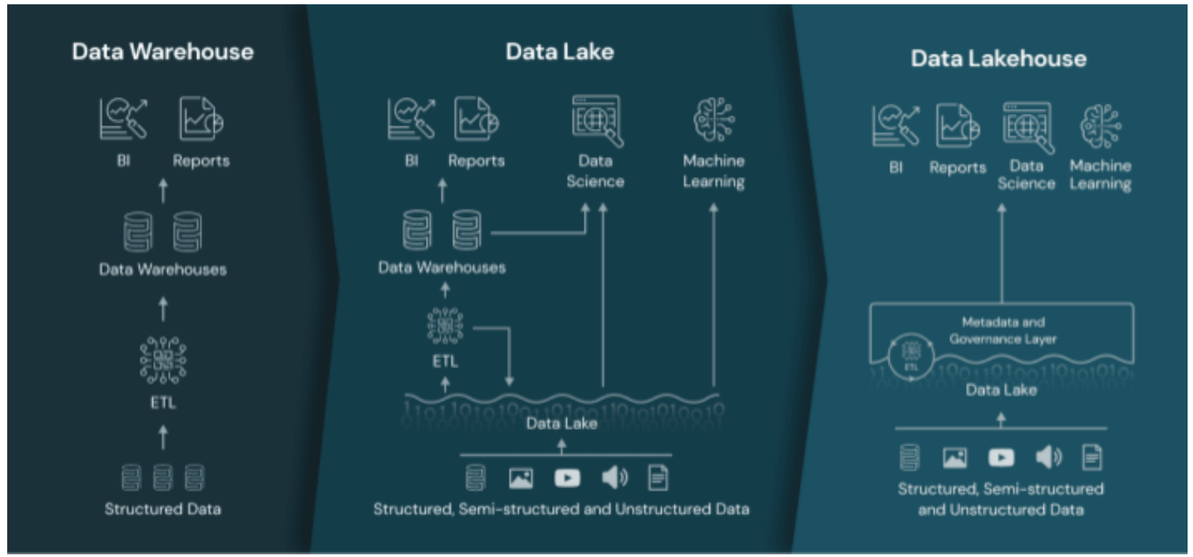
Overview of Lakehouse AI

DATABRICKS CONCEPTS

Kevin Barlow
Data Practitioner

datacamp

Lakehouse AI

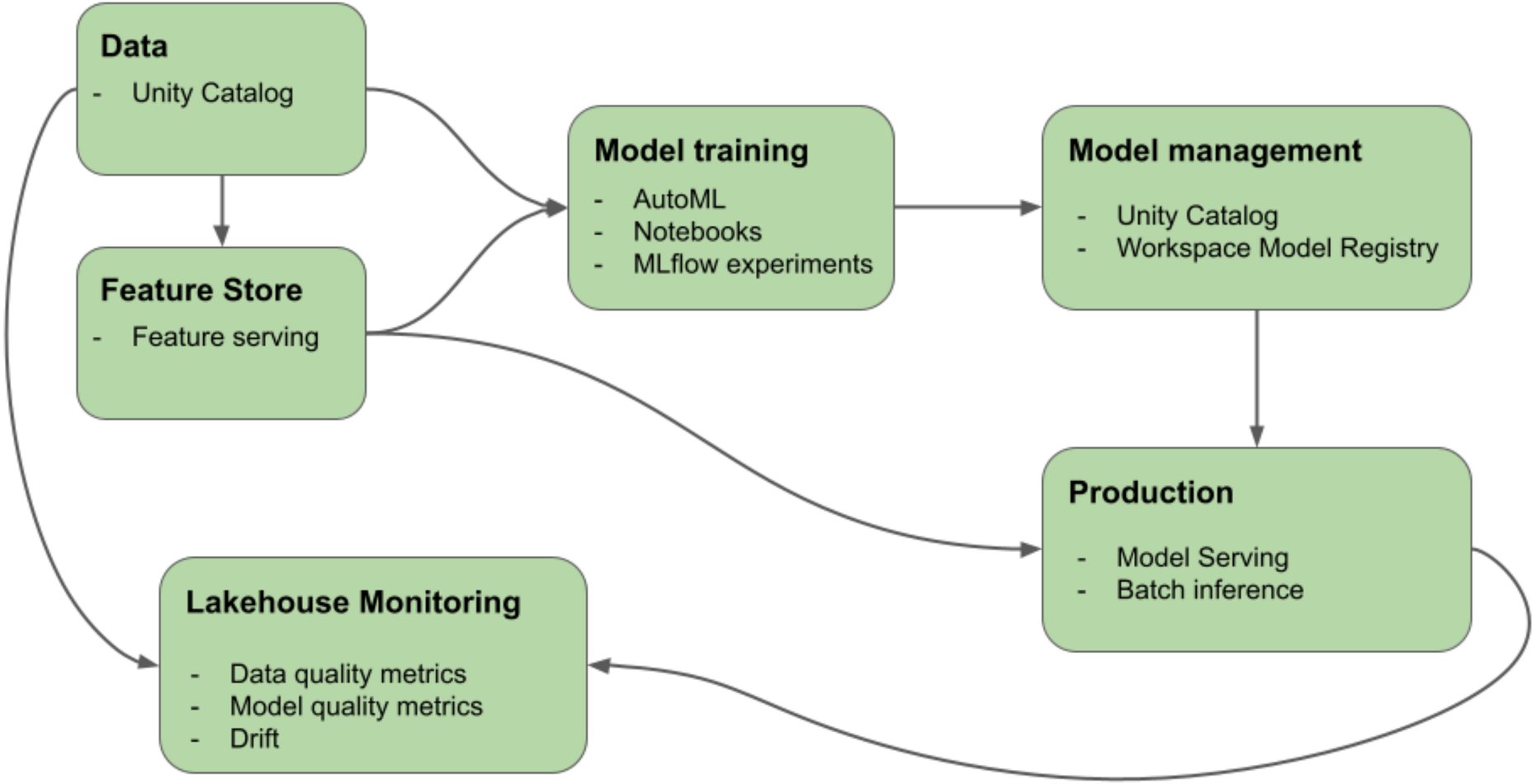


Why the Lakehouse for AI / ML?

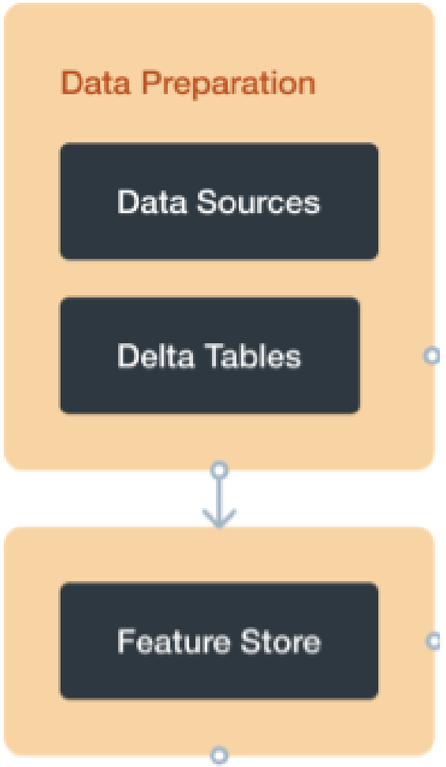
1. Reliable data and files in the Delta lake
2. Highly scalable compute
3. Open standards, libraries, frameworks
4. Unification with other data teams

[1] https://www.databricks.com/blog/2020/01/30/what-is-a-data-lakehouse.html

MLOps Lifecycle



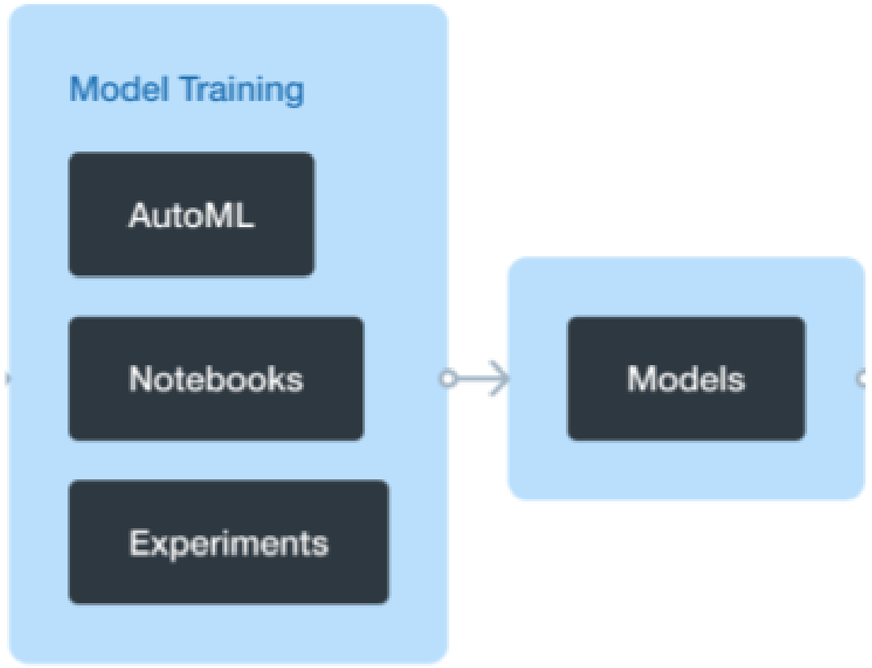
MLOps in the Lakehouse



DataOps

- Integrating data across different sources (*AutoLoader*)
- Transforming data into a usable, clean format (*Delta Live Tables*)
- Creating useful features for models (*Feature Store*)

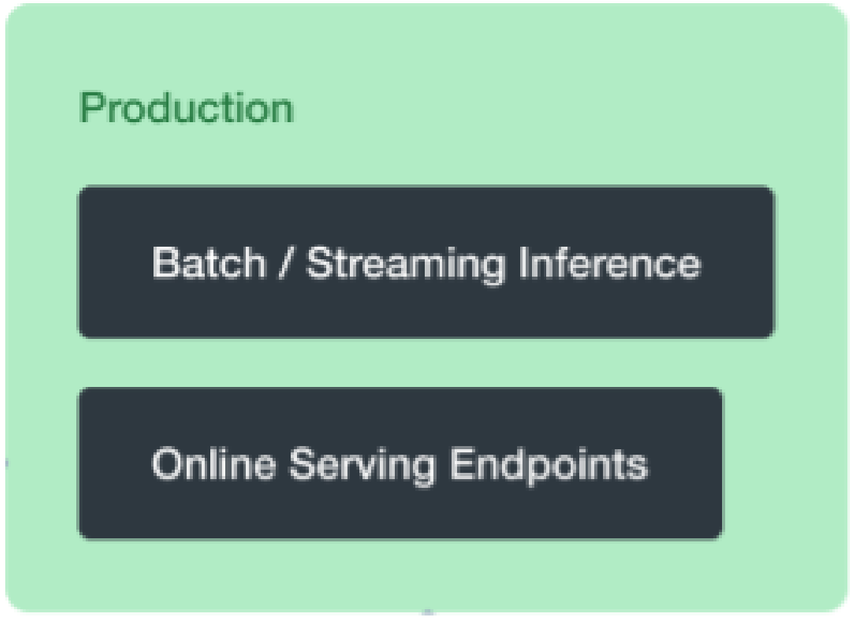
MLOps in the Lakehouse



ModelOps

- Develop and train different models (*Notebooks*)
- Machine learning templates and automation (*AutoML*)
- Track parameters, metrics, and trials (*MLFlow*)
- Centralize and consume models (*Model Registry*)

MLOps in the Lakehouse



DevOps

- Govern access to different models (*Unity Catalog*)
- Continuous Integration and Continuous Deployment (CI/CD) for model versions (*Model Registry*)
- Deploy models for consumption (*Serving Endpoints*)

Let's review!

DATABRICKS CONCEPTS

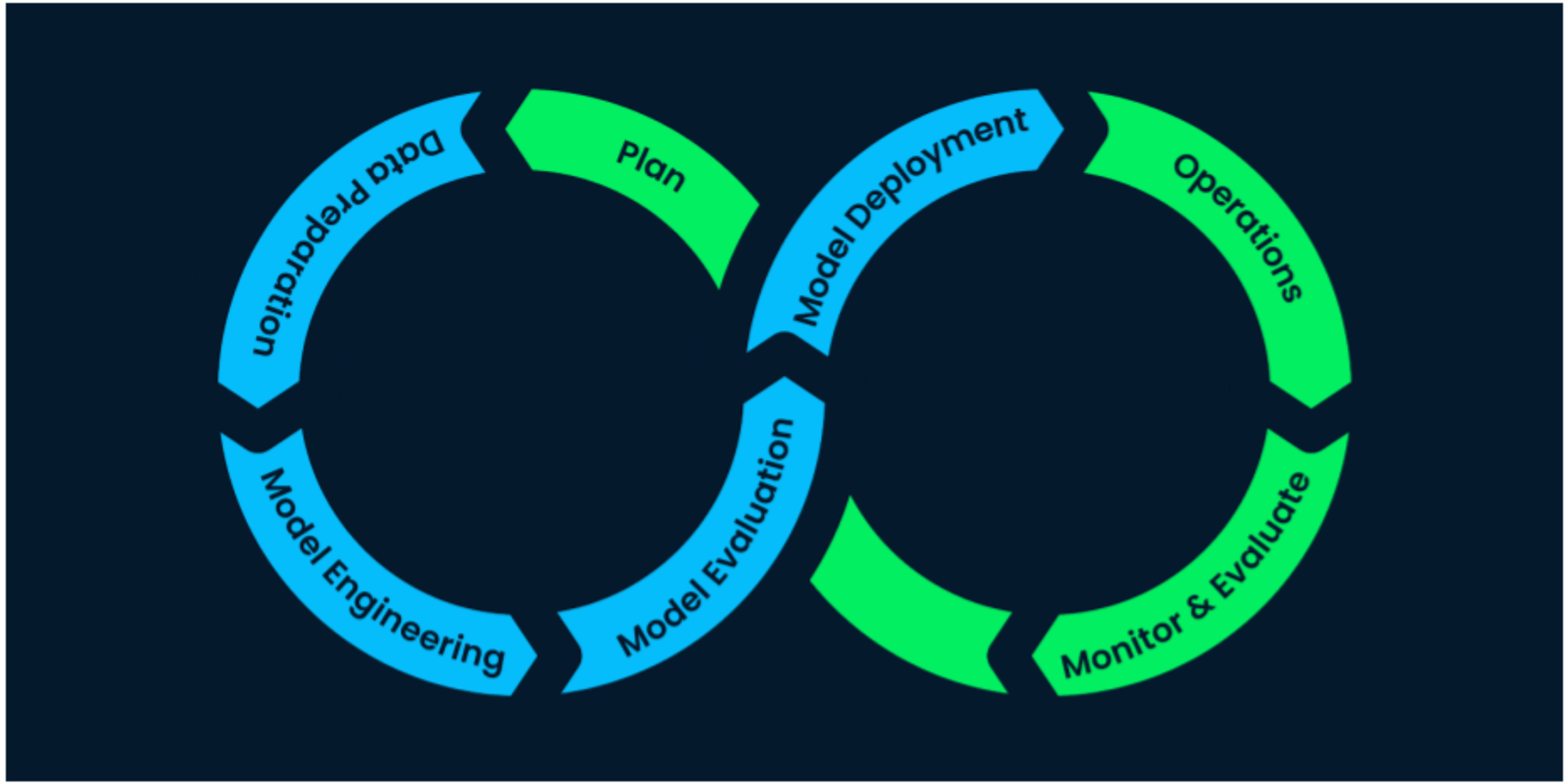
Using Databricks for machine learning

DATABRICKS CONCEPTS

Kevin Barlow
Data Practitioner

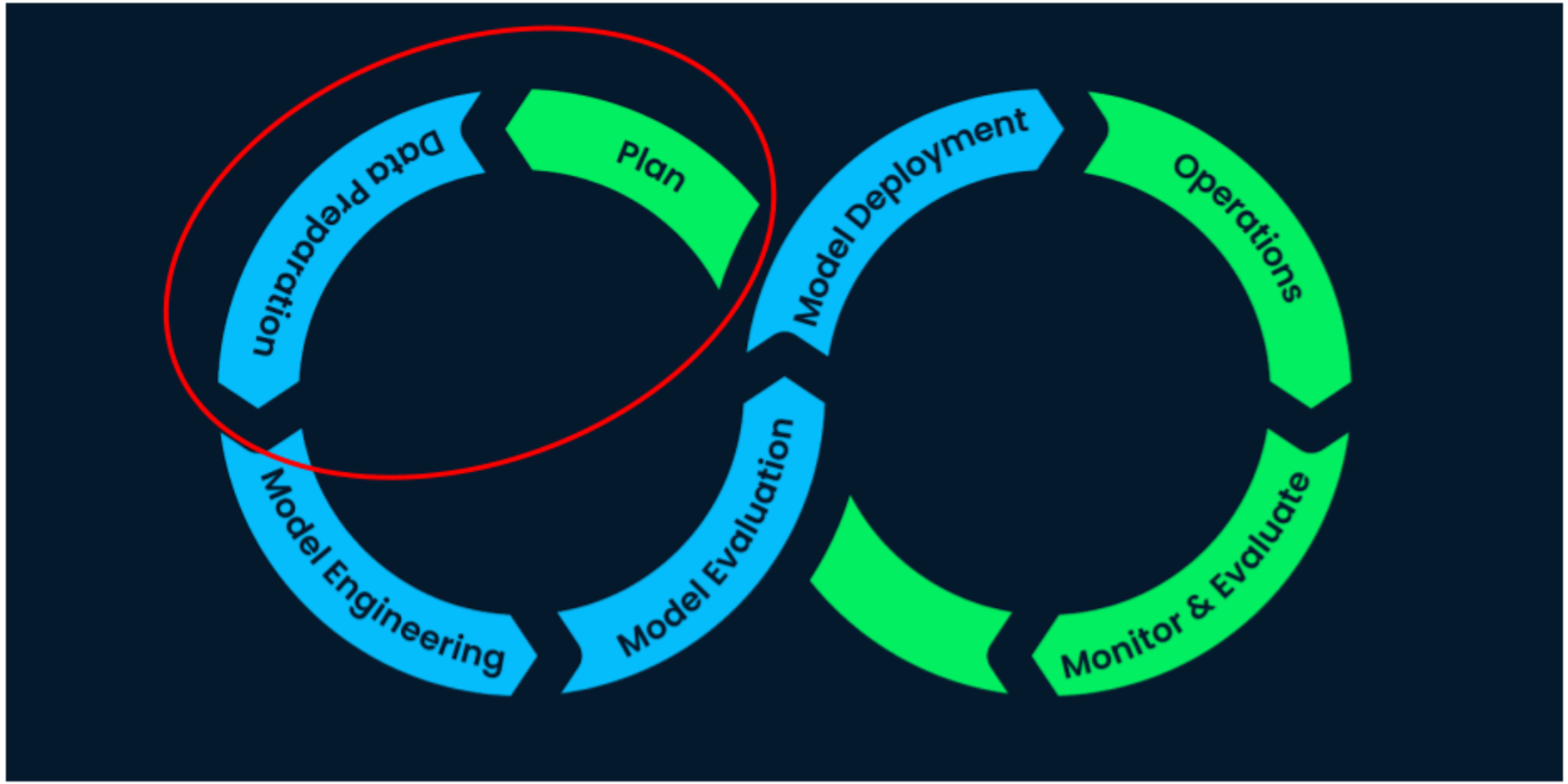
datacamp

Machine Learning Lifecycle



[1] https://www.datacamp.com/blog/machine-learning-lifecycle-explained

Planning and preparation



Planning for machine learning

What do I have?

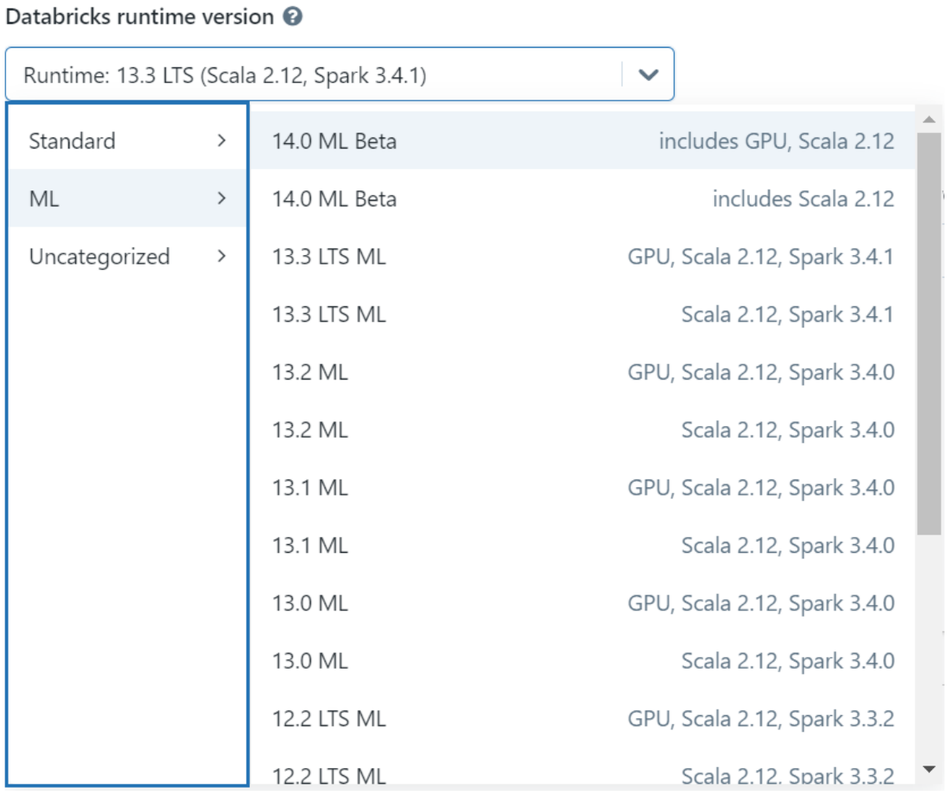
1. Data availability
2. Business requirements
3. Data scientists/data analysts

What do I want?

1. Use cases
2. Legal and security compliance
3. Business outcomes

ML Runtime

- Extension of Databricks compute
- Optimized for machine learning applications
- Contains most common libraries and frameworks
  - `scikit-learn`, `SparkML`, `TensorFlow`
  - `MLFlow`
- Works with cluster library management



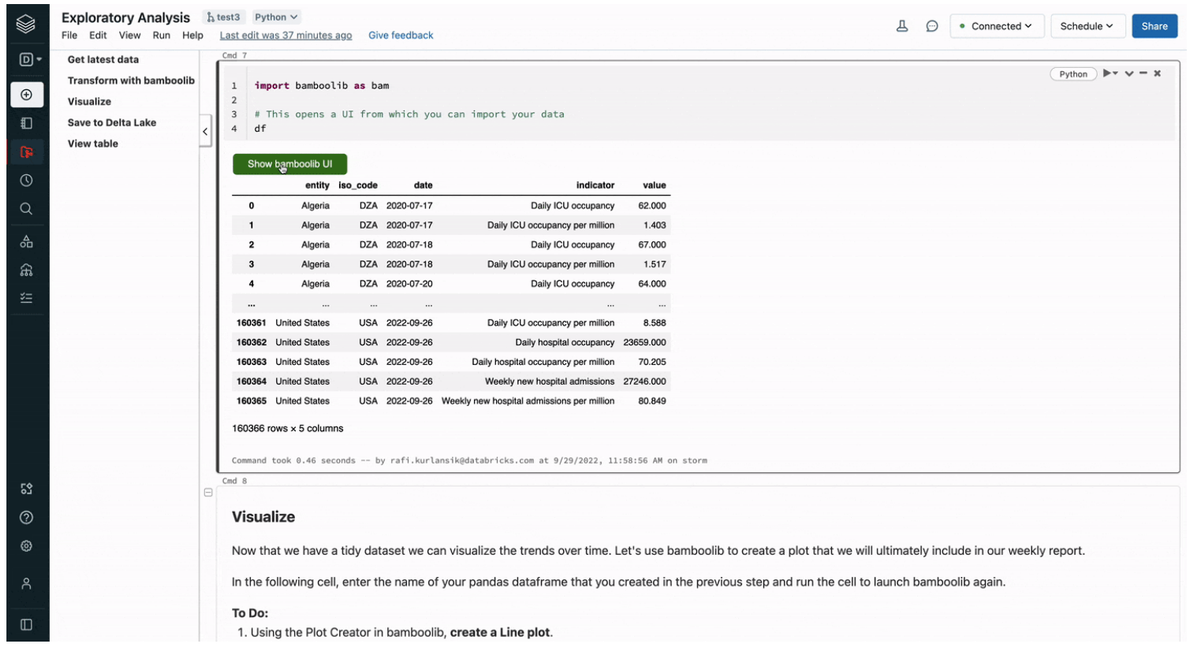
Exploratory Data Analysis

```
import pandas as pd
pd.describe(df)
```

```
# Spark DF
df.summary()
```

```
dbutils.data.summarize()
```

```
import bamboolib as bam
df
```



Feature tables and feature stores

Raw Data

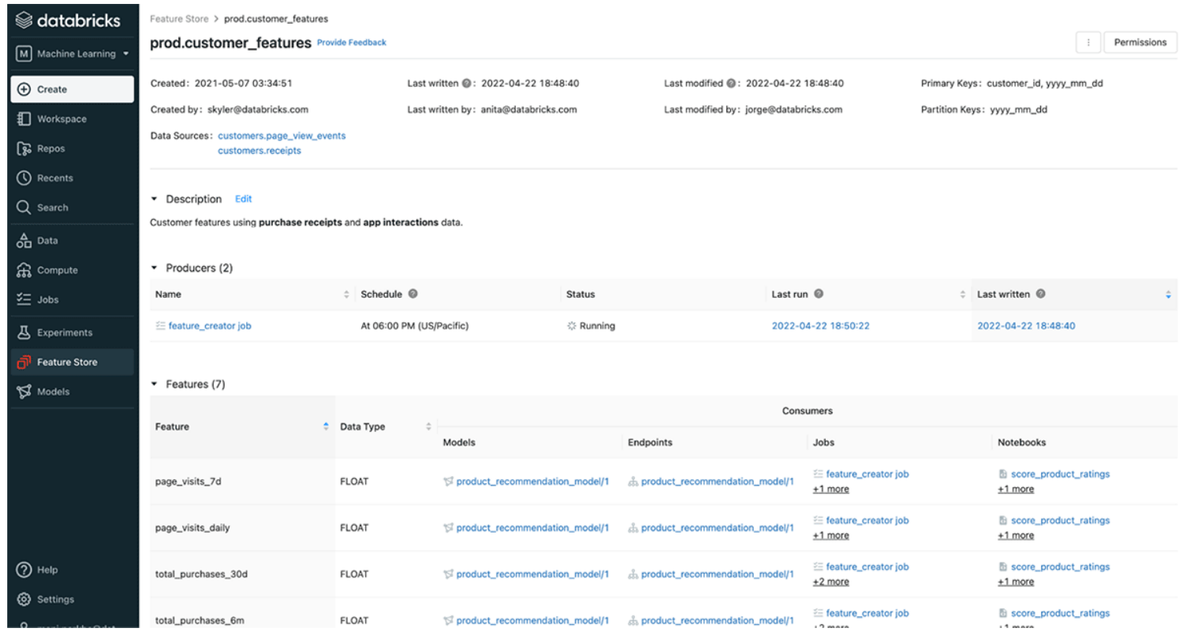
count	category	price	shelf_loc	rating
4	horror	12.50	end	3
6	romance	13.99	top	4.5
12	sci-fi	16.50	bottom	5
31	romance	9.99	bottom	3.5
23	fantasy	24.99	top	4
18	horror	19.99	end	2.5
19	cooking	17.50	end	4.5
7	fantasy	12.99	top	3
37	sci-fi	14.99	bottom	5

Feature table

count	category	price	shelf_loc	rating
4	1	12.50	1	3
6	2	13.99	2	4.5
12	3	16.50	3	5
31	2	9.99	3	3.5
23	4	24.99	2	4
18	1	19.99	1	2.5
19	5	17.50	1	4.5
7	4	12.99	2	3
37	3	14.99	3	5

Databricks Feature Store

- Centralized storage for featurized datasets
- Easily discover and re-use features for machine learning models
- Upstream and downstream lineage




```
from databricks import feature_store


fs = feature_store.FeatureStoreClient()


fs.create_table(
    name=table_name,
    primary_keys=["wine_id"],
    df=features_df,
    schema=features_df.schema,
    description="wine features"
)
```

Let's practice!

DATABRICKS CONCEPTS

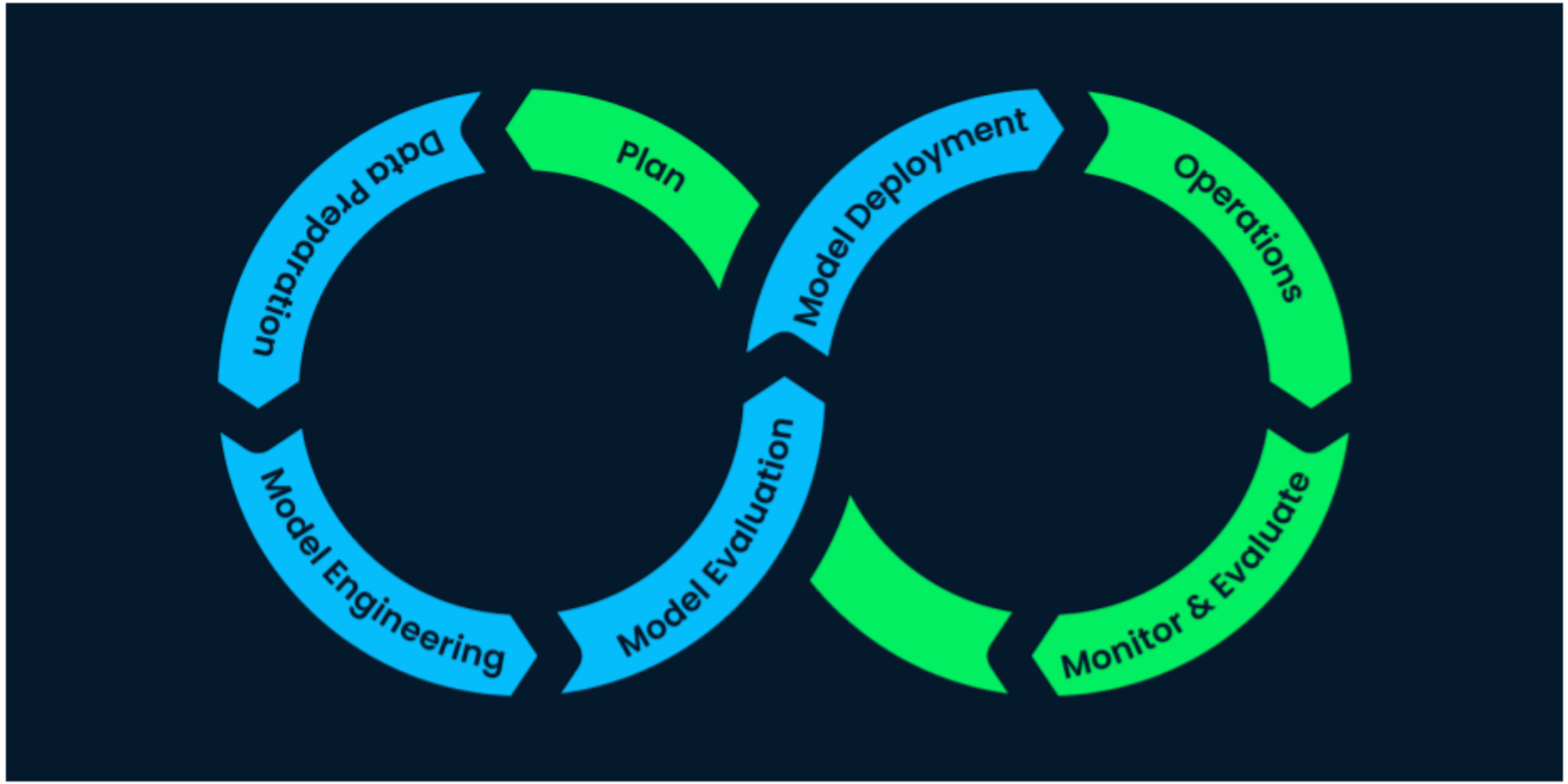
Model training with MLFlow in Databricks

DATABRICKS CONCEPTS

Kevin Barlow
Data Practitioner

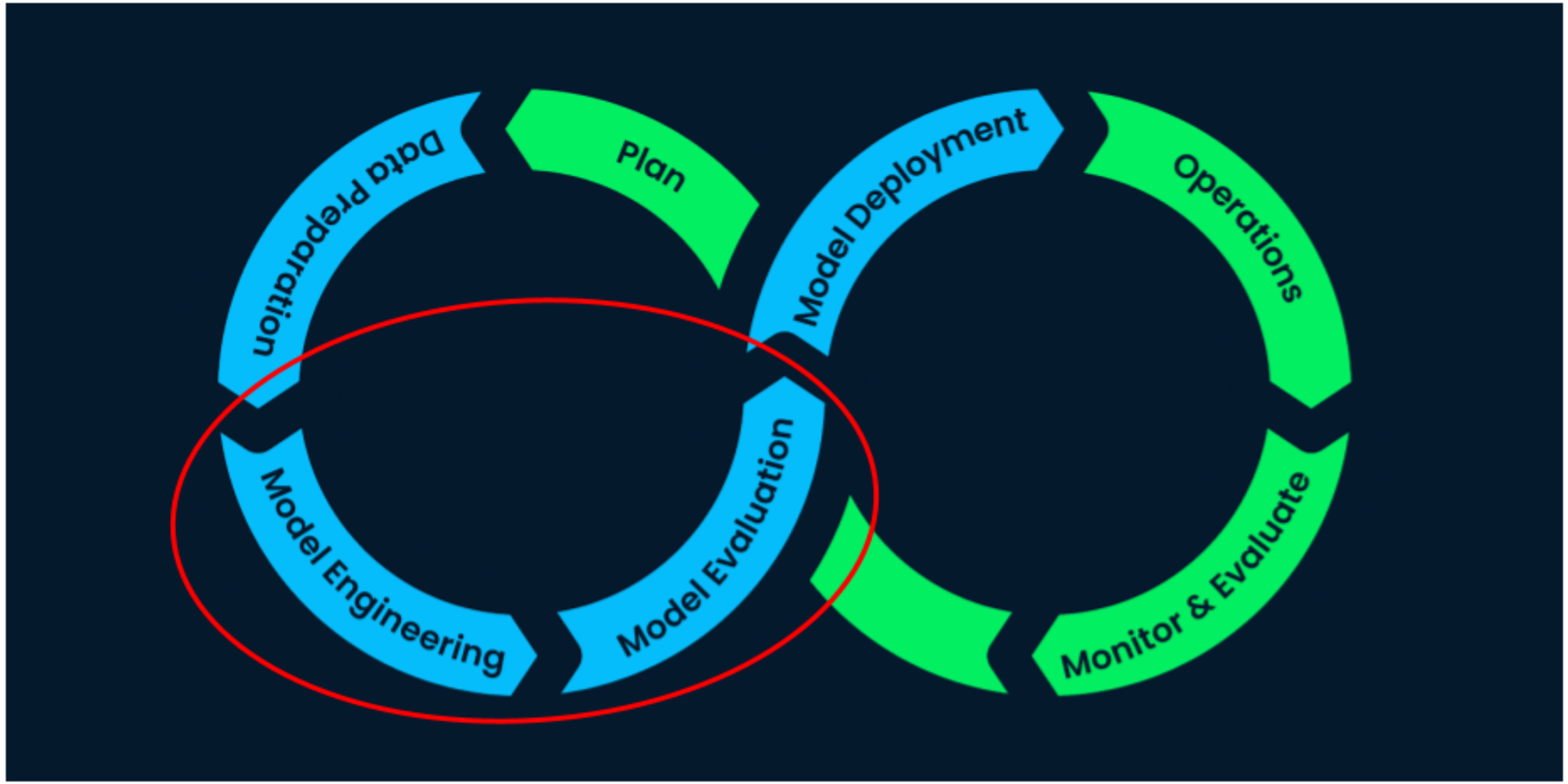
datacamp

Machine Learning Lifecycle



[1] https://www.datacamp.com/blog/machine-learning-lifecycle-explained

Model training and development



Single-node vs. Multi-node

Single-node machine learning

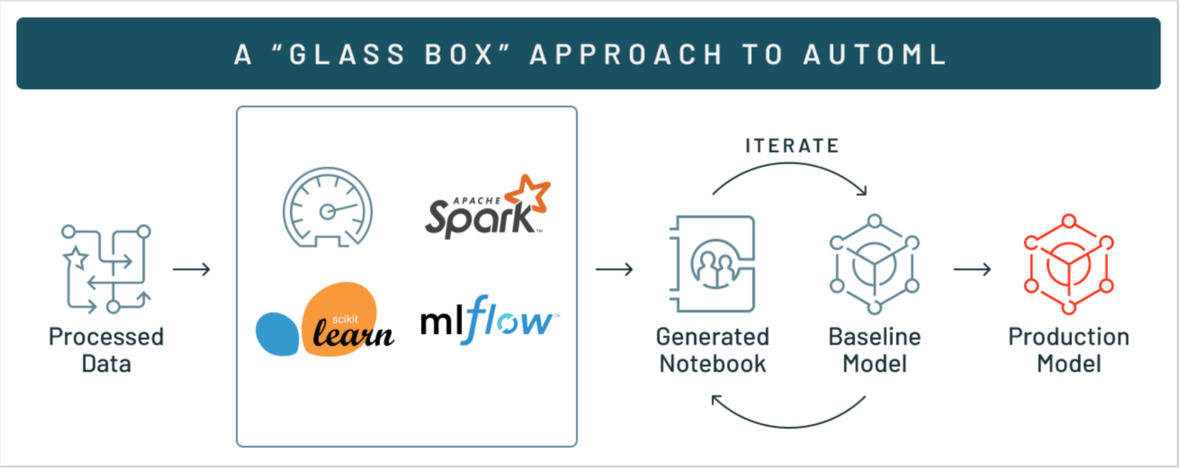
- Great for experimenting and starting
- Easier initial setup
- Hard to implement in production

Multi-node machine learning

- Great for production workloads
- Easier maintenance long-term
- Highly scalable

AutoML

- "Glass box" approach to automated machine learning
- Leverages open-source libraries
- Creates models based on data and targeted prediction
- Provides notebook with generated code for further



[1] https://www.databricks.com/product/automl

MLFlow

- Open-source framework
- End-to-end machine learning lifecycle management
- Track, evaluate, manage, and deploy
- Pre-installed on ML Runtime!


```
import mlflow

with mlflow.start_run() as run:
  # machine learning training

mlflow.autolog()

mlflow.log_metric('accuracy', acc)

mlflow.lot_param('k', kNum)
```

MLFlow Experiments

- Collect information across multiple runs in a single location
- Sort and compare model runs
- Find and promote the best model

Let's practice!

DATABRICKS CONCEPTS

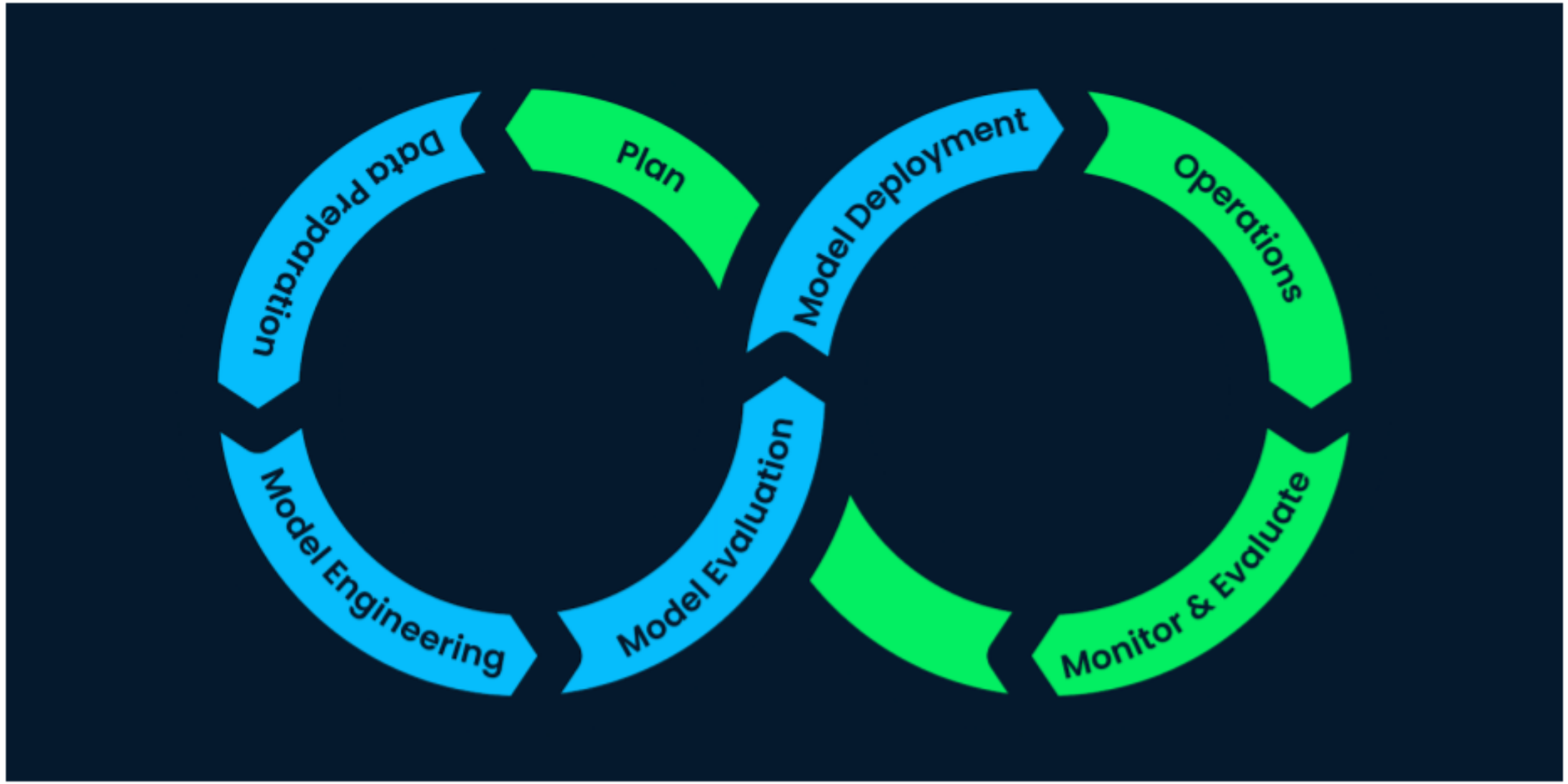
Deploying a model in Databricks

DATABRICKS CONCEPTS

Kevin Barlow
Data Practitioner

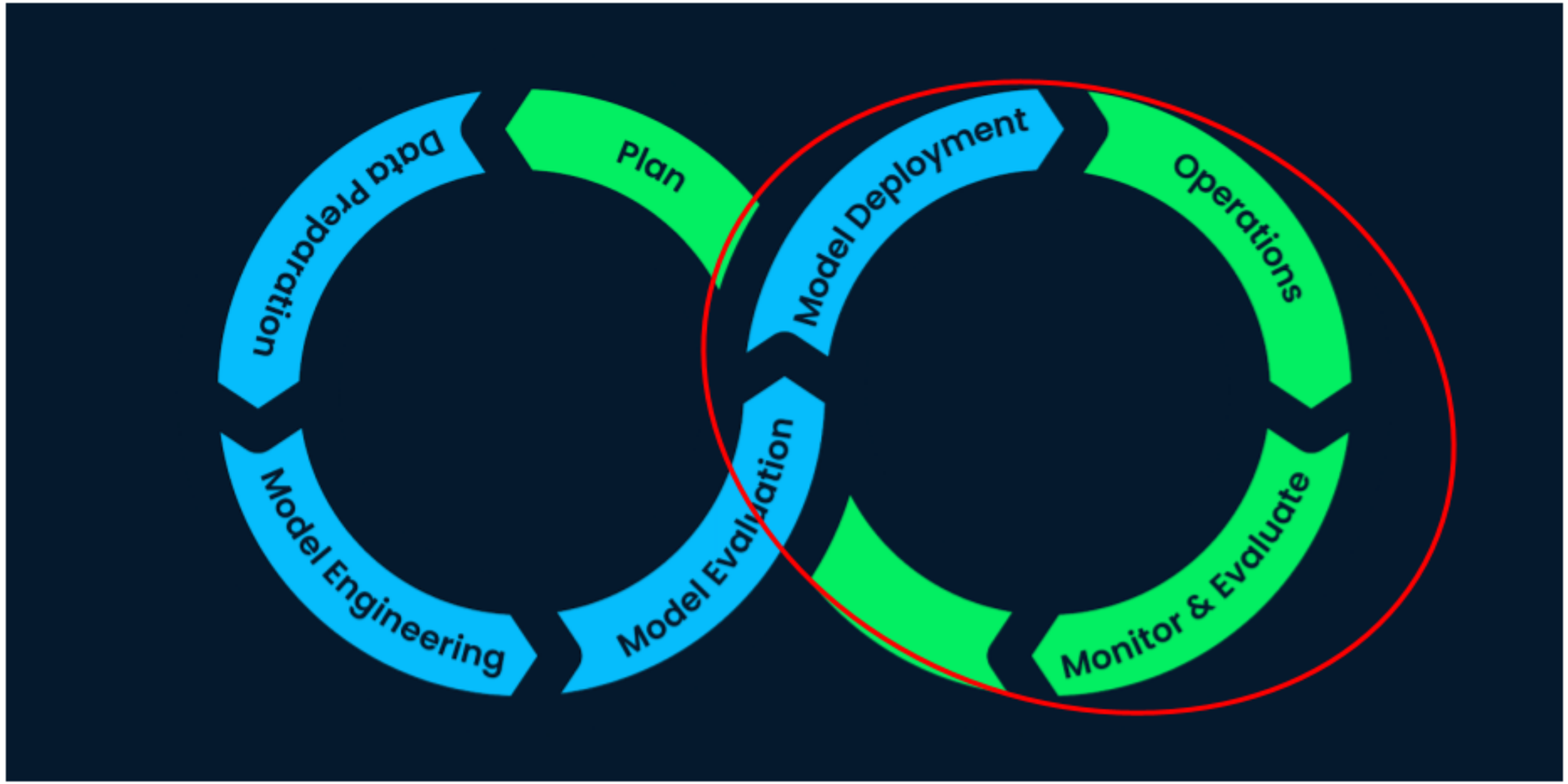
datacamp

Machine Learning Lifecycle



[1] https://www.datacamp.com/blog/machine-learning-lifecycle-explained

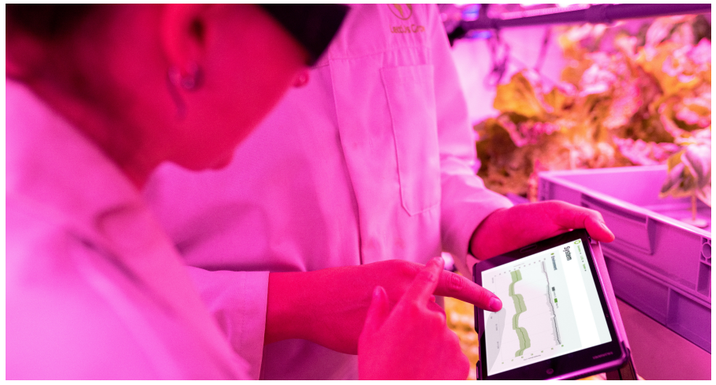
Model Deployment and Operations



Concerns with deploying models

Availability

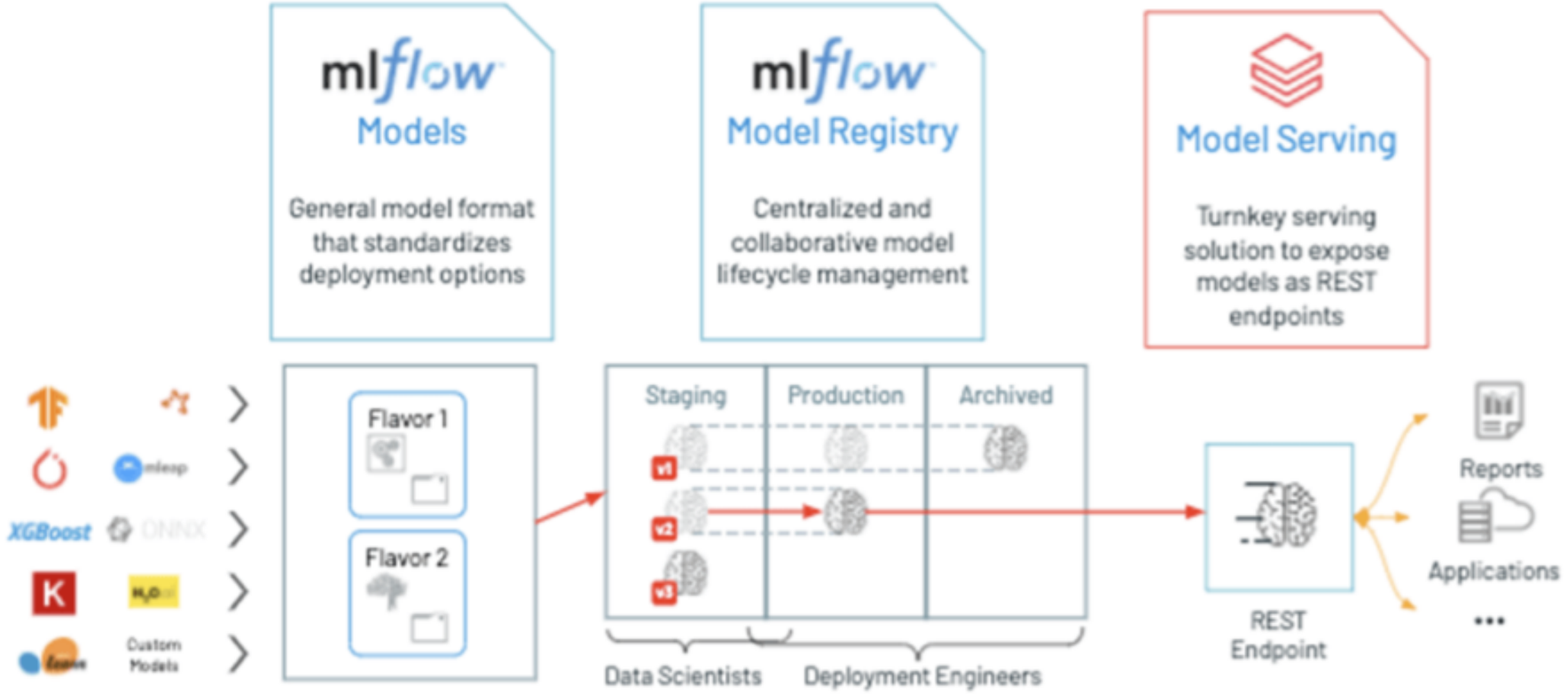
- How will my end users or application use the model?
- Where do I need to put my model to access it?
- Will the model be easy to understand or use?



Evaluation

- Are my users *actually* using my model?
- Is my model still performing well?
- Do I need to retrain my model?
- Do I need a new model that is better?

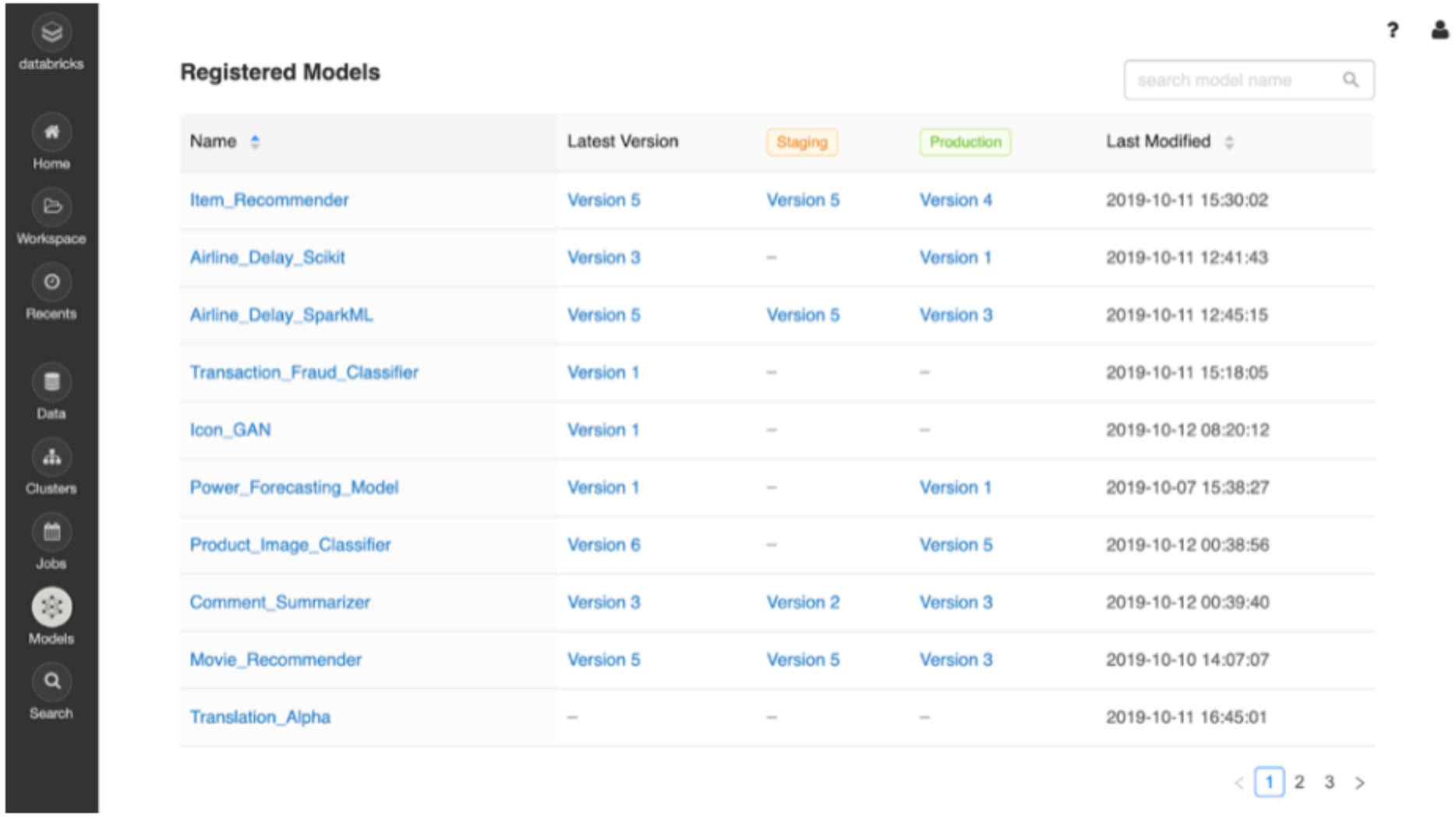
Model Deployment Process



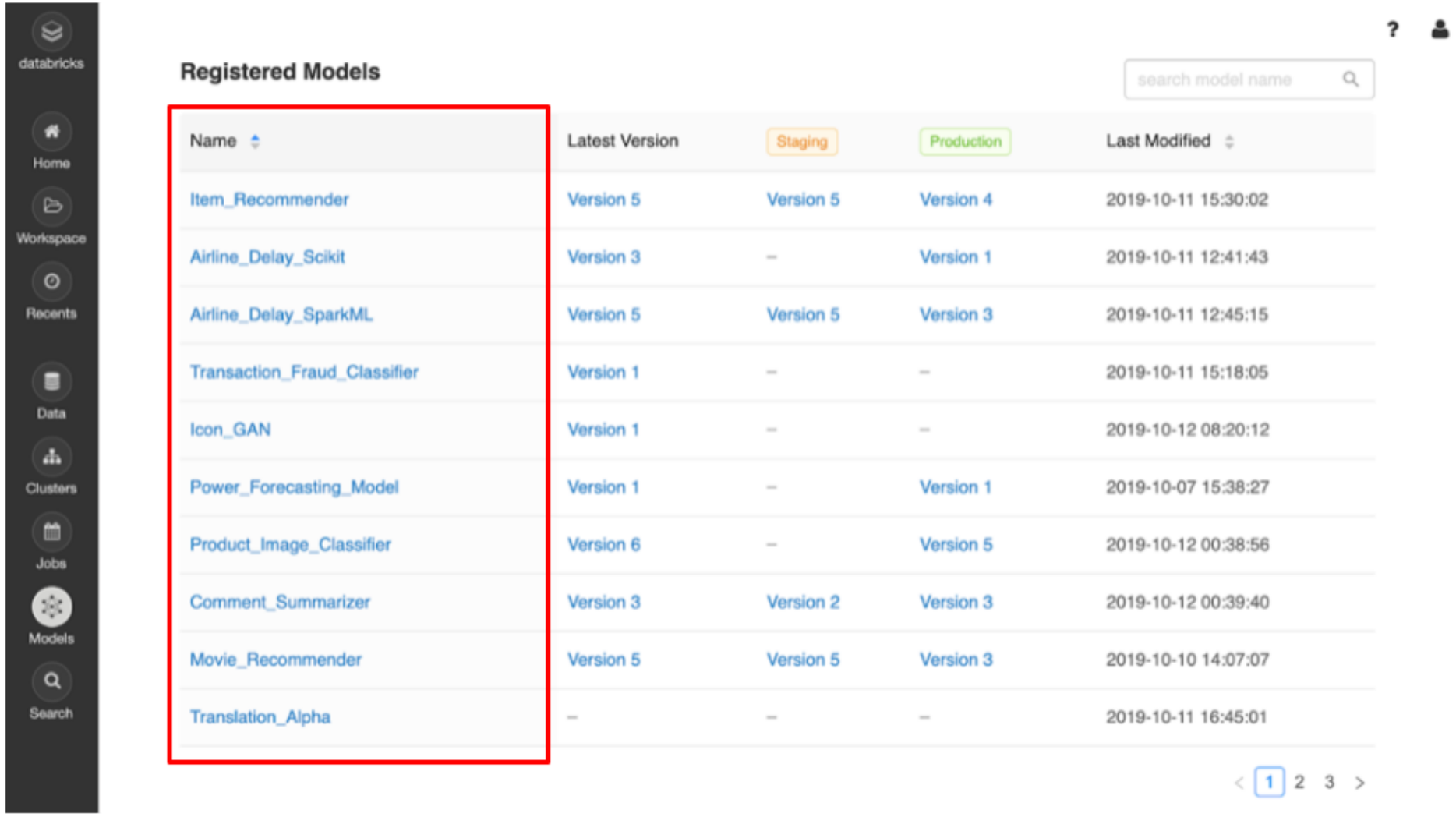
Model Flavors

- MLFlow Models can store a model from any machine learning framework
- Models are stored alongside different configurations and artifacts
- Models can be "translated" into another kind of model based on needs. For example:
  - scikit-learn
  - pyfunc
  - spark
  - tensorflow

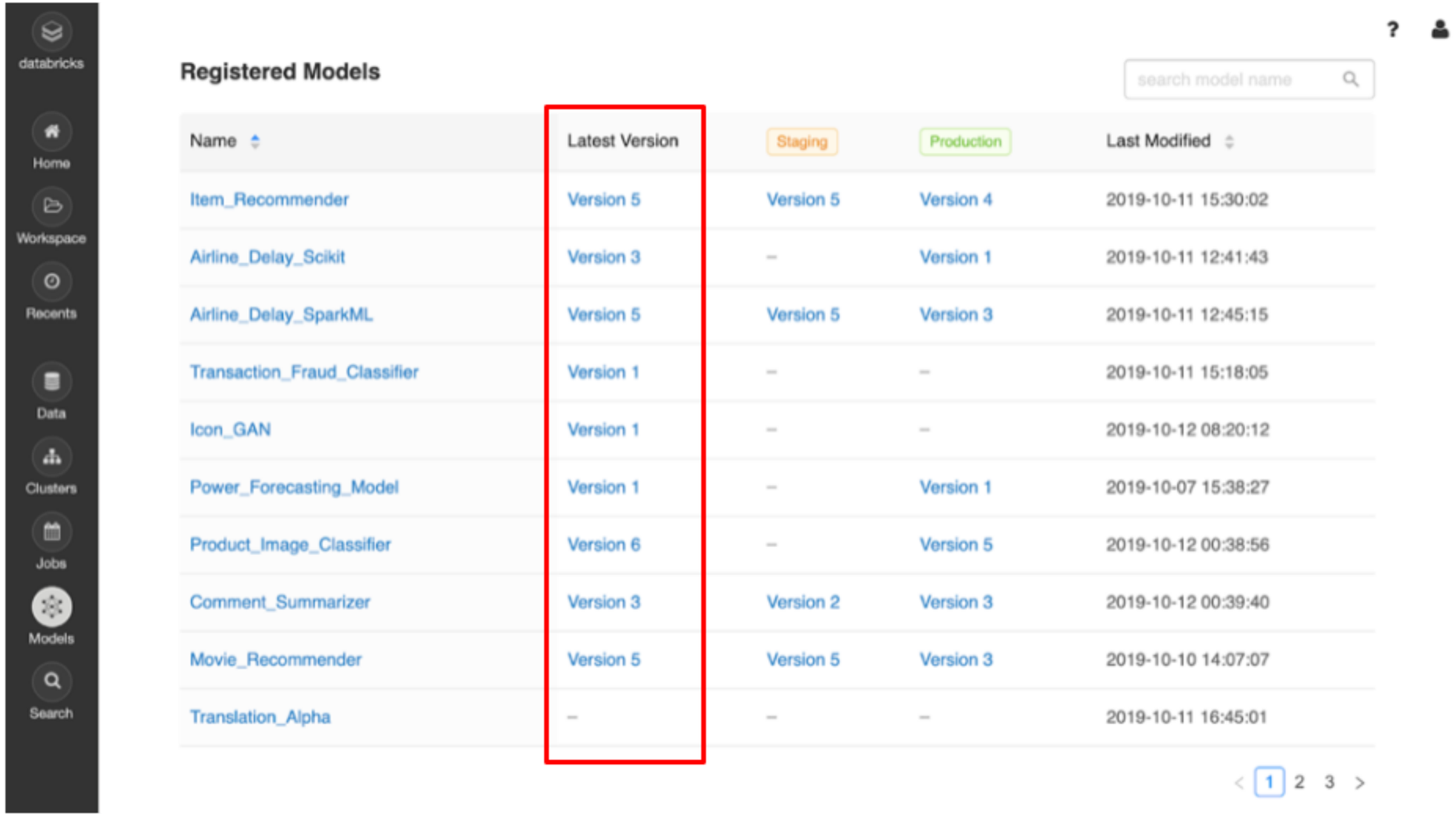
Model Registry



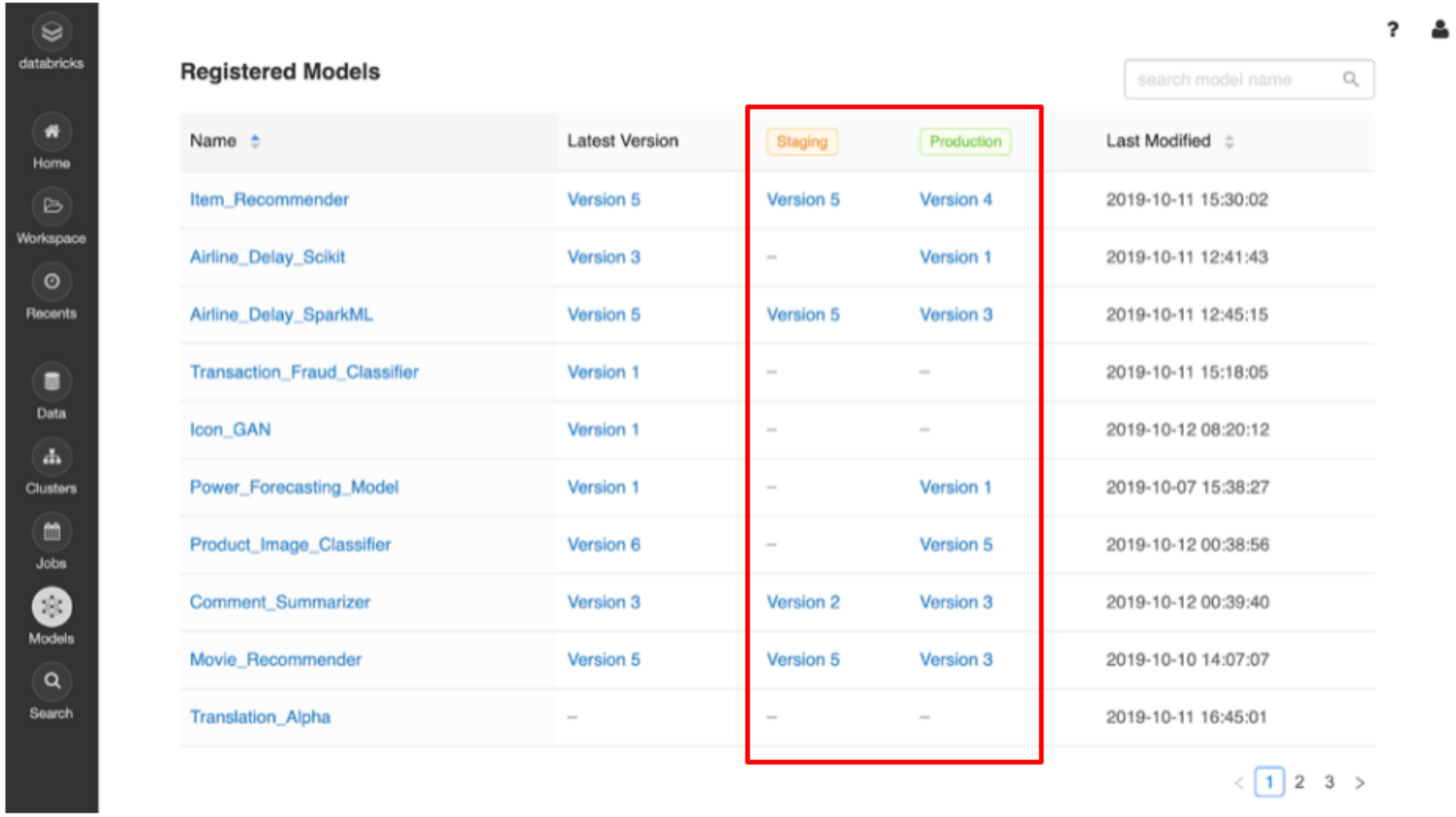
Model Registry



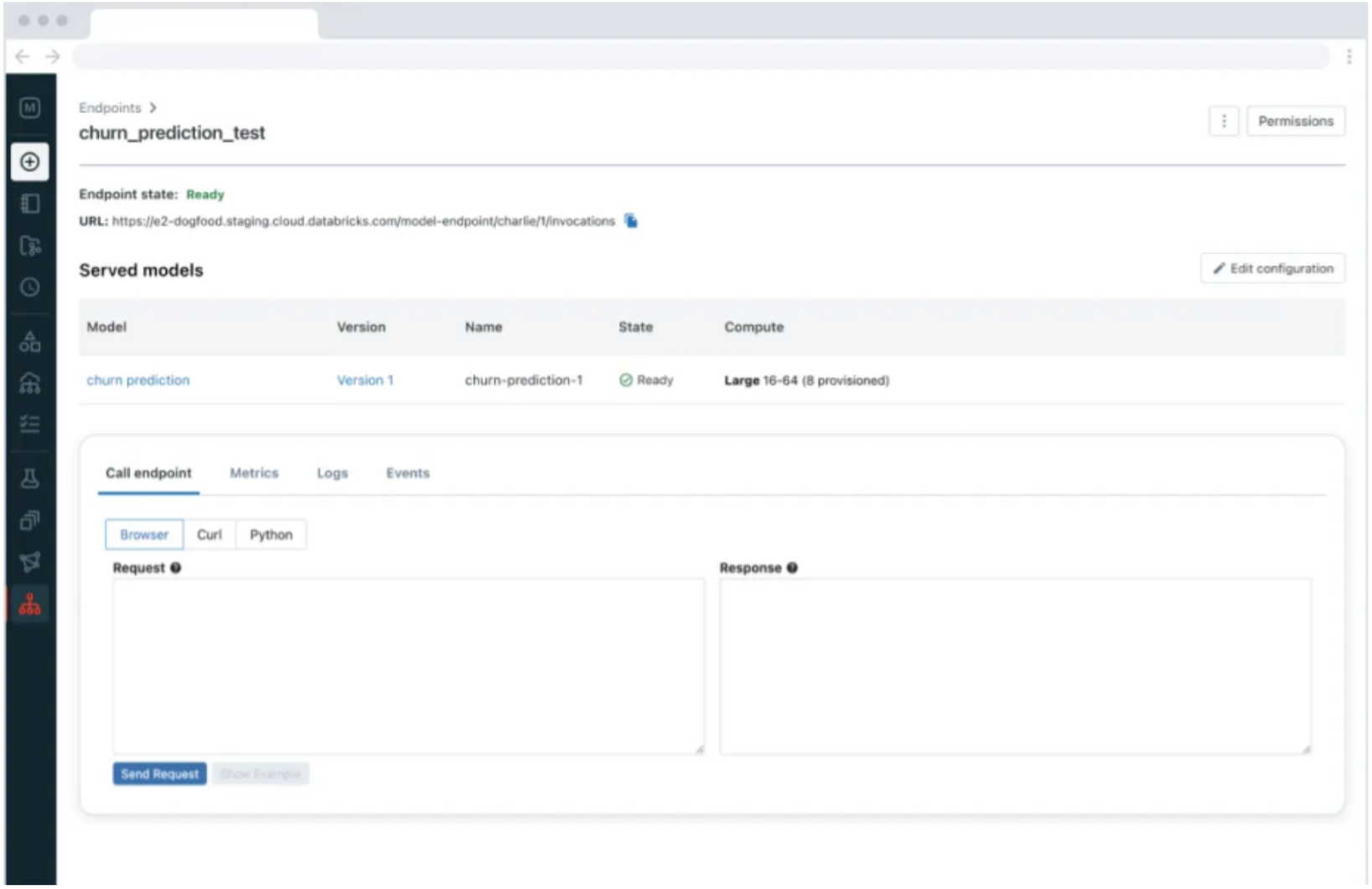
Model Registry



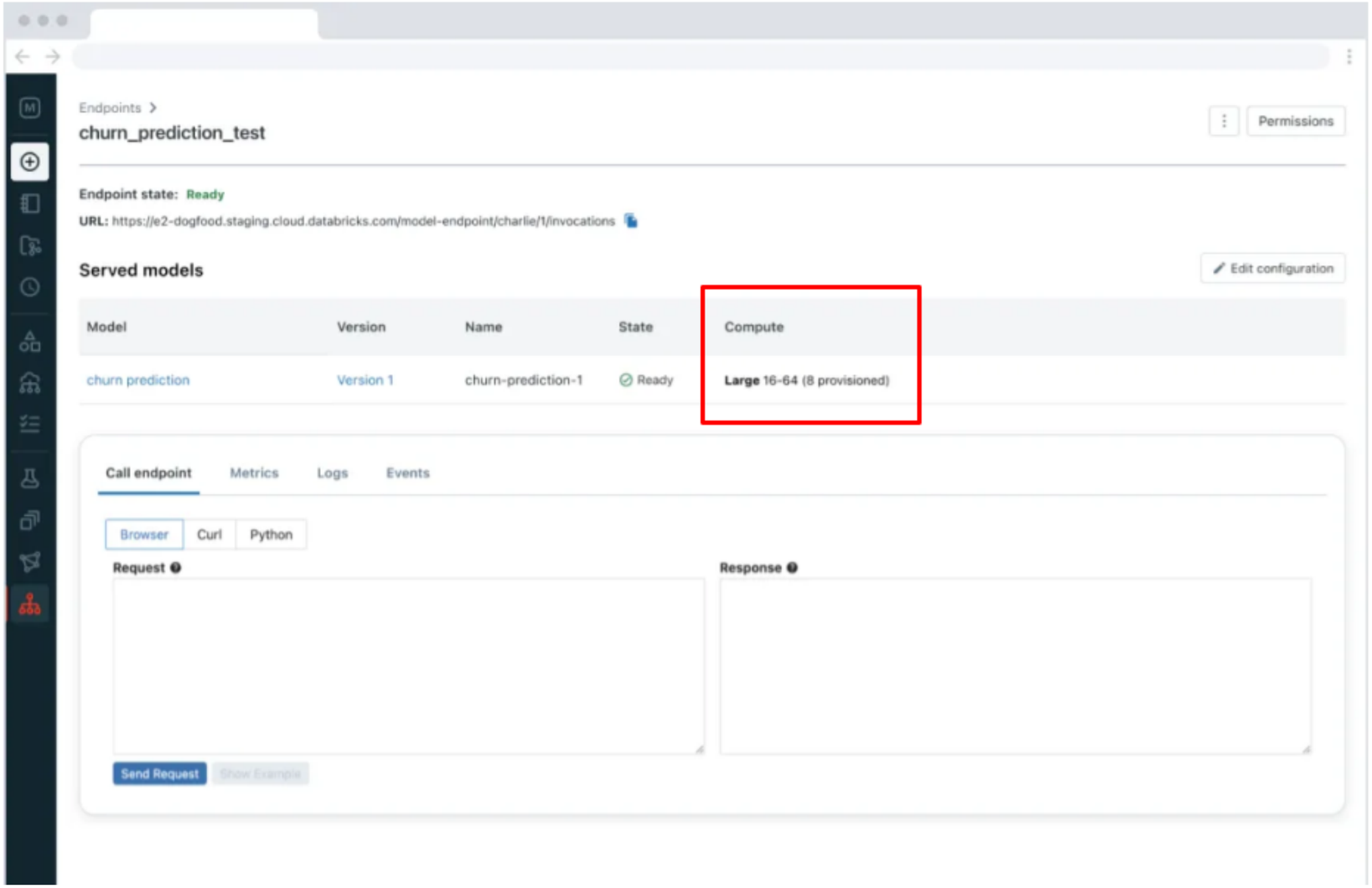
Model Registry



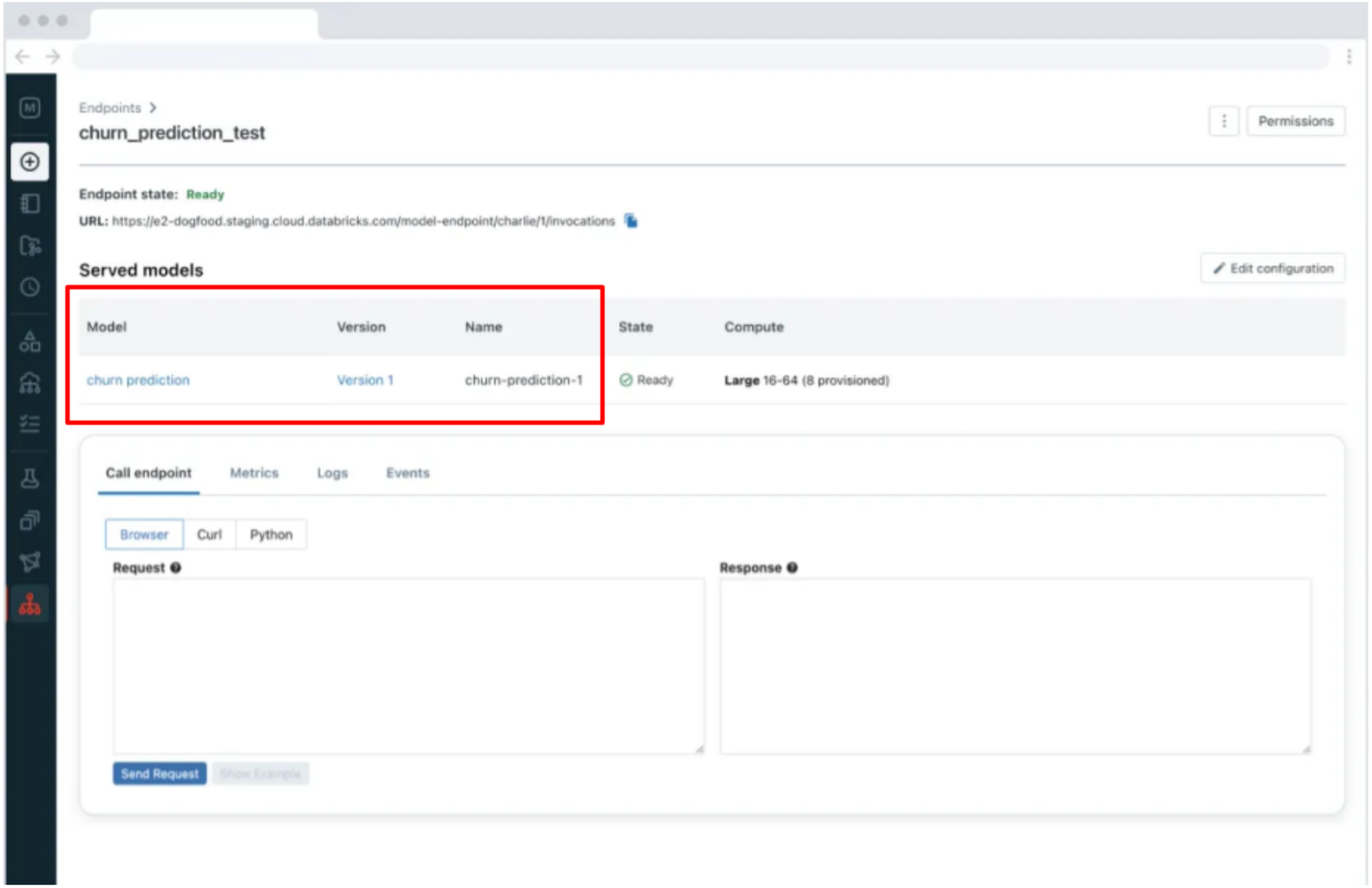
Model Serving



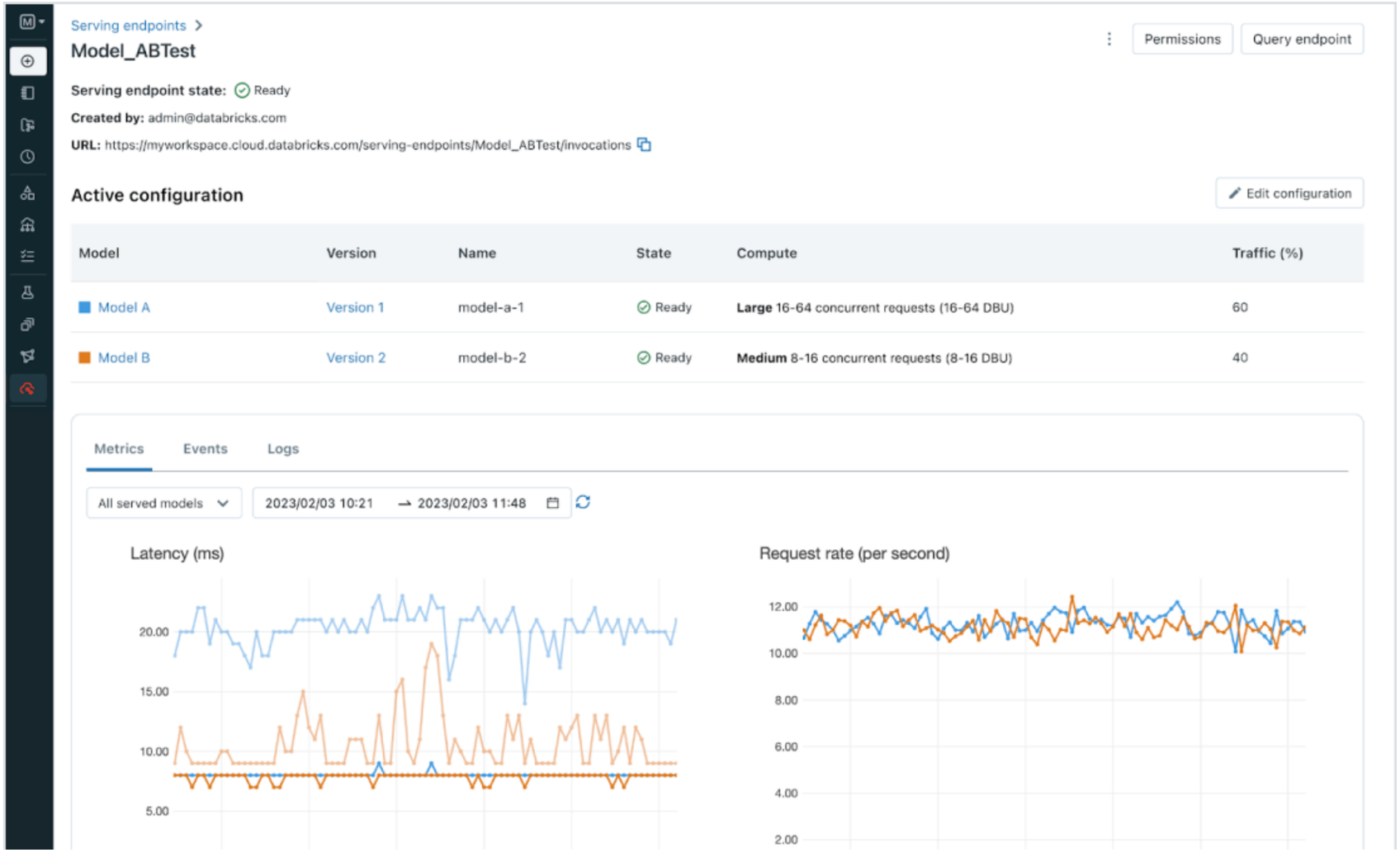
Model Serving



Model Serving



Model Serving



[1] https://www.databricks.com/product/model-serving

Let's practice!

DATABRICKS CONCEPTS

Example end-to-end machine learning pipeline

DATABRICKS CONCEPTS

Kevin Barlow
Data Practitioner

datacamp

Let's practice!

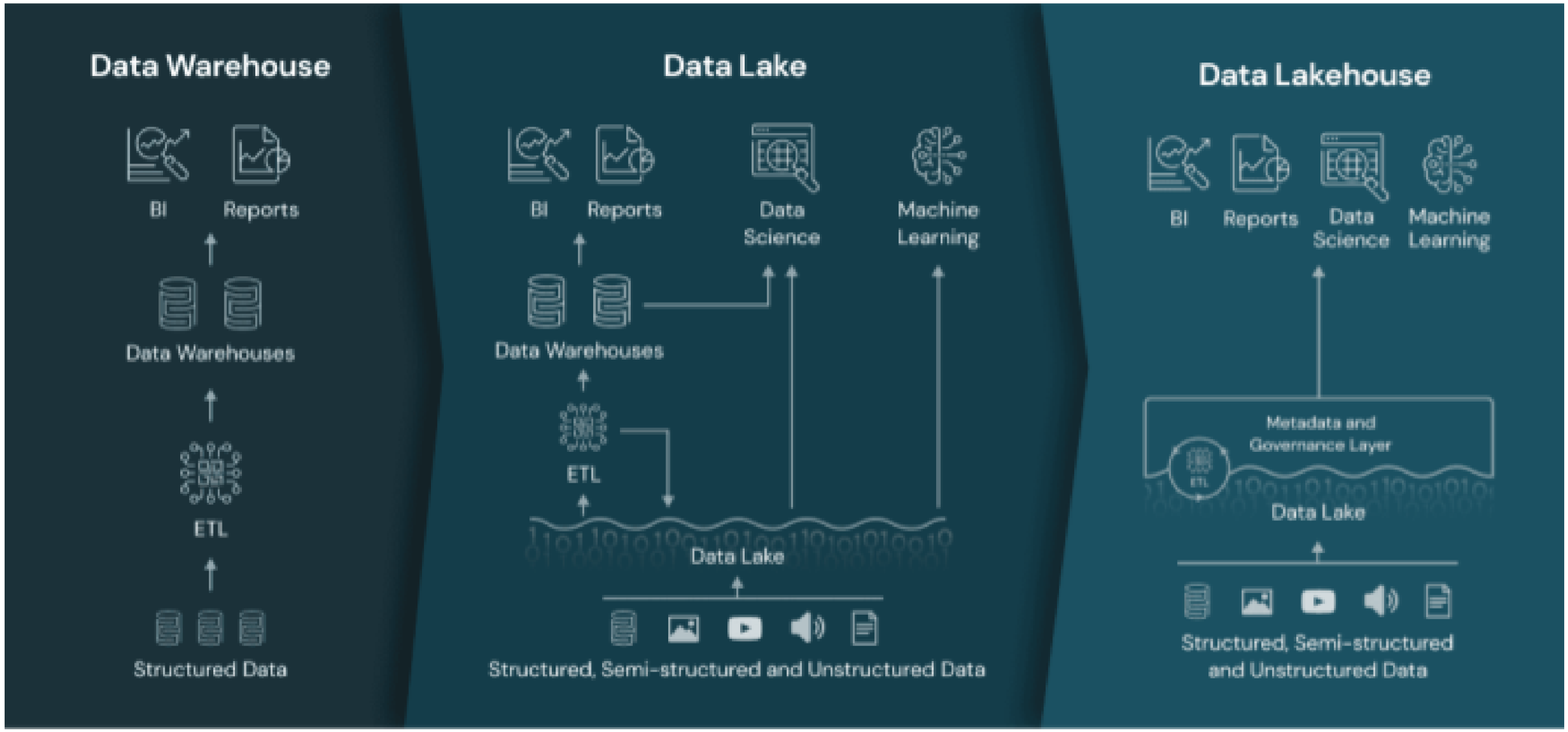
DATABRICKS CONCEPTS

Wrap Up

DATABRICKS CONCEPTS

Kevin Barlow
Data Practitioner

Why the Lakehouse?



[1] https://www.databricks.com/blog/2020/01/30/what-is-a-data-lakehouse.html

Databricks for data engineering

Apache Spark

Delta

Delta Live Tables

Auto Loader

Structured Streaming

Workflows

Databricks for data warehousing

SparkSQL

ANSI SQL

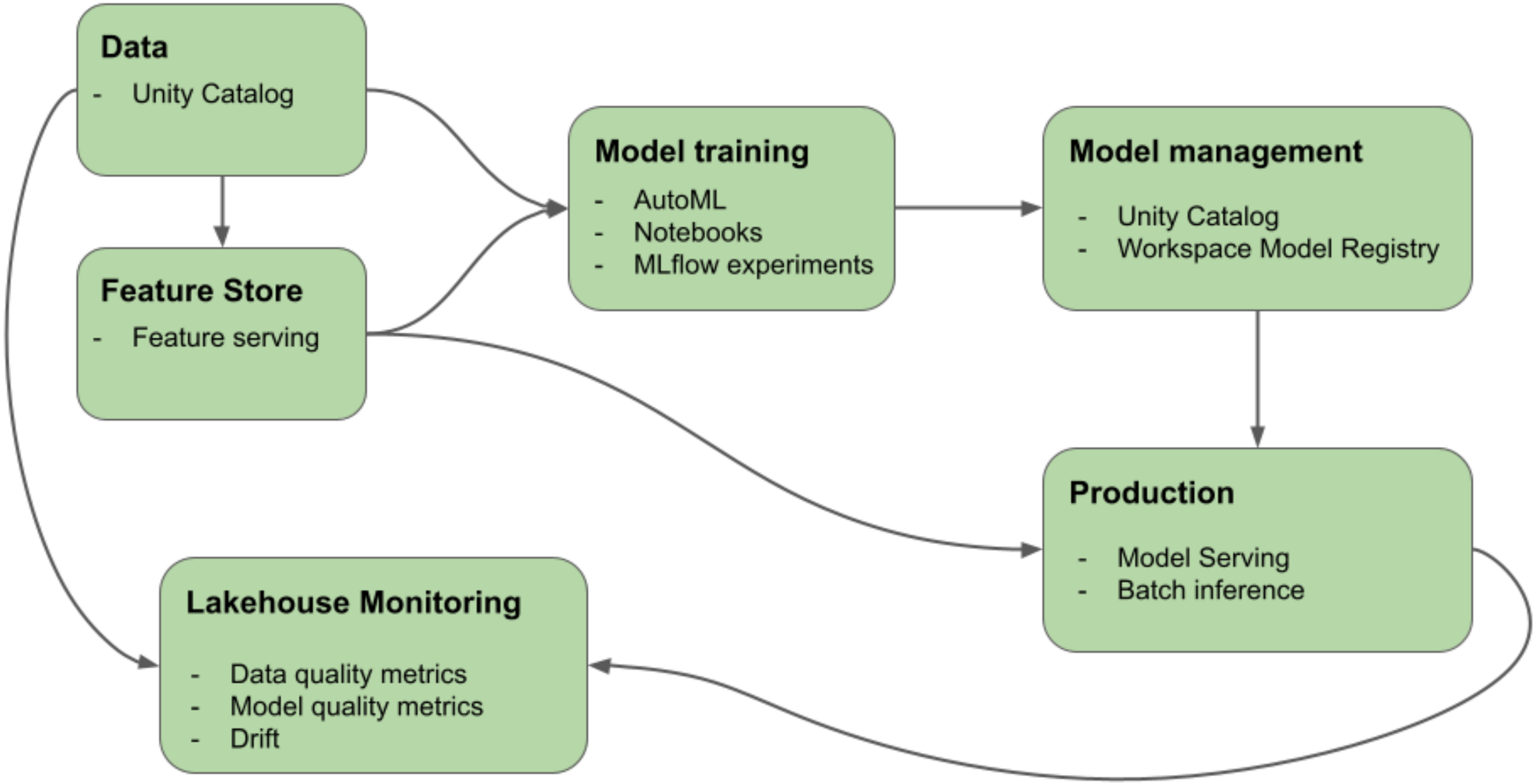
SQL Warehouses

Queries

Visualizations

Dashboards

Databricks for machine learning



Congratulations!

DATABRICKS CONCEPTS