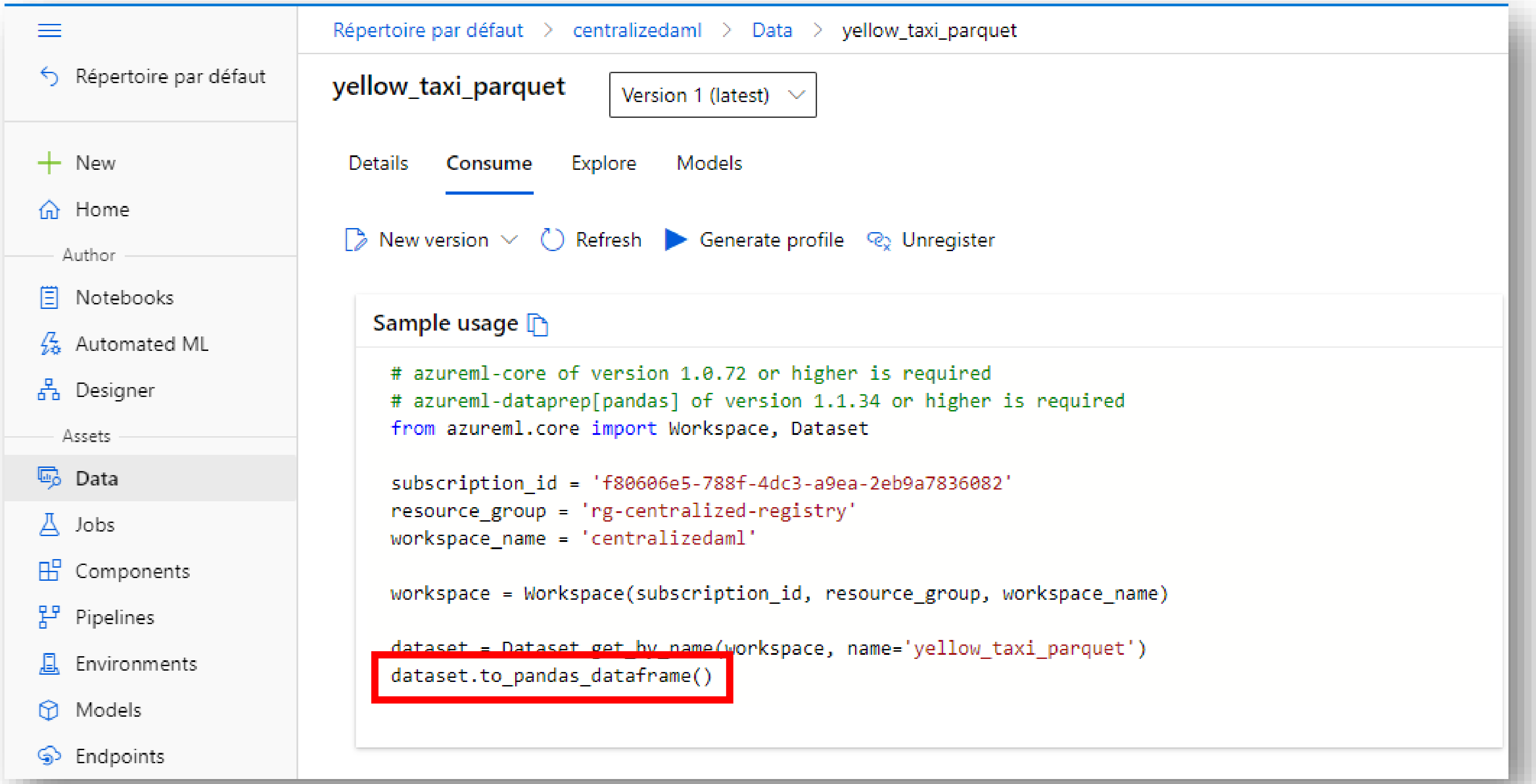
We need to scale up !

It's not a shame

But it can be done differently

Read data as a Pandas dataframe

pandas



pandas provides data structures for in-memory analytics

Scale up : the only option ?

Instance	vCPU	RAM (GiB)	Temporary storage (GiB)	Pay as you go ($)
DS1 v2	1	3,5	7	**91,98**
DS2 v2	2	7	14	**183,96**
DS3 v2	4	14	28	**367,92**
DS4 v2	8	28	56	**735,84**
DS5 v2	16	56	112	**1471,68**

And what about a compute target ?

Define a compute cluster in Azure ML and run code remotely

VM size : Standard_DS3_v2

Maximum number of nodes : 4

Run error

UserError

```
{'code': ExecutionFailed, 'message': [{"exit_code":-1,"error_message":"Possibly out of memory. Execution
failed with error: INFO: current python version is 3.8.13 (default, Mar 28 2022, 11:38:47) \n[GCC
```

Can’t distribute Pandas dataframe

We need a Spark context !

Going to Azure Databricks...

Start a Databricks cluster, authenticate to Azure ML workspace

VM size : Standard_DS3_v2 — 1 worker and 1 driver node

dataset.to_spark_dataframe() — Should work with Scala 2.11, issue on Scala 2.12

```
java.lang.NoSuchMethodError: okhttp3.HttpUrl.get(Ljava/lang/String;)Lokhttp3/HttpUrl;
```

Can’t read data from Datastore — Forget about Azure ML Datasets !

… trying to distribute ML training

Try with Scikit-Learn which needs a Numpy Array

Same issue as Pandas dataframe — We need to use Spark libraries

```
org.apache.spark.SparkException: Job aborted due to stage failure: Task 7 in stage 12.0 failed 4 times, most recent failure: Lost task 7.3 in stage 12.0 (TID 82) (10.139.64.6 executor 0): java.lang.UnsupportedOperationException: org.apache.parquet.column.values.dictionary.PlainValuesDictionary$PlainDoubleDictionary
```

MLLib is obsolete — Spark ML could be an option

And what about SynapseML ? — Let’s see this wonderful library !

Synapse Machine Learning library

*SynapseML (previously MMLSpark) is an **open source library** to simplify the creation of scalable machine learning pipelines. SynapseML builds **on Apache Spark and SparkML** to enable new kinds of machine learning, analytics, and model deployment workflows.*

*SynapseML adds many deep learning and data science tools to the Spark ecosystem, including **seamless integration of Spark Machine Learning pipelines** with :*

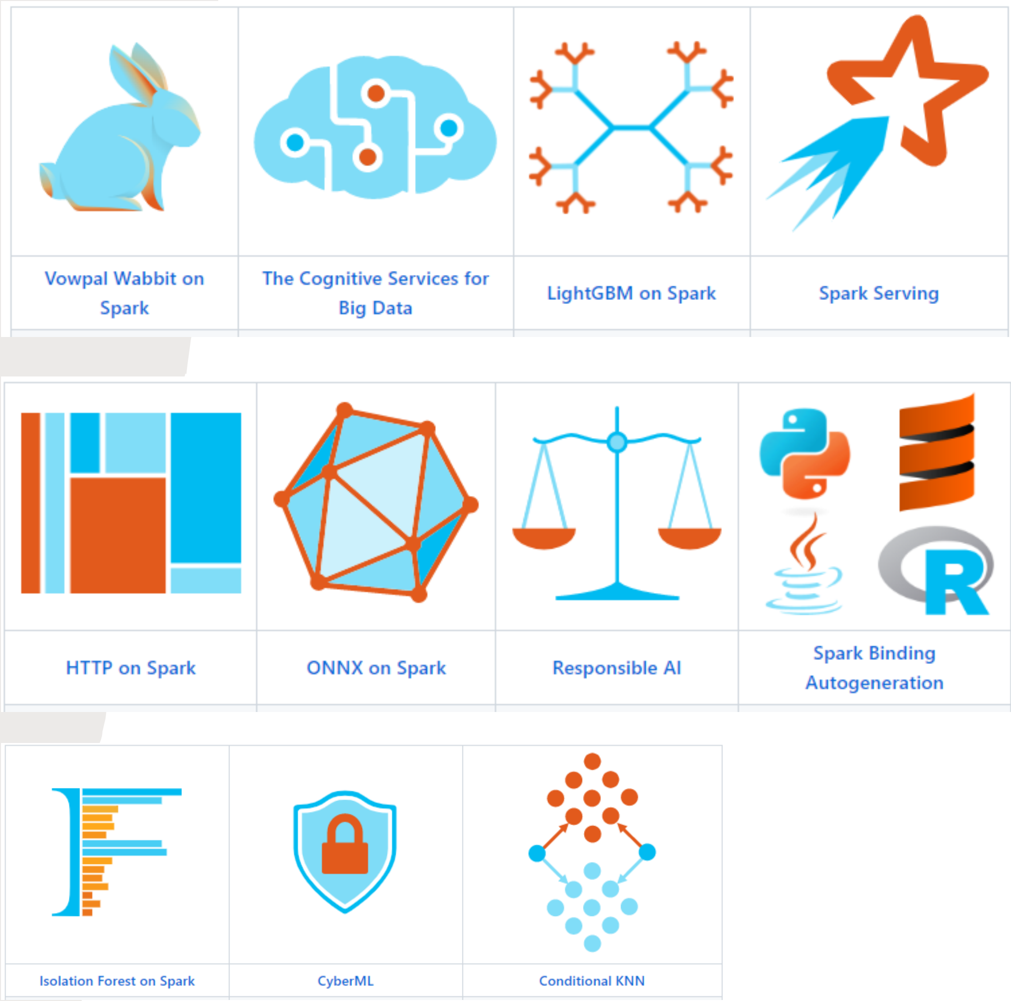
Open Neural Network Exchange (ONNX)

LightGBM

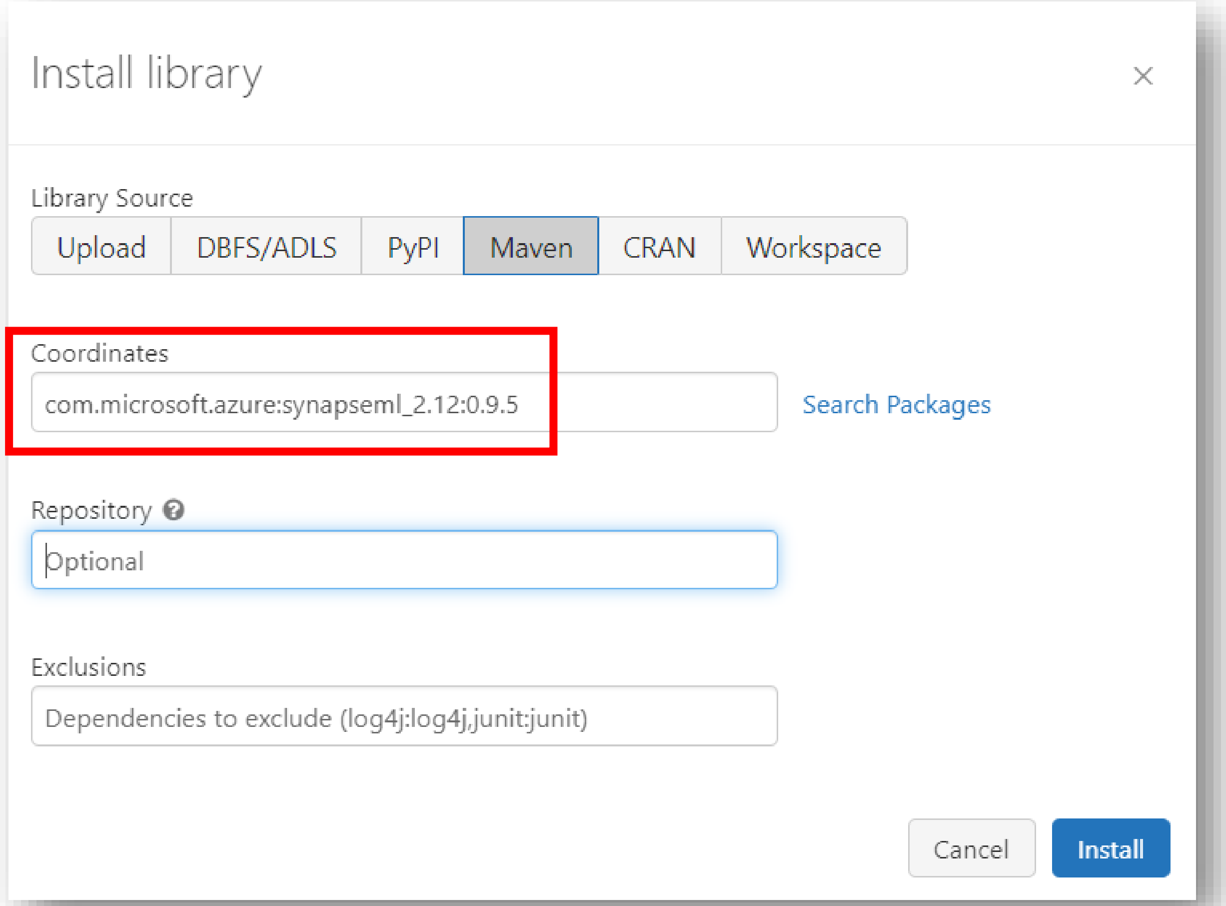
The Cognitive Services

Vowpal Wabbit

OpenCV.



Install SynapseML on Azure Databricks



SynapseML requires **Scala 2.12**, **Spark 3.2+**, and **Python 3.6+**.

LightGBM is an open-source, distributed, high-performance gradient boosting (GBDT, GBRT, GBM, or MART) framework. LightGBM is part of Microsoft's DMTK project.

Advantages :

- Composability : can be incorporated into existing SparkML Pipelines
- Performance : 10-30% faster than SparkML
- Functonality : wide array of tunable parameters
- Cross platform : available on Spark, PySpark, and SparklyR

Usages :

- Classifier
- Regressor including quantile regression
- Ranker

ONNX, what's that ?!?

- **Open Neural Network Exchange**
- ONNX is an open format built to represent Machine Learning models
  - Community project
  - Interoperability
  - Pre-trained models
    - https://github.com/onnx/models

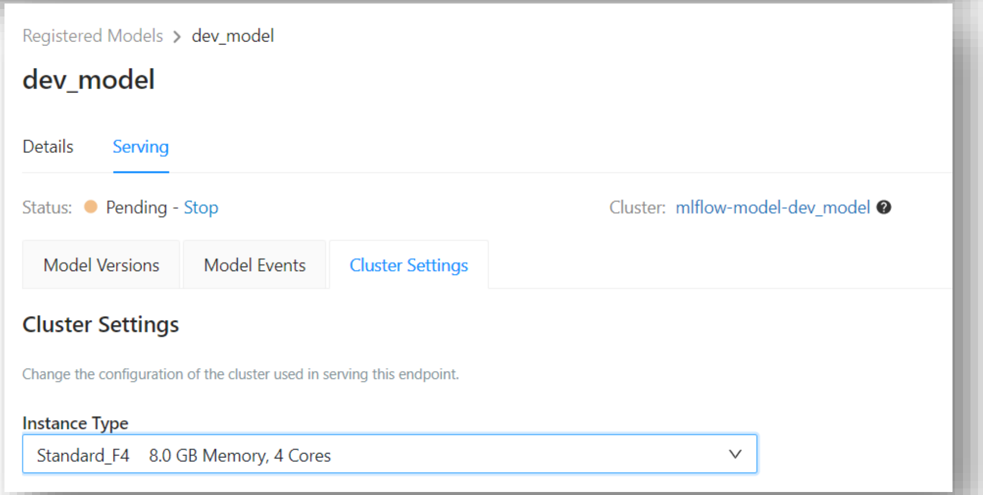
- Converting to ONNX format
  - http://onnx.ai/sklearn-onnx/index.html

Azure Databricks limitations

- MLFlow is too much integrated in the Databricks workspace
  - We don’t need a registry per environment
  - Stages of a model can’t be **inside** a workspace
  - We need an **external registry**

Name	Latest Version	Staging	Production
dev_model	Version 1	Version 1	–
housing_elastic	Version 1	–	–
housing_model	Version 6	Version 6	Version 5

- Serving
  - without monitoring
  - without advanced authentication
  - without configuration fine tuning
  - with DBU cost



Some conclusions

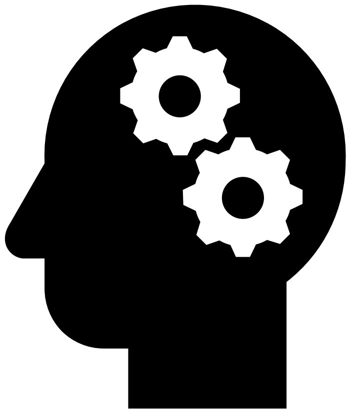
- At scale or in production, **don’t use every feature of Azure ML**
    - Forget about Datastore and Dataset

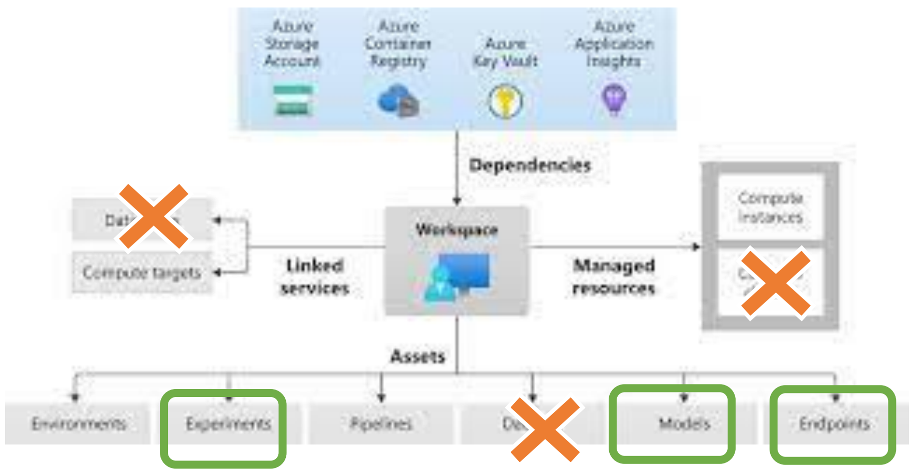
      Use Delta format in the Data Lake
    - Just explore data inside notebooks on compute instance

      Don’t forget refactoring and packaging of the code
    - Forget about compute cluster

      Can’t really distribute the code
    - Don’t create an Azure ML workspace per environment

- In a « MLOps stack », Azure ML must be « **central** »
    - Logs of experiments
    - Model Tracking
    - Model Registry
    - Serving : deploy predictive endpoints





Let’s talk about MLOps stack

What do we need to version in MLOps ?

- Classical development project

$$\{\text{infra as code, app code, CI/CD code}\} + \text{last data}$$

- Machine Learning project

$$\{\text{data, code, model+requirements, container}\}$$

MLOps : what we need to version

{ (raw / clean) data,

ML code + tests,

model + requirements,

CI/CD code,

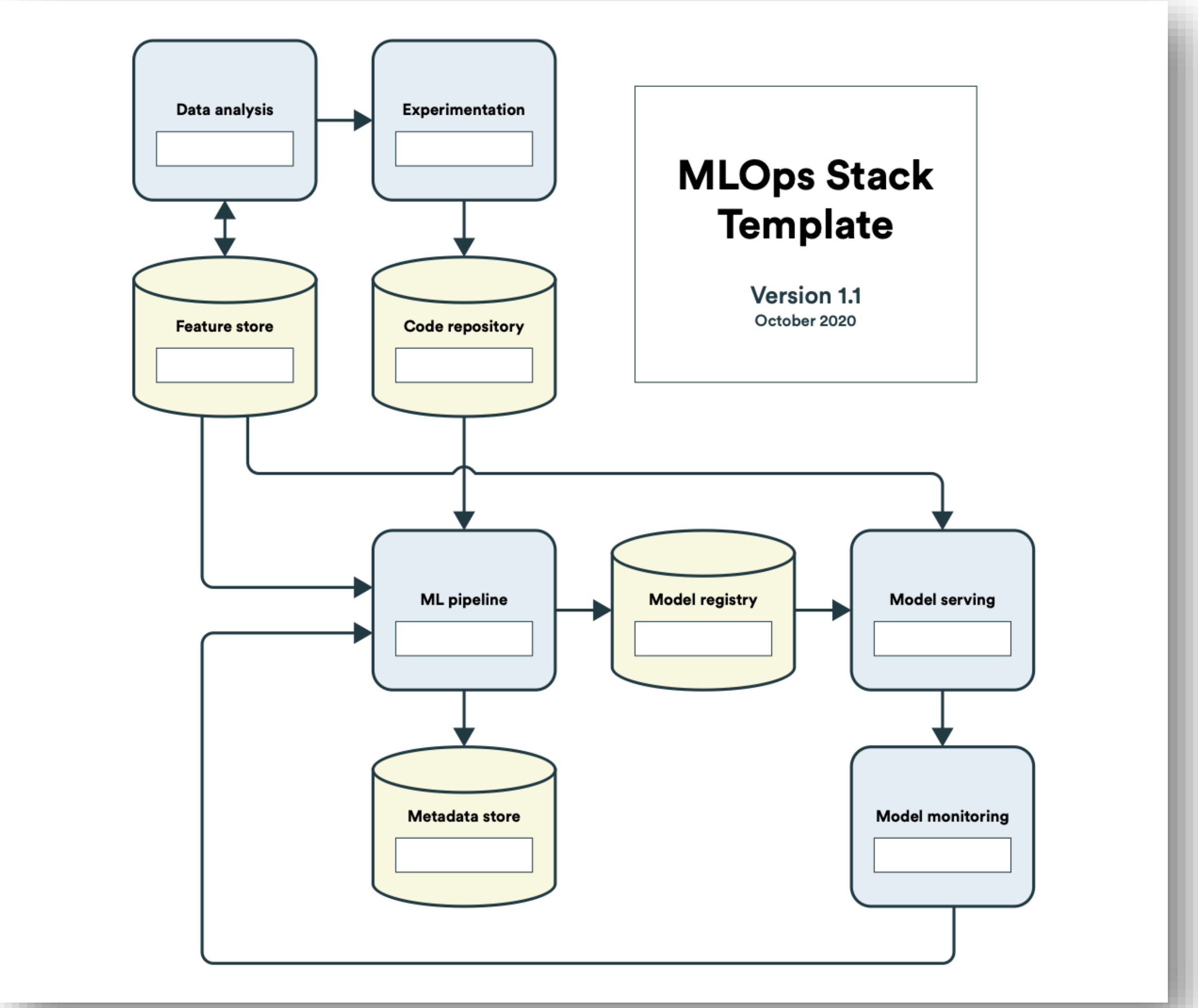
container}

ML-Ops.org - principles

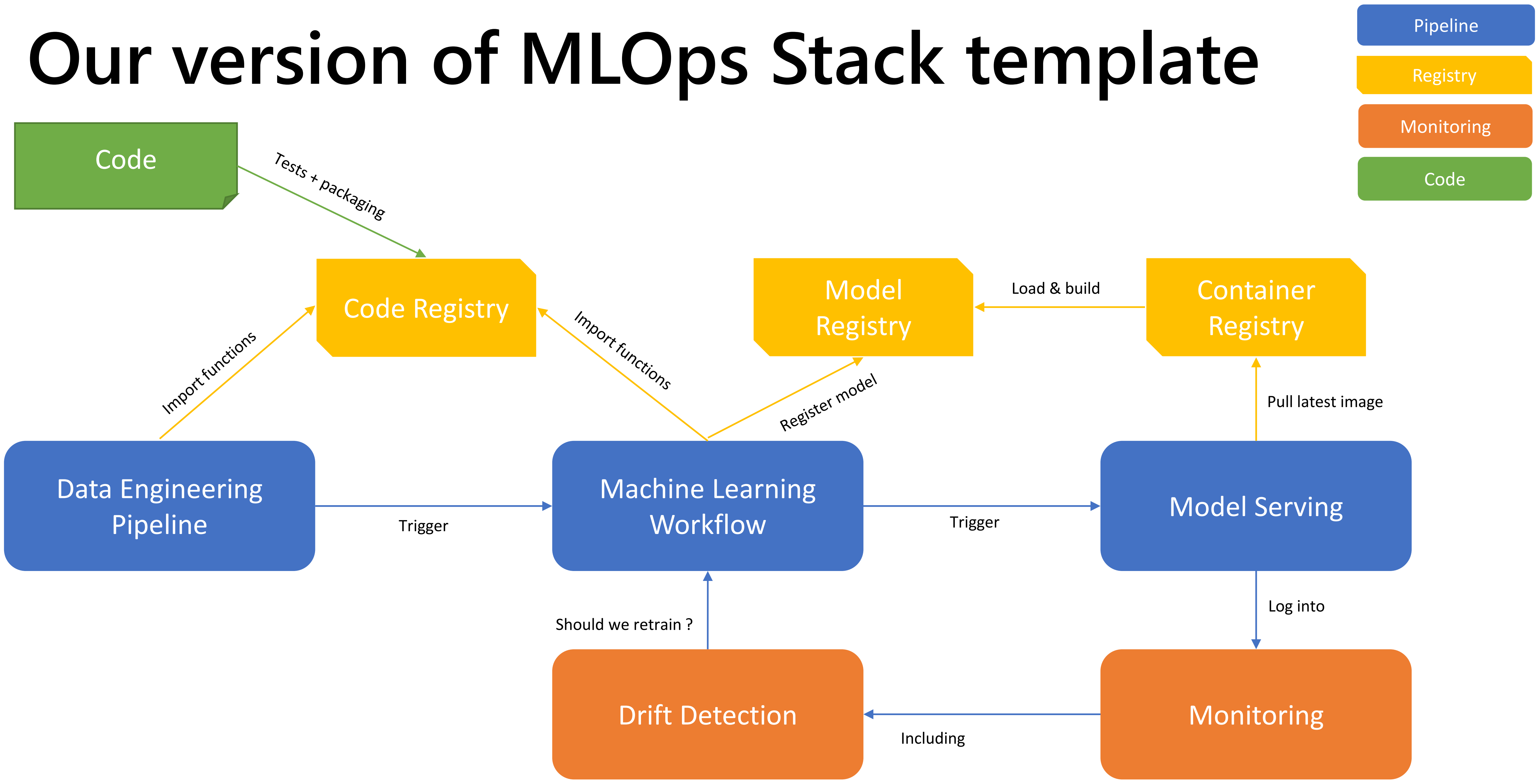
created by Dr. Larysa Visen2geriyeva, Anja Kammer, Isabel Bär, Alexander Kniesz, and Michael Plöd (DDD Advisor)

- The three main « pipelines » :
  - Data Engineering Pipeline
  - Machine Learning Pipeline / Workflow
  - Model Serving

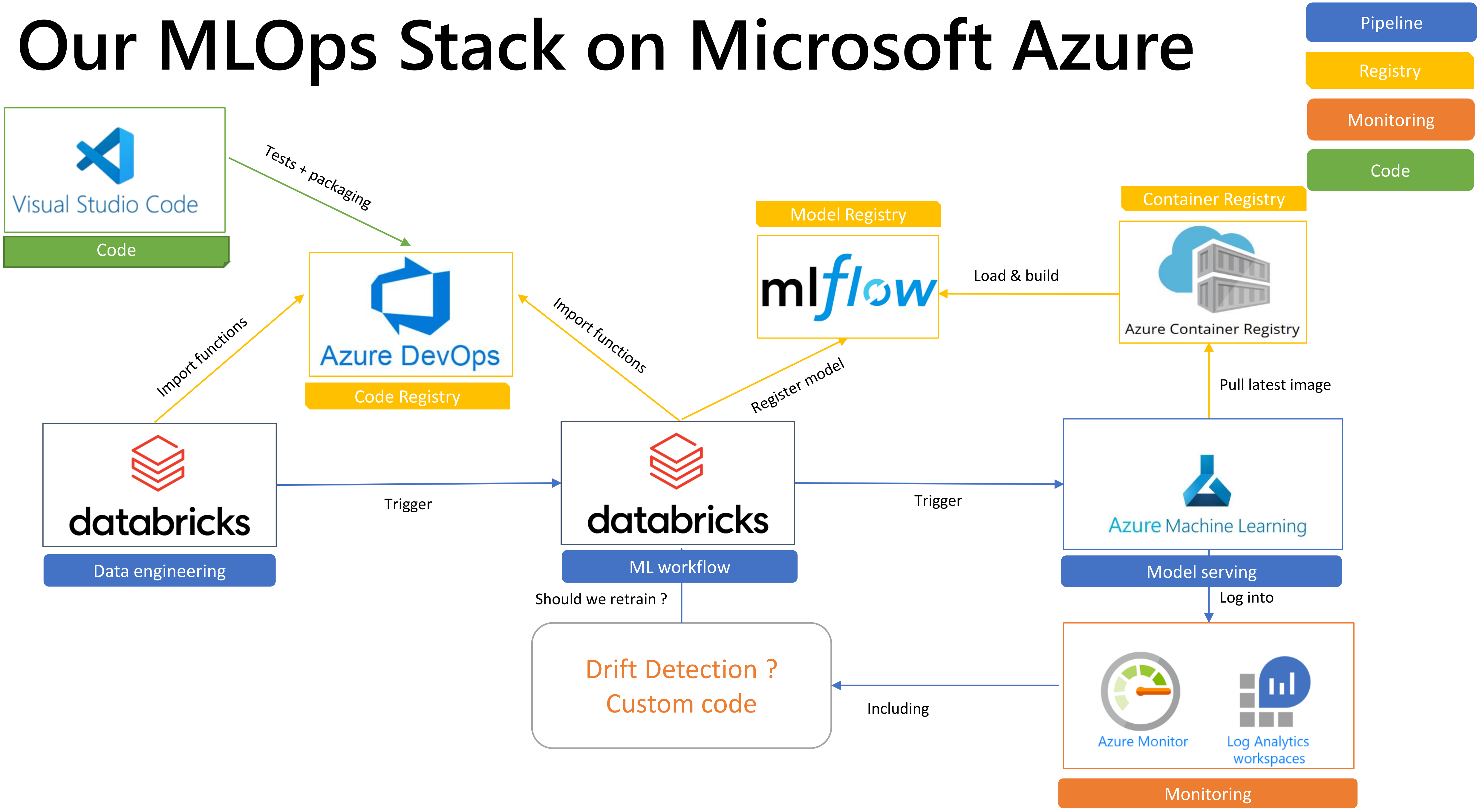
- The concept of « registry » :
  - Storage adapted to the type of deliverable
    - Code
    - Binary (pickle, H5, etc.)
    - Docker image
    - etc.
  - Allows versioning of objects
  - Ensures reproducibility



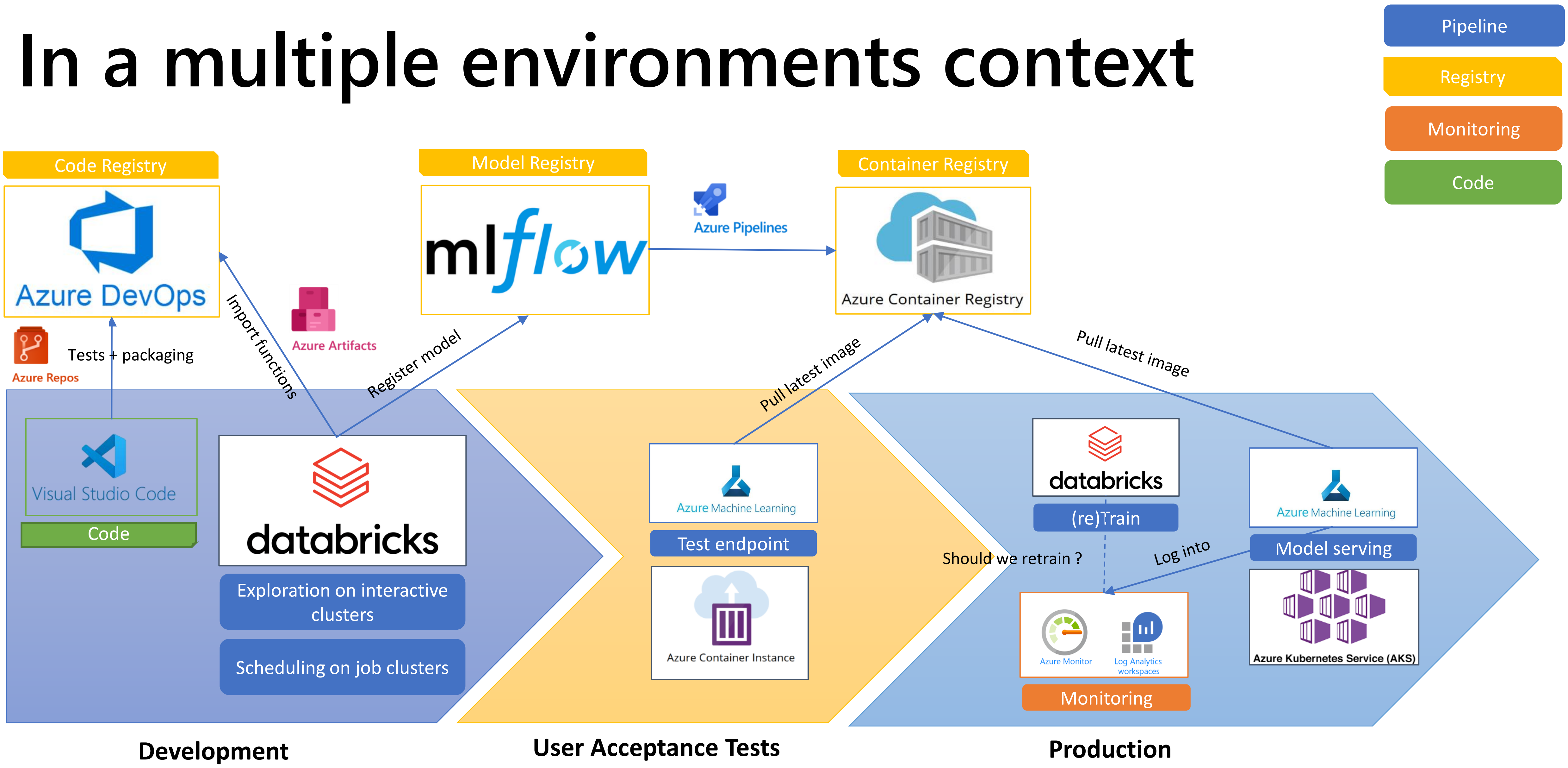
Our version of MLOps Stack template



Our MLOps Stack on Microsoft Azure



In a multiple environments context



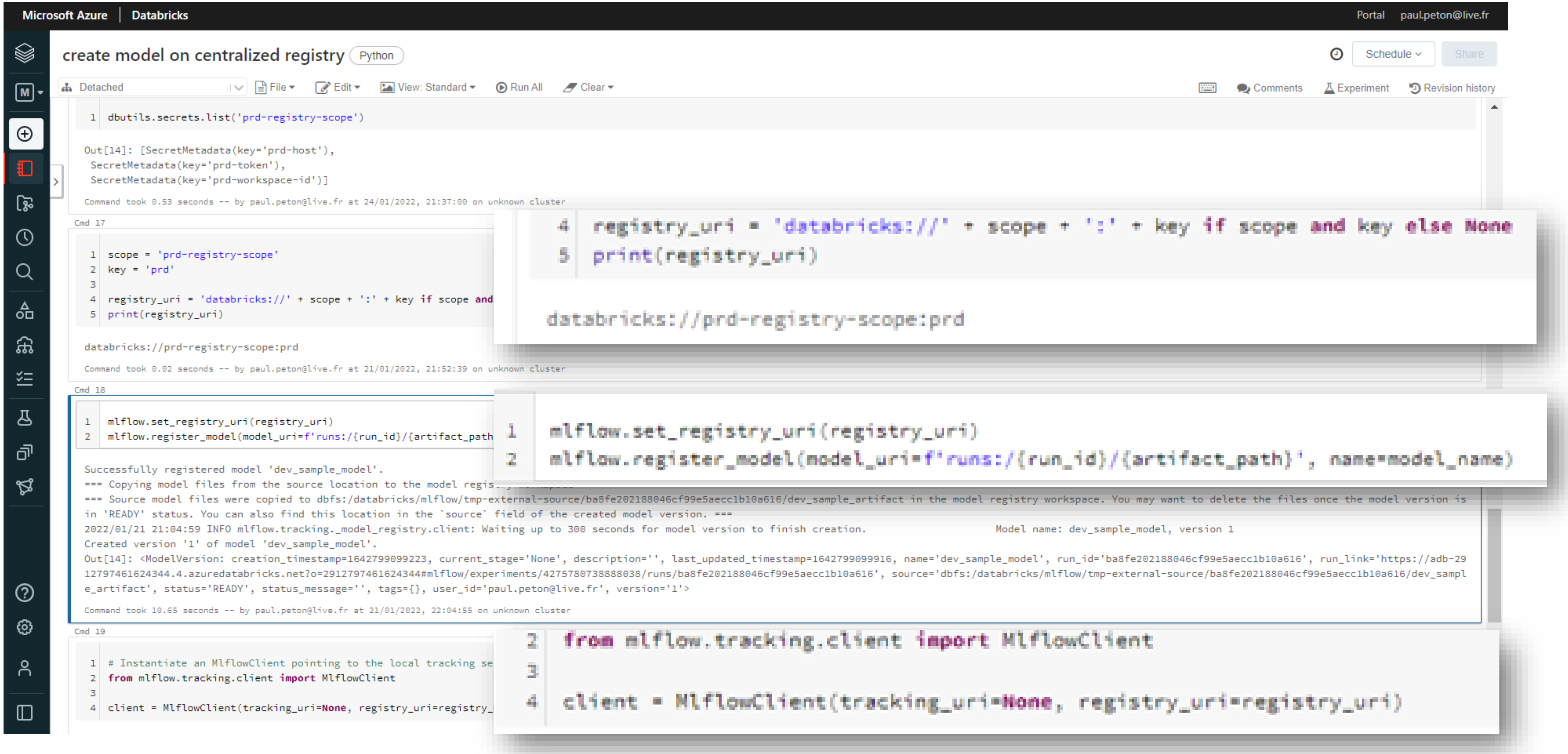
Azure ML + Databricks

1. Authentication
2. MLFlow tracking URI
3. Attached compute

How to authenticate with a SPN

```
from azureml.core.authentication import ServicePrincipalAuthentication

tenant_id = '[redacted]'
service_principal_id = '[redacted]'

subscription_id = '[redacted]'
resource_group = 'rg-centralized-registry'
workspace_name = 'centralizedaml'

svc_pr = ServicePrincipalAuthentication(
   tenant_id=tenant_id,
   service_principal_id=service_principal_id,
   service_principal_password=svc_pr_password)


ws = Workspace(
   subscription_id=subscription_id,
   resource_group=resource_group,
   workspace_name=workspace_name,
   auth=svc_pr
   )

print("Found workspace {} at location {}".format(ws.name, ws.location))
```

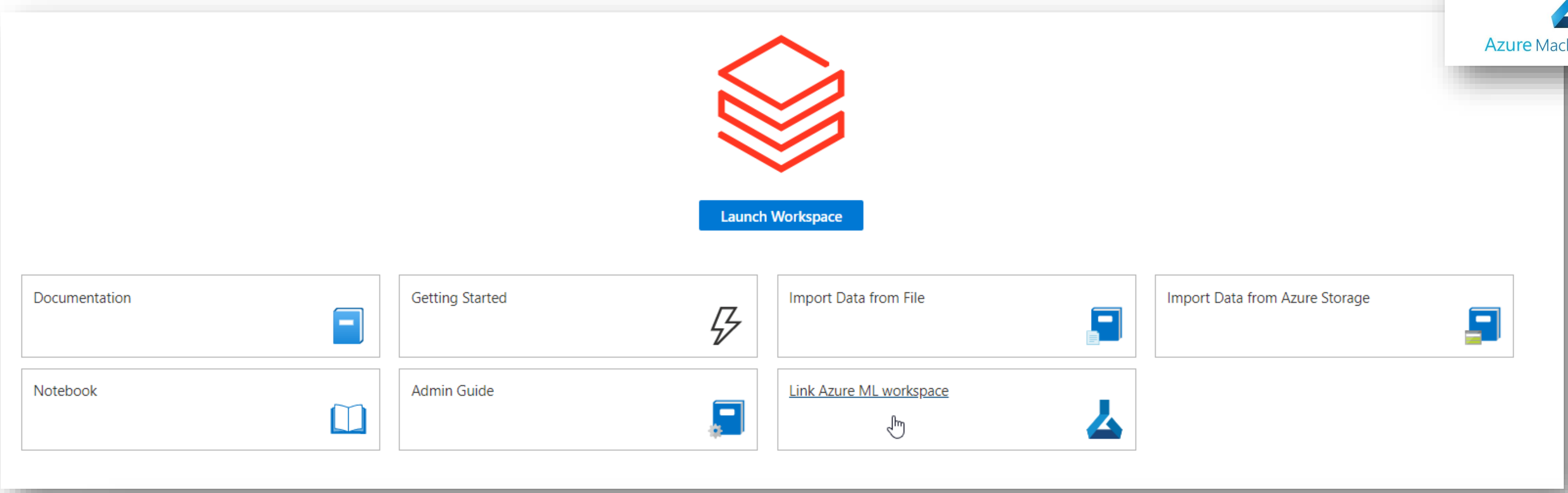
MLFlow : specify the tracking URI

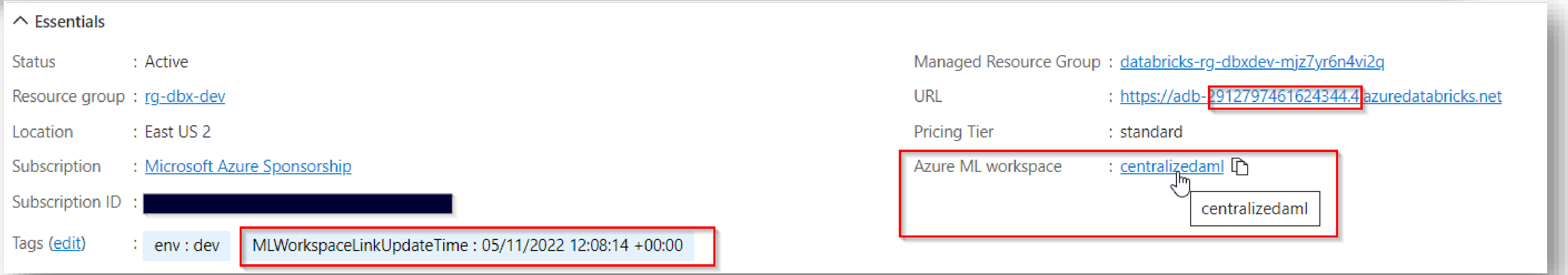


Databricks : link Azure ML workspace

databricks

Azure Machine Learning





Use the tracking URI of Azure ML

databricks

Azure Machine Learning

Tracking URI : Azure ML

Cmd 76

```
azureml_tracking_uri = ws.get_mlflow_tracking_uri()
print(f"Azure ML Tracking URI: {azureml_tracking_uri}")

mlflow.set_tracking_uri(azureml_tracking_uri)
print("MLflow Tracking URI:", mlflow.get_tracking_uri())

client = mlflow.tracking.MlflowClient()
```

```
Azure ML Tracking URI: azureml://eastus2.api.azureml.ms/mlflow/v1.0/subscriptions/f80606e5-788f-4dc3-a9ea-2eb9a7836082/resourceGroups/rg-centralized-registry/providers/Microsoft.MachineLearningServices/workspaces/centralizedaml?
MLflow Tracking URI: azureml://eastus2.api.azureml.ms/mlflow/v1.0/subscriptions/f80606e5-788f-4dc3-a9ea-2eb9a7836082/resourceGroups/rg-centralized-registry/providers/Microsoft.MachineLearningServices/workspaces/centralizedaml?
```

Command took 0.10 seconds -- by paul.peton@live.fr at 31/05/2022, 13:35:40 on cluster-dev

Cmd 77

```
experiment_name = 'Yellow_Taxi_fromDBX_experiment'
mlflow.set_experiment(experiment_name)
```

```
2022/05/31 11:36:27 INFO mlflow.tracking.fluent: Experiment with name 'Yellow_Taxi_fromDBX_experiment' does not exist. Creating a new experiment.
Out[182]: <Experiment: artifact_location='', experiment_id='d140cd27-08d8-4638-ab62-bf60ca444378', lifecycle_stage='active', name='Yellow_Taxi_fromDBX_experiment', tags={}>
```

Command took 1.04 seconds -- by paul.peton@live.fr at 31/05/2022, 13:36:27 on cluster-dev

Attached compute for Azure ML

Azure Machine Learning

databricks

- Azure Databricks
- Data Lake Analytics
- HDInsight
- Kubernetes (preview)
- Synapse Spark pool (preview)
- Virtual machine

Attach your compute target and manage your platform using Azure Machine Learning

Bring your own compute like an HDInsight cluster, a Virtual Machine, or a Databricks cluster to use as a compute target with your Azure Machine Learning workspace. Learn more

+ New

Attach Databricks compute

Compute name *

Subscription *

Microsoft Azure Sponsorship

Refresh subscriptions

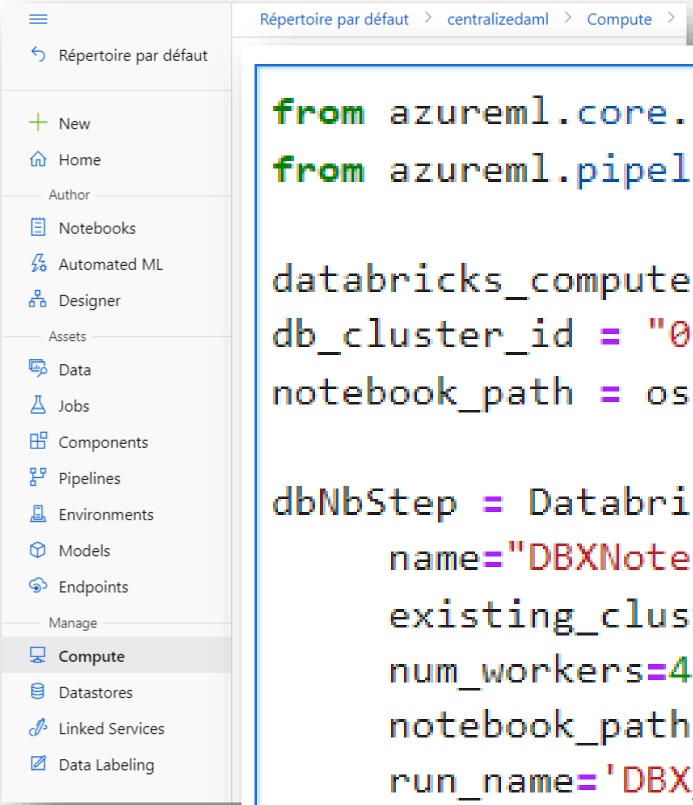
Databricks workspace *

Search or select an Azure Databricks workspace

Refresh Databricks workspaces

Databricks access token *

Launch a Pipeline step on Databricks




```python
from azureml.core.compute import DatabricksCompute
from azureml.pipeline.steps import DatabricksStep

databricks_compute = DatabricksCompute(workspace=ws, name="databricks-prod")
db_cluster_id = "0530-190209-eievpfbj"
notebook_path = os.getenv("DATABRICKS_NOTEBOOK_PATH", "/Shared/hello")

dbNbStep = DatabricksStep(
     name="DBXNotebookInWS",
     existing_cluster_id=db_cluster_id,
     num_workers=4,
     notebook_path=notebook_path,
     run_name='DBX_notebook_run',
     compute_target=databricks_compute,
     allow_reuse=False
)
```

```python
steps = [dbNbStep]
```

```python
from azureml.pipeline.core import Pipeline


pipeline = Pipeline(workspace = ws, steps = [steps])
print('Pipeline is built')
pipeline.validate()
print('Pipeline validation complete')
pipeline_run = exp.submit(pipeline)
print('Pipeline is submitted for execution')

pipeline_run.wait_for_completion(show_output = False)
```

Azure Synapse Analytics

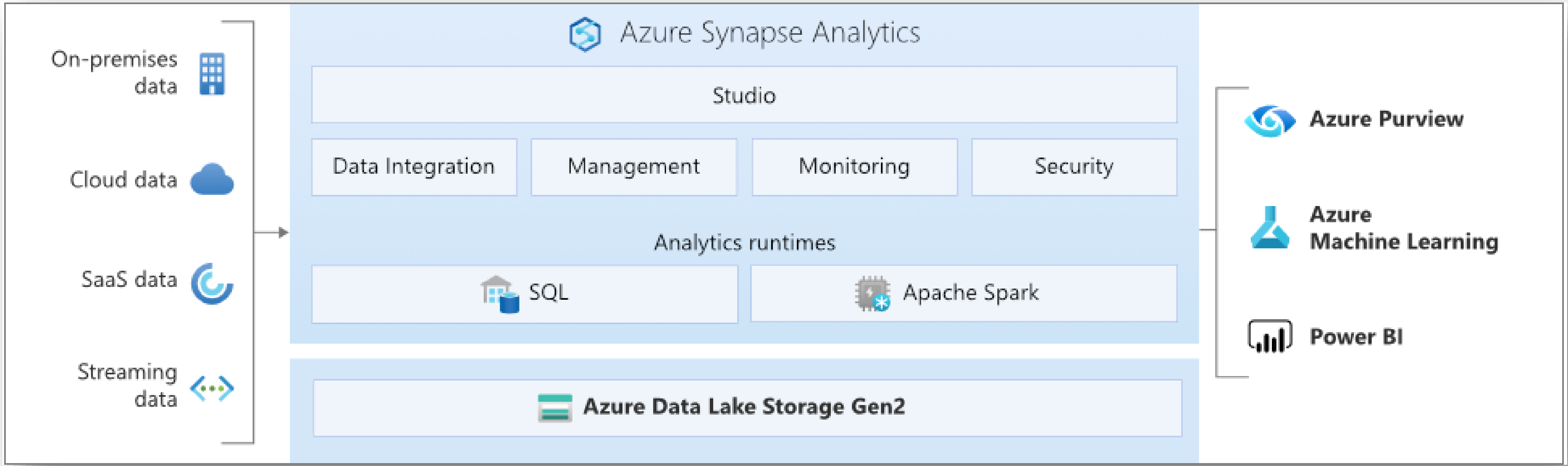
Azure Machine Learning

And what about Synapse Analytics + Azure ML ?

Azure Synapse Analytics in a few words

Azure Synapse Analytics

The data toolbox **Unified experience** for data project on Azure



Azure Synapse brings together the best of **SQL** technologies used in data warehousing :

Spark technologies used for big data

Pipelines for data integration and ETL/ELT

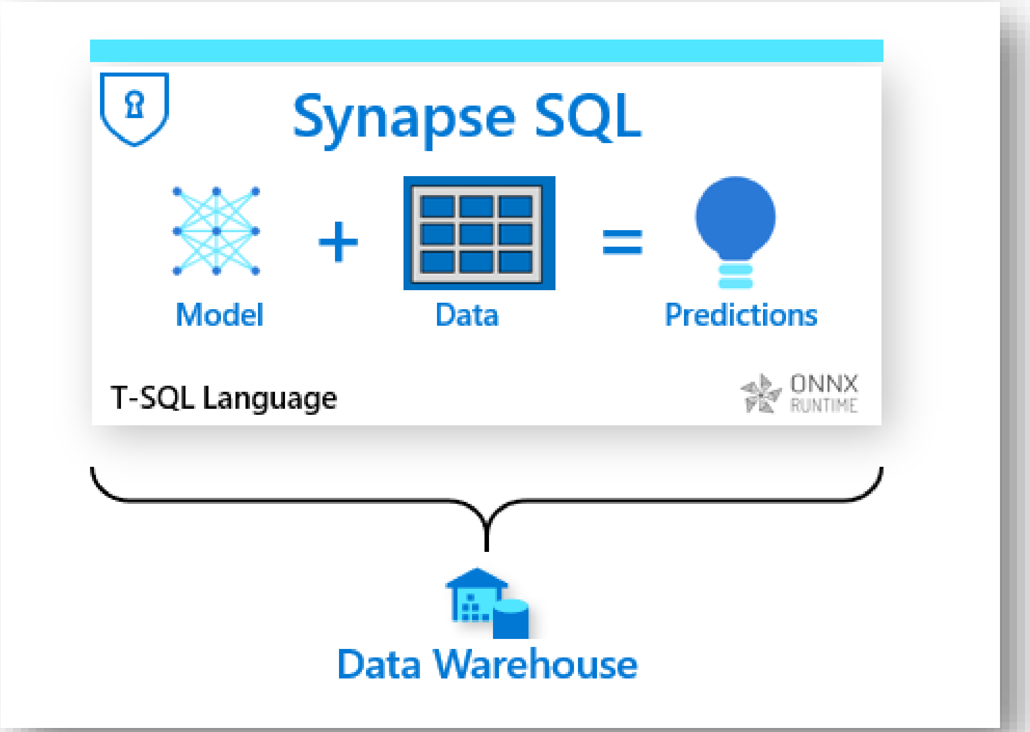
deep integration with other Azure services such as **Power BI**, **CosmosDB**, and **AzureML**.

Use ONNX model in Synapse

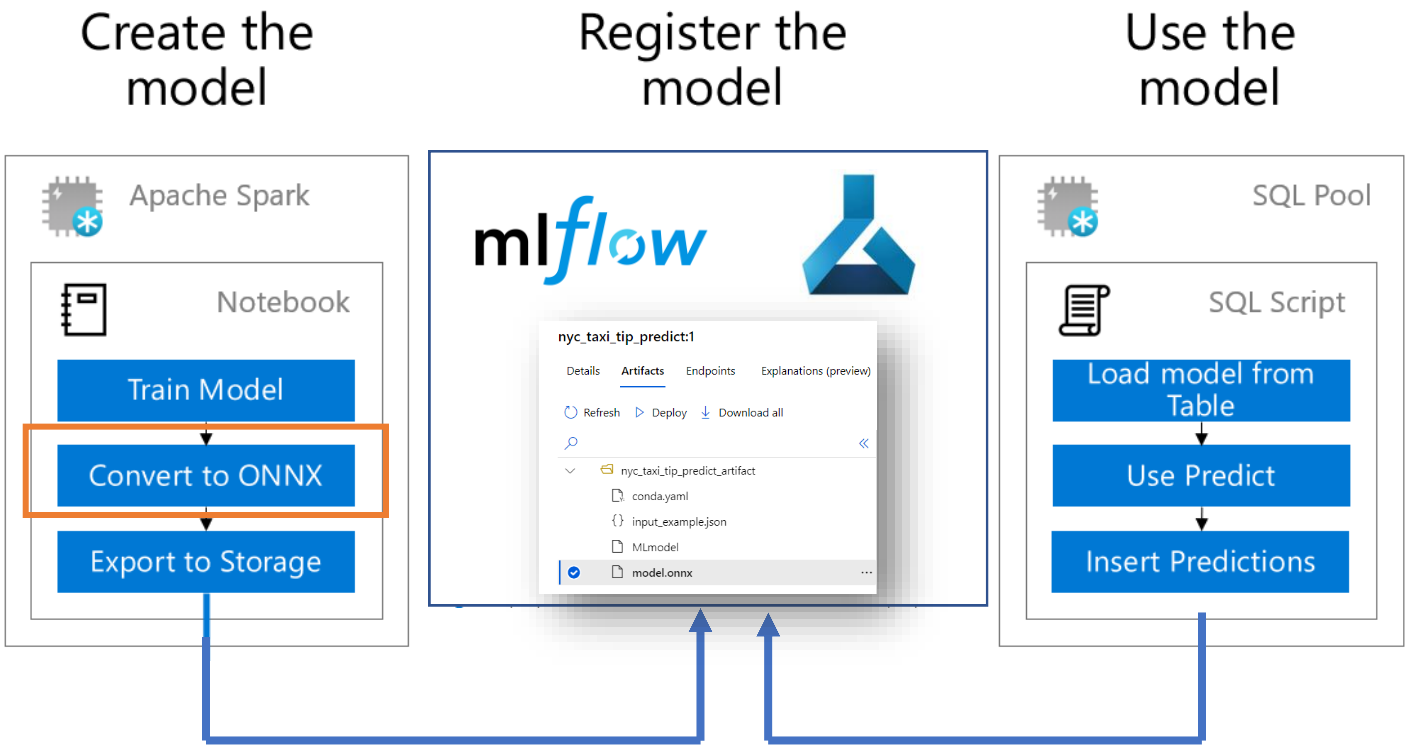
- Dedicated SQL pool expects a **pre-trained** model
  - only supports **ONNX** format models
  - The scoring data needs to be in the **same format** as the training data
  - Make sure that the **names** and **data types** of the model inputs match the column names and data types of the new prediction data.

- The model is stored in a dedicated SQL pool user table as a **hexadecimal string**.

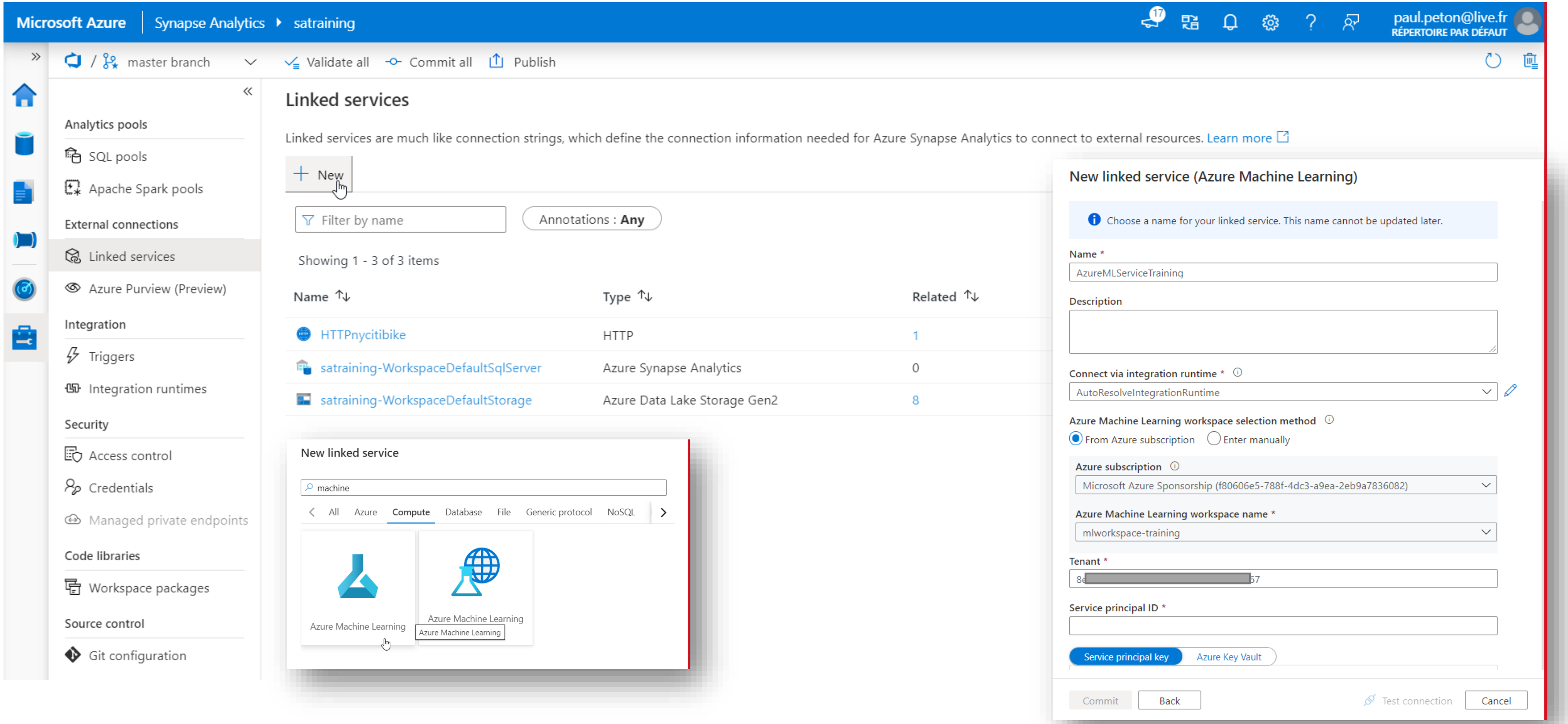
- Use the **T-SQL PREDICT** function to score the model



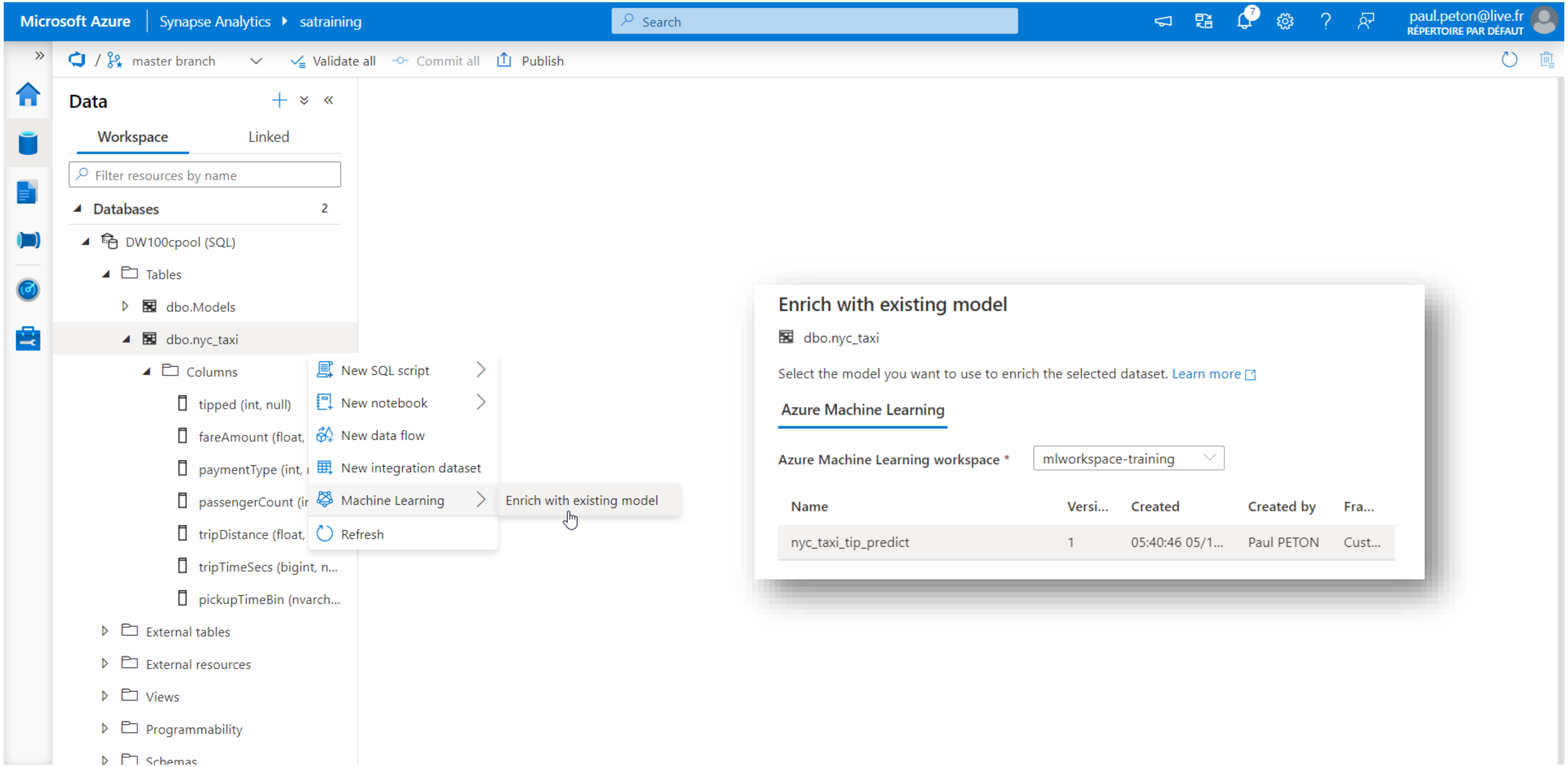
Machine Learning workflow in Synapse Analytics



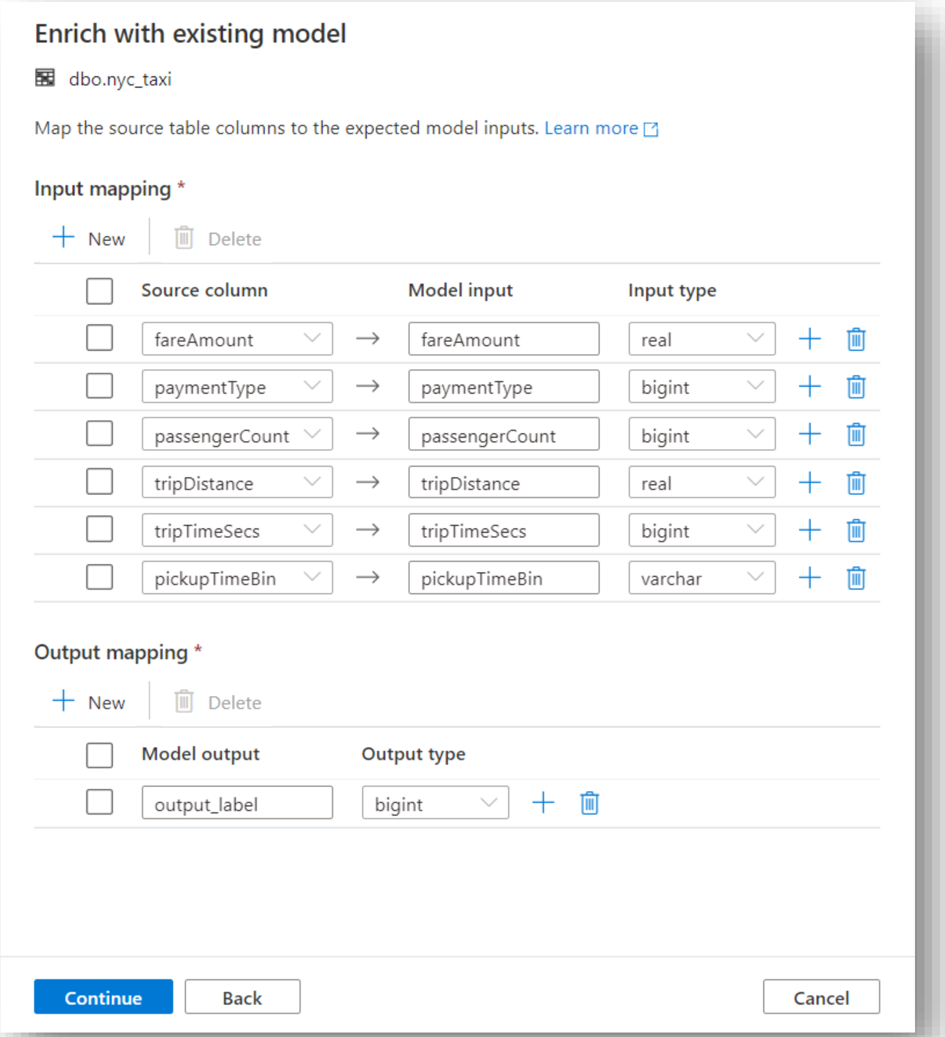
Linked services : from Synapse to Azure ML

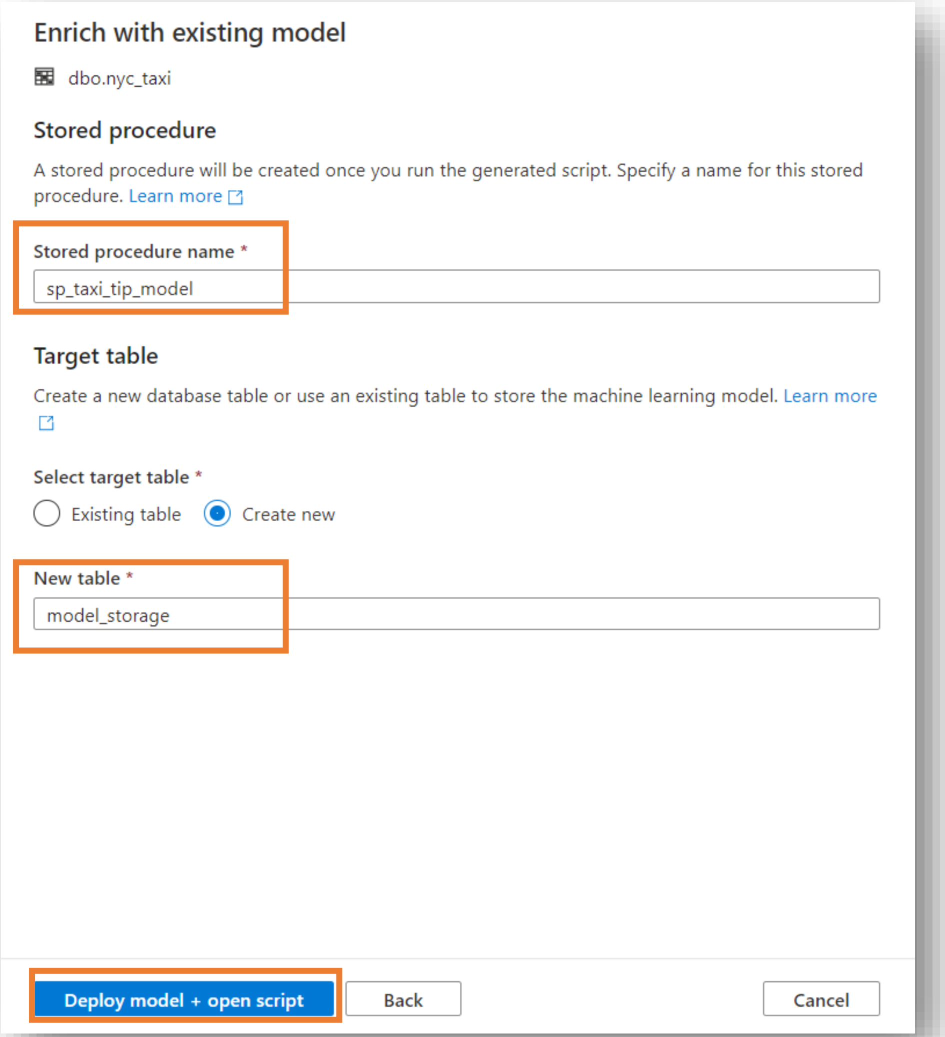


SQL Pool: enrich table with ML model



SQL Pool: Enrich table with existing model





Script: stored procedure

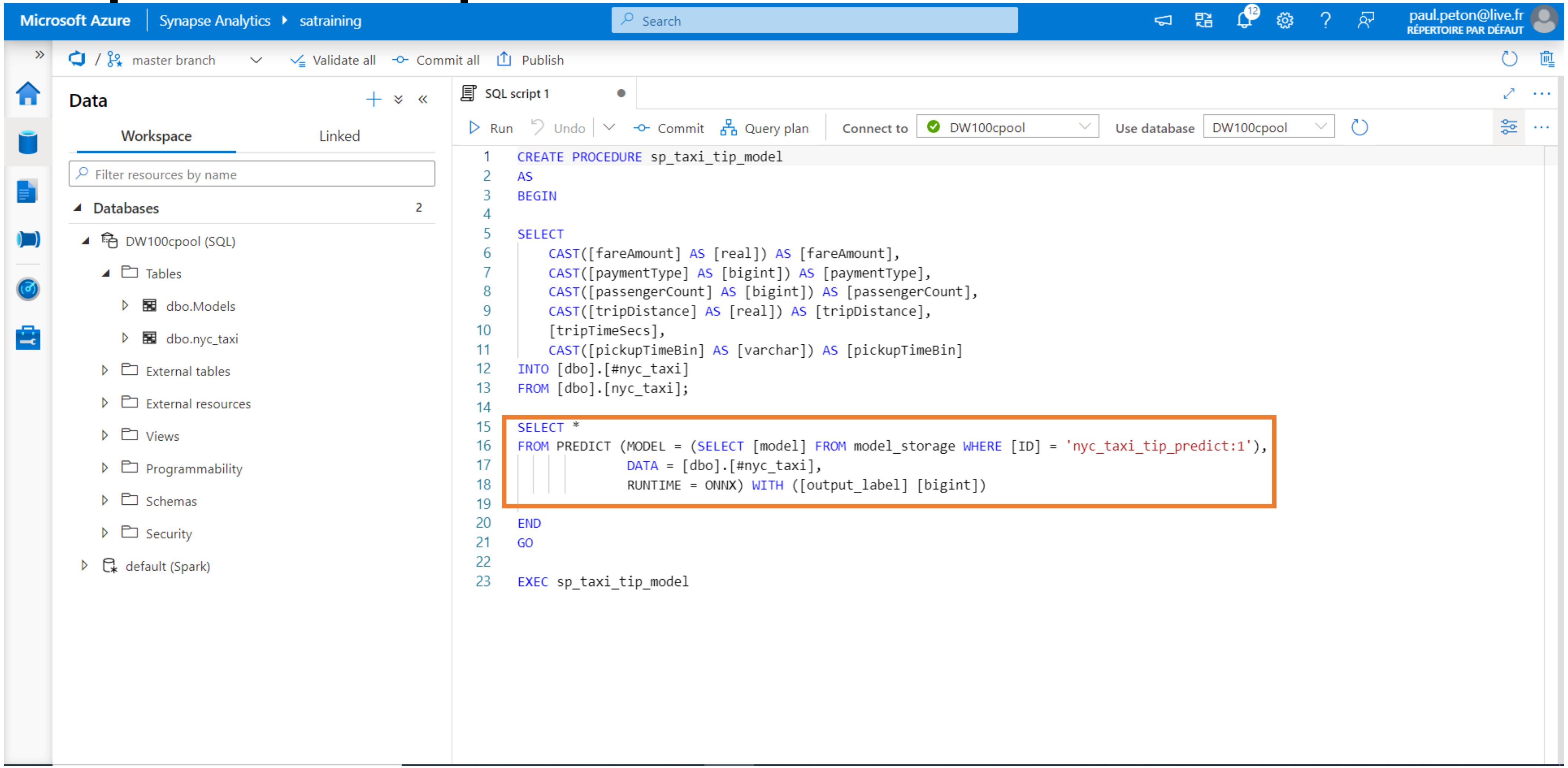
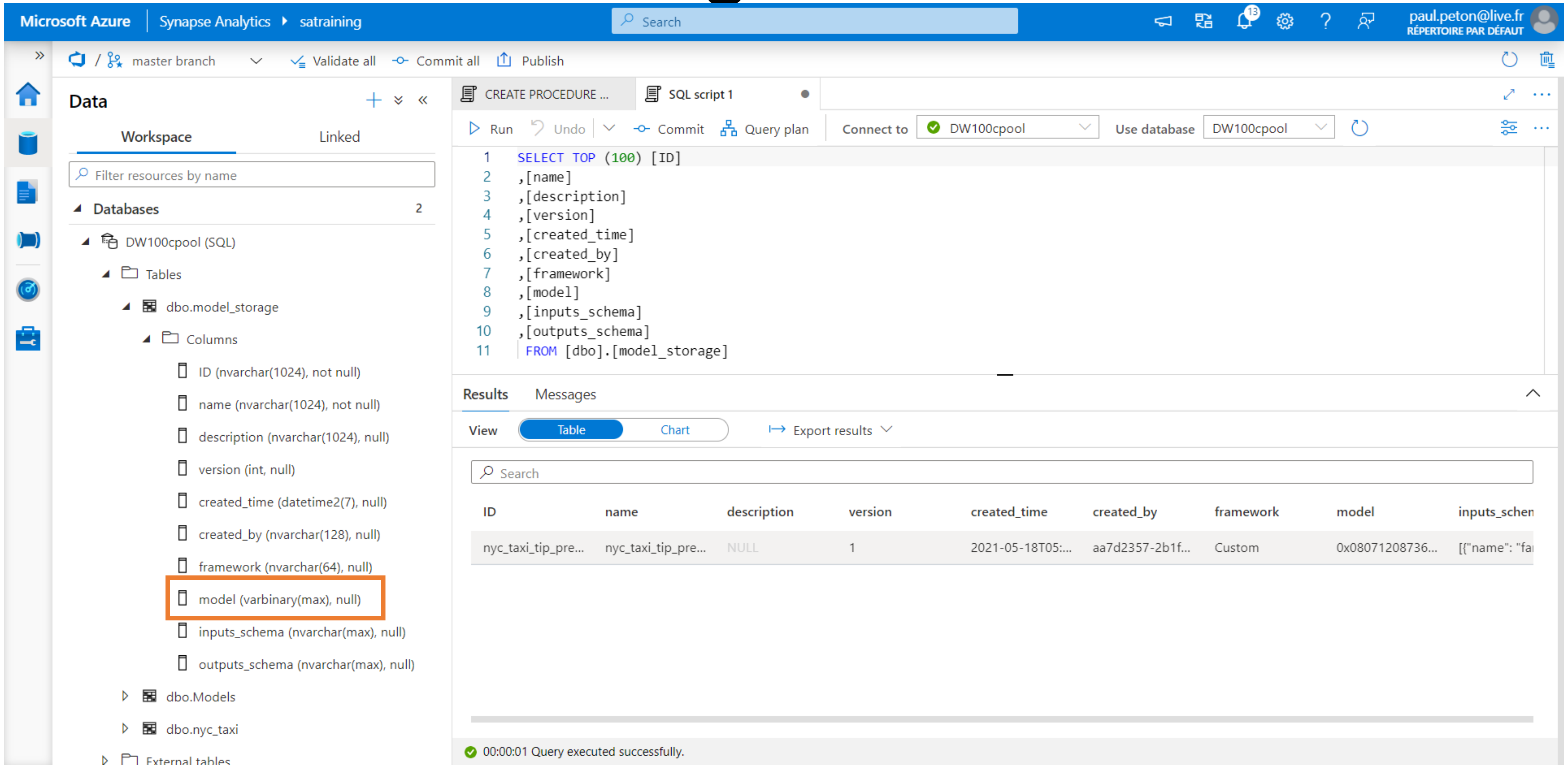
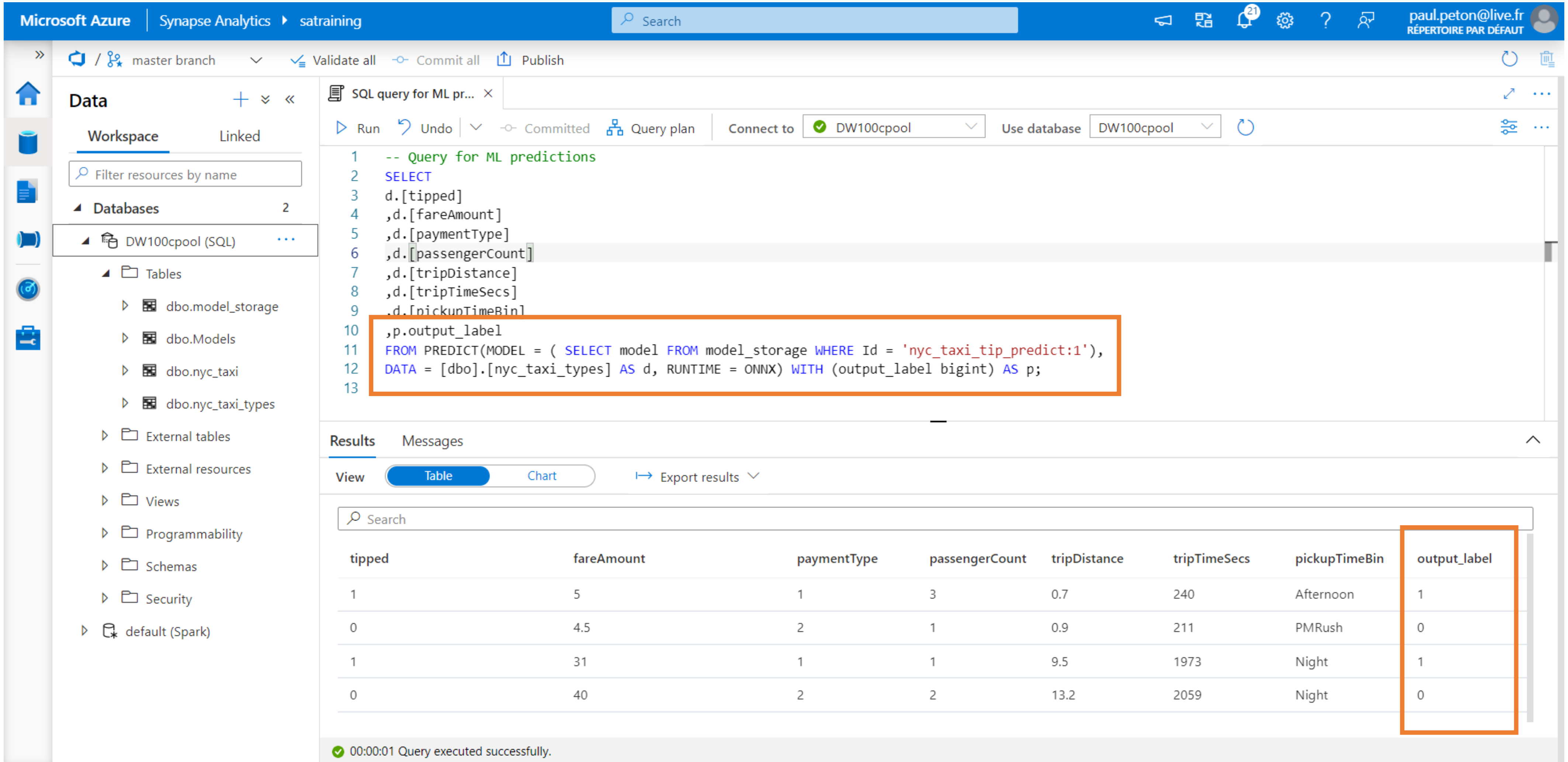


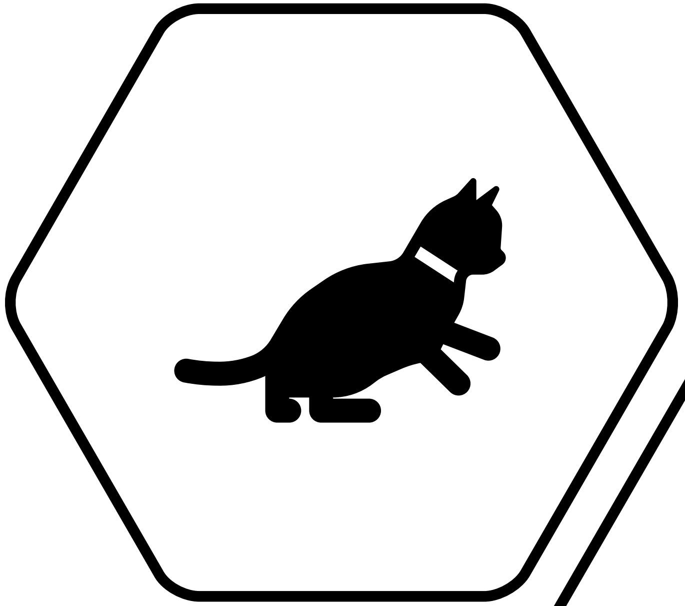
Table for model storage

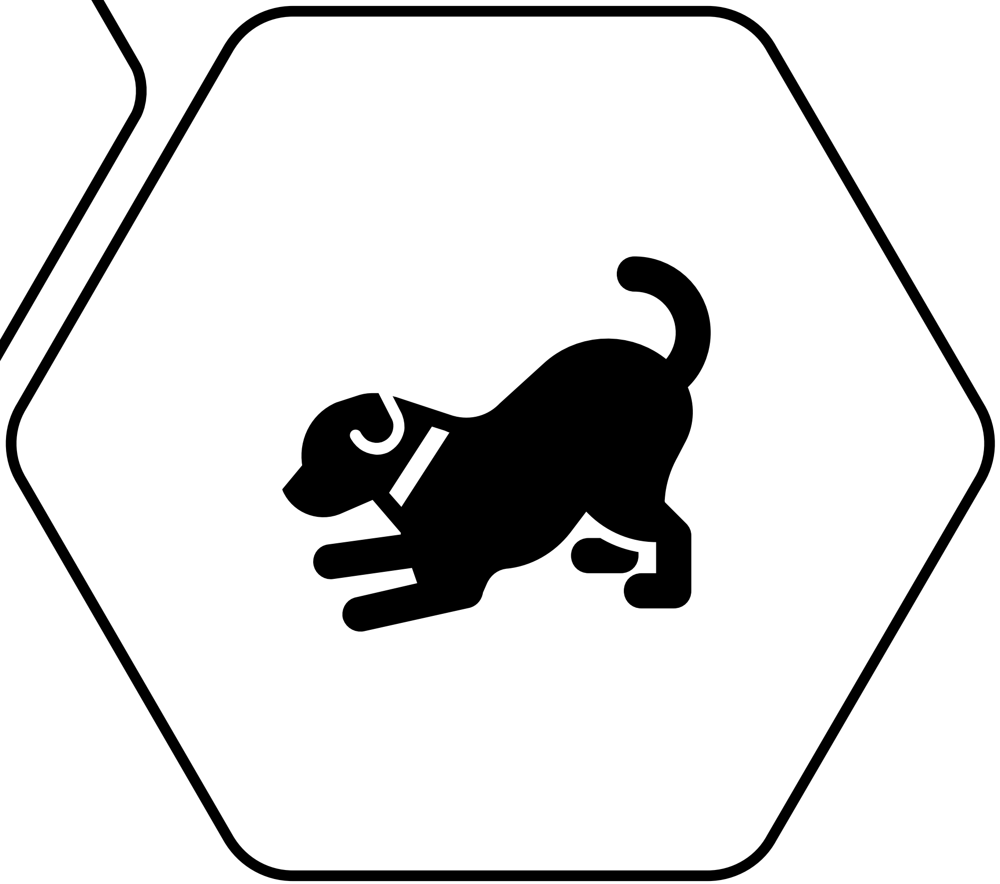


T-SQL PREDICT function



So, friend or foe ?





The main differences

Dedicated to Machine Learning

- By and for **Data Scientists**
  - Explore on compute instances
  - Simple batch ou real time scenarios
- Guided for the industrialization of their models and pipelines
  - Data versioning (?)
  - Model versioning
  - Web service

Dedicated to Big Data Science

- Work with big volume
- Version data with Delta format
- Use Spark to benefit of “scale out” power

The main differences

Azure Machine Learning

Azure Synapse Analytics

Dedicated to Machine Learning

- By and for **Data Scientists**
  - Explore on compute instances
  - Simple batch ou real time scenarios
- Guided for the industrialization of their models and pipelines
  - Data versioning (?)
  - Model versioning
  - Web service

Dedicated to Data Engineering

- Data preparation : ETL / ELT
- Orchestration
- Dedicated SQL pool for table storage
- Serverless SQL pool for exploration

#finops : some tips to save money in Azure

- Azure Machine Learning
  - Schedule stop on compute instances
  - Use compte cluster only for hyperparameter tuning or autoML
- Azure Databricks
  - Use Spot VM
  - Use Databricks Jobs instead of Databricks interactive clusters
- Azure Synapse Analytics
  - Use serverless SQL Pool to explore files with SQL