Building Deep Learning Models on Databricks

Introducing MLflow on Databricks

Janani Ravi
Co-founder, Loonycorn

www.loonycorn.com

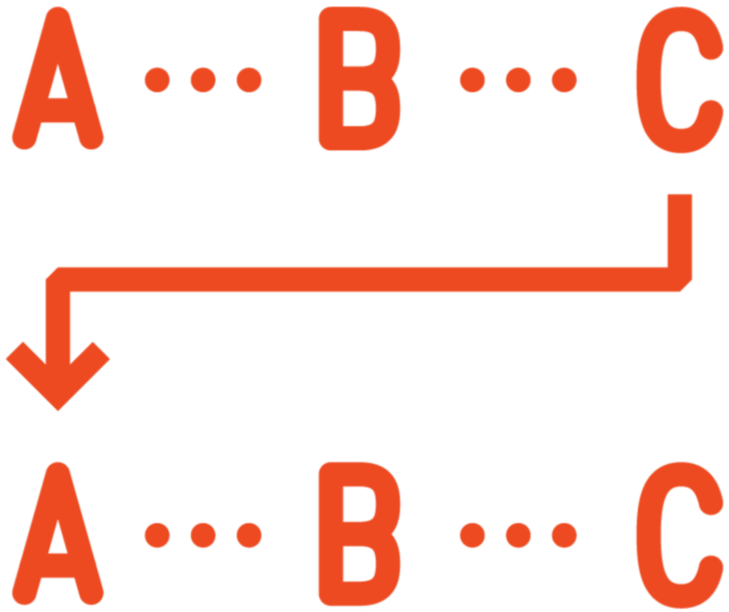
Overview

The Databricks ML runtime

MLflow to manage the ML workflow

Prerequisites and Course Outline

Prerequisites



Comfortable programming in Python

Comfortable with deep learning models using TensorFlow/PyTorch

Familiar with the Databricks platform

Prerequisite Courses

Getting Started with TensorFlow 2.0

Building Your First PyTorch Solution

Prerequisite Courses

Getting Started with the Databricks Lakehouse Platform

Getting Started with Apache Spark on Databricks

Course Outline

Introducing MLflow on Databricks

Implementing Deep Learning Models Using TensorFlow and Keras

Implementing Deep Learning Models Using PyTorch

Hyperparameter Tuning Using Hyperopt

The Databricks Machine Learning Runtime

Databricks

An enterprise software company founded by the creators of Apache Spark. The company has also created Delta Lake, MLflow, and Koalas, – all open source projects that span data engineering, data science, and machine learning.

https://en.wikipedia.org/wiki/Databricks

Databricks

A cloud-native platform for big data processing, machine learning, and analytics built using the Data Lakehouse architecture.

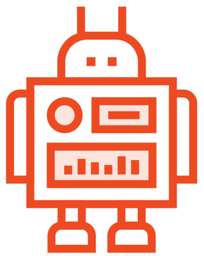
Databricks Data Analytics Platform

Databricks SQL

Platform for analysts to run SQL queries on data, create visualizations, share dashboards

Databricks Data Science and Engineering

Interactive workspace for collaboration between data engineers, data science, and ML engineers to generate insights using Spark.



Databricks Machine Learning

Integrated end-to-end machine learning environment with managed services for the ML workflow

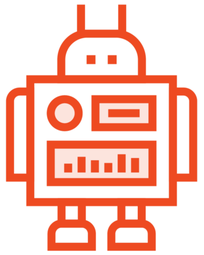
Databricks Data Analytics Platform

Databricks SQL

Platform for analysts to run SQL queries on data, create visualizations, share dashboards

Databricks Data Science and Engineering

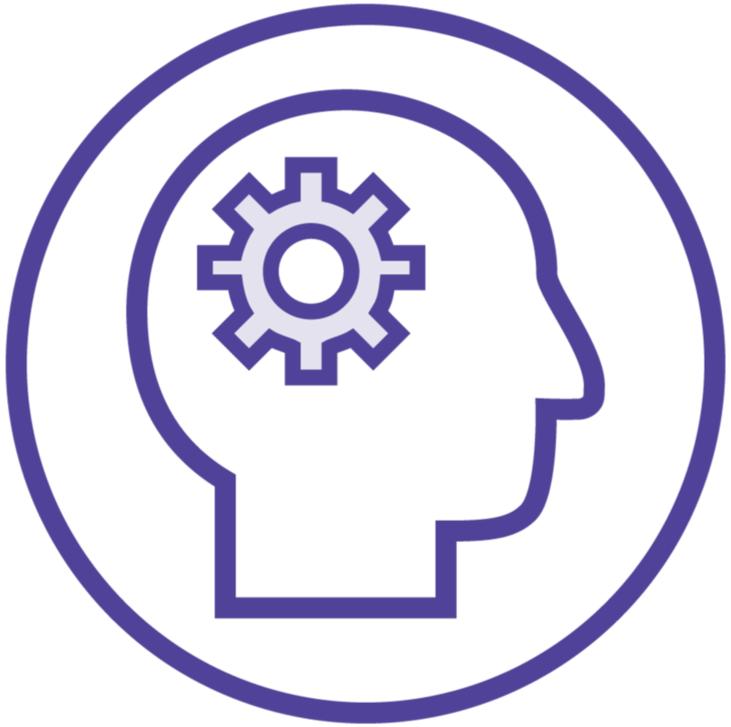
Interactive workspace for collaboration between data engineers, data science, and ML engineers to generate insights using Spark.



Databricks Machine Learning

Integrated end-to-end machine learning environment with managed services for the ML workflow

Databricks ML Runtime

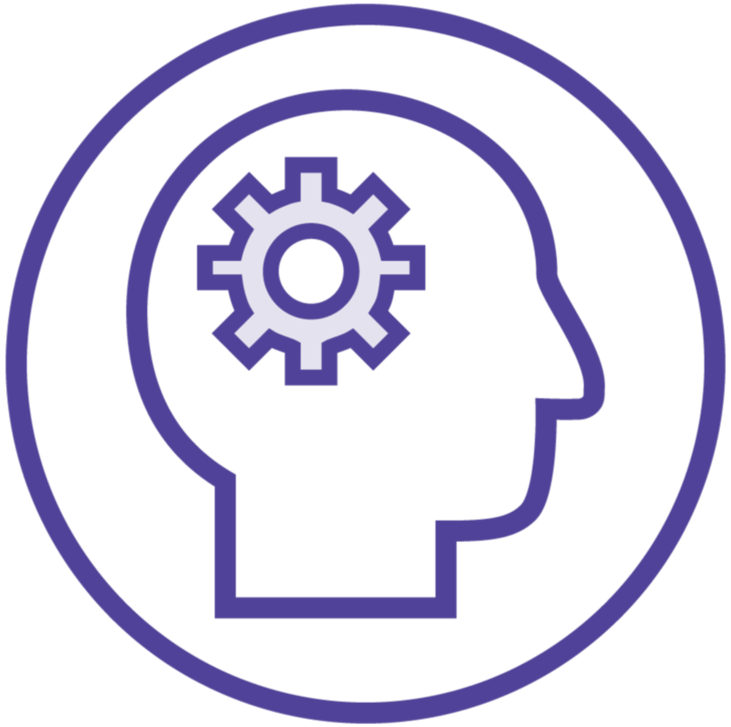


Automates the creation of a cluster optimized for machine learning

Includes popular ML libraries:

- scikit-learn
- XGBoost
- Spark ML
- TensorFlow
- PyTorch

Databricks ML Runtime



Includes support for distributed training libraries such as Horovod

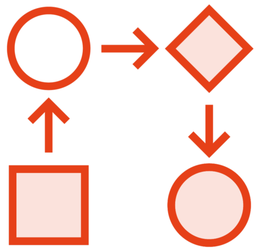
Includes tools to automate the model development process

Performs hyperparameter tuning to find the best model

Automate Machine Learning

AutoML

Automatically creates, tunes, and evaluates models.



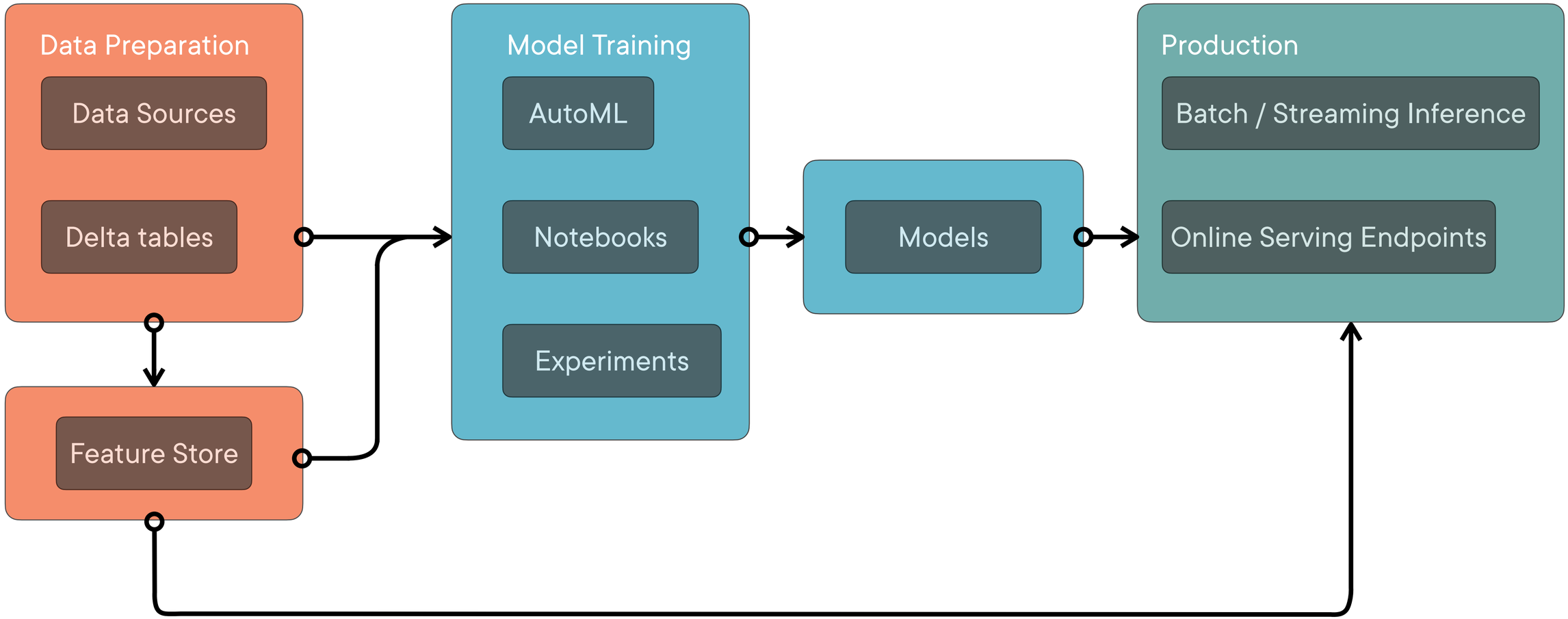
Managed ML Flow

Manages the end-to-end model lifecycle, including tracking experiment runs, deploying, registering, and sharing models

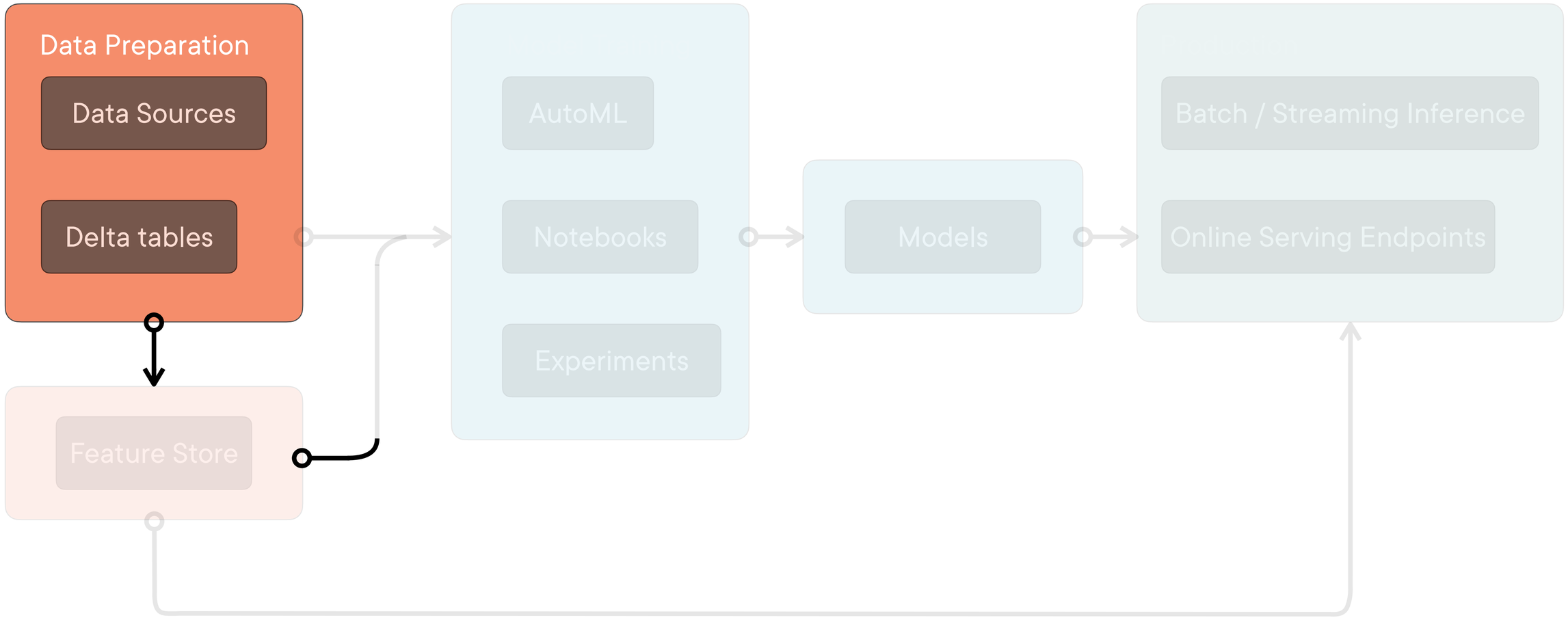
Hyperopt

Uses the SparkTrials class to simplify hyperparameter tuning by automating and distributing model tuning runs

End-to-end Machine Learning Environment

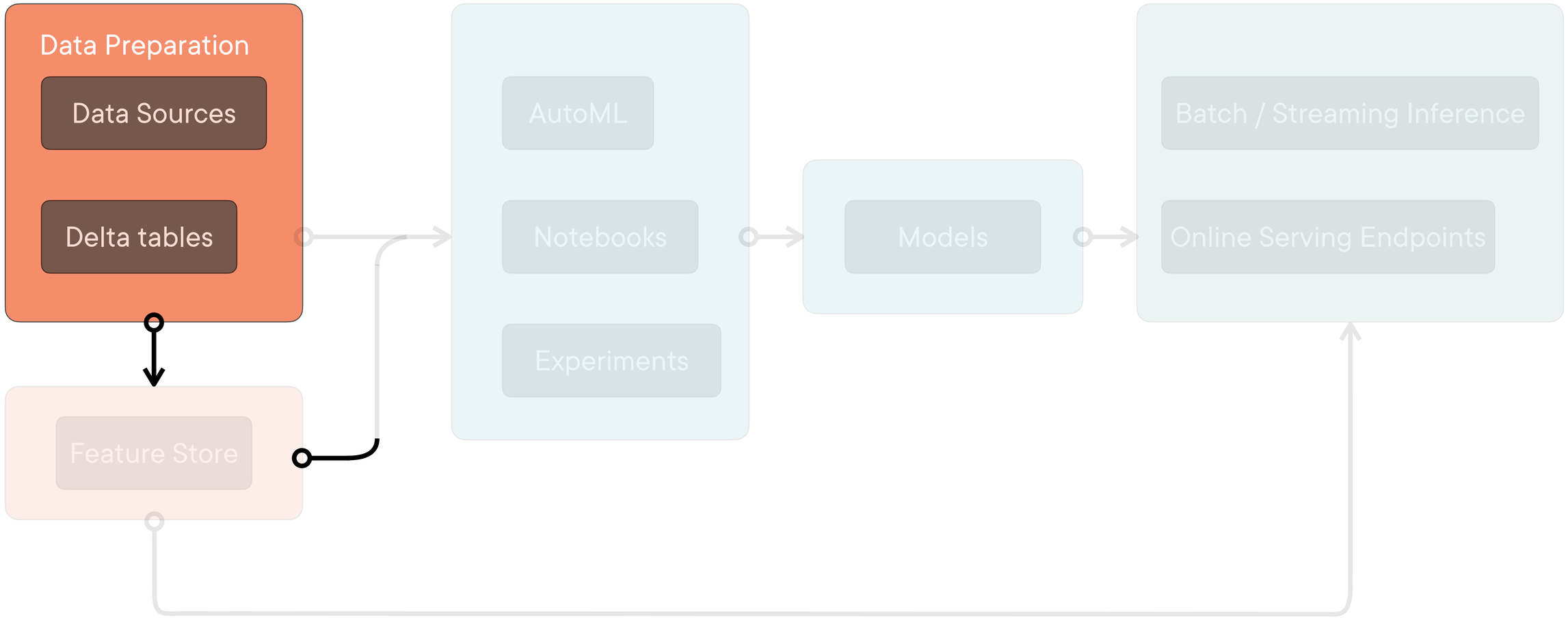


Data Preparation



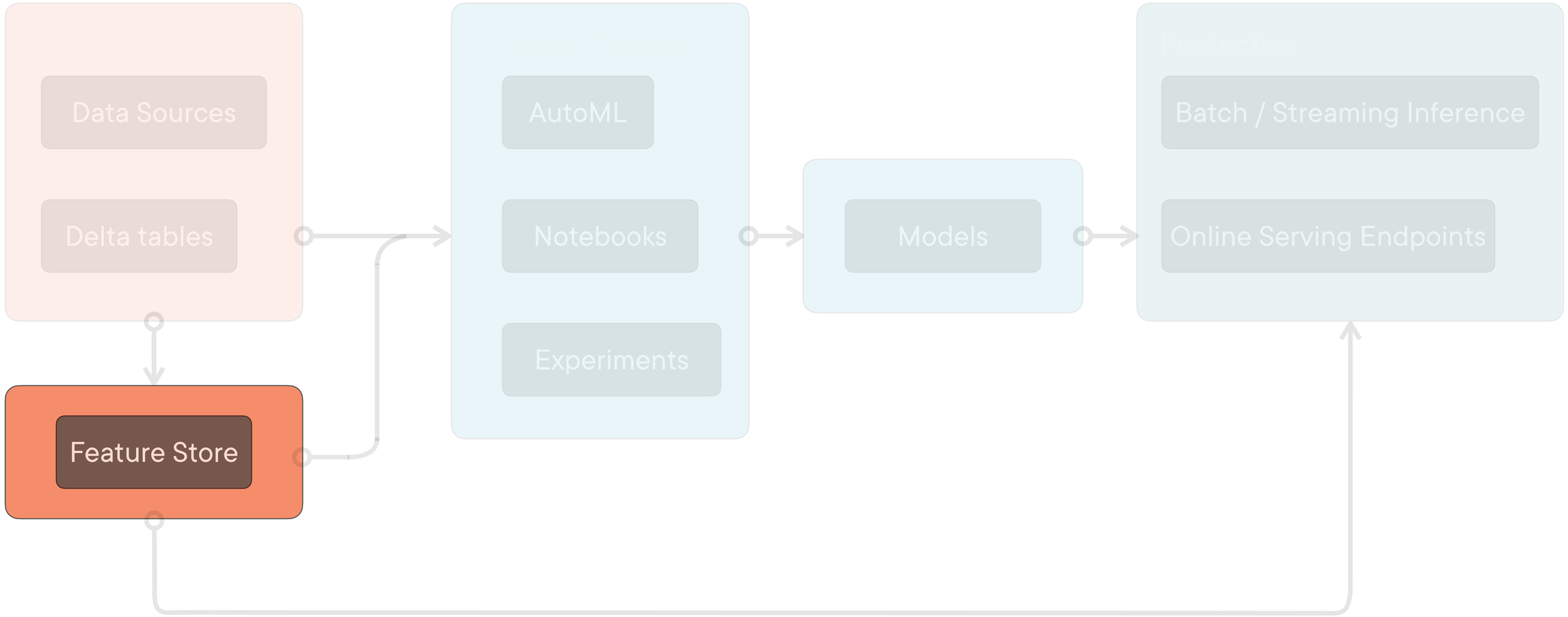
Use Spark or native programming language libraries to connect to data sources

Data Preparation



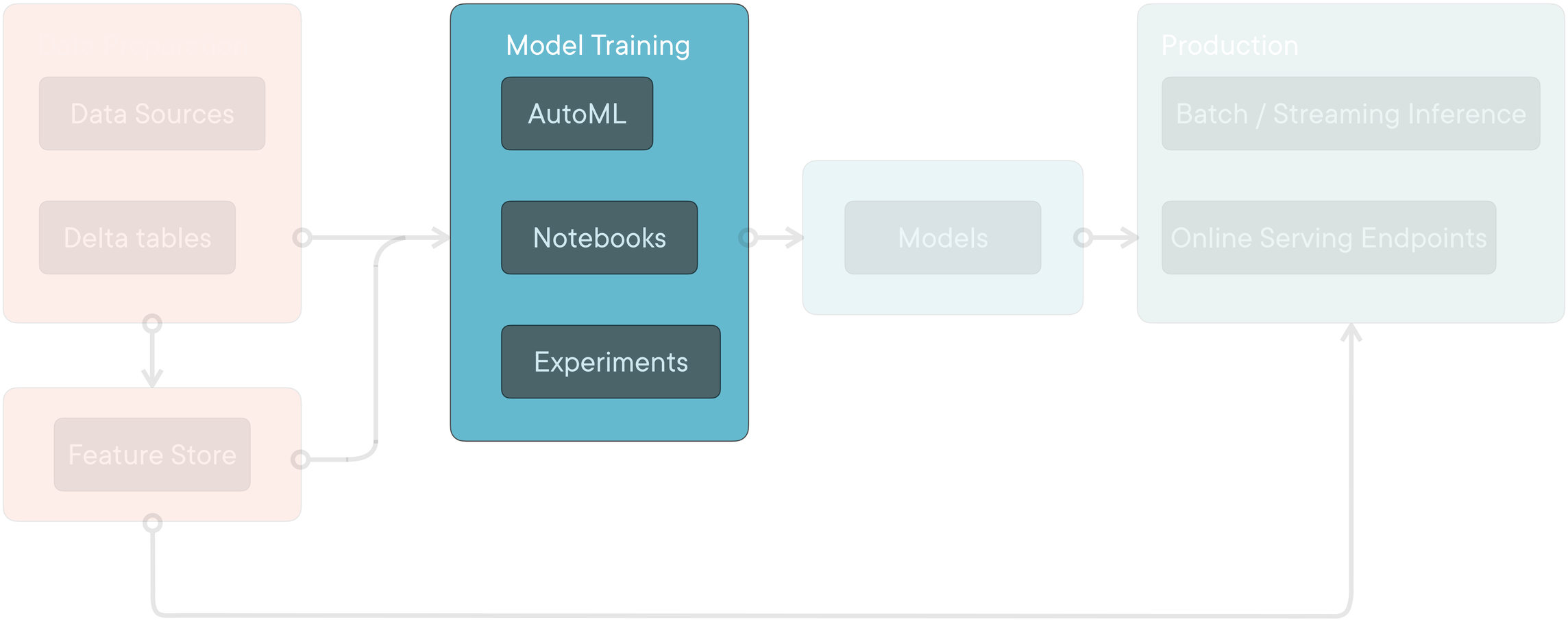
Delta tables offer transaction support, version control, and revision history for huge datasets

Feature Store



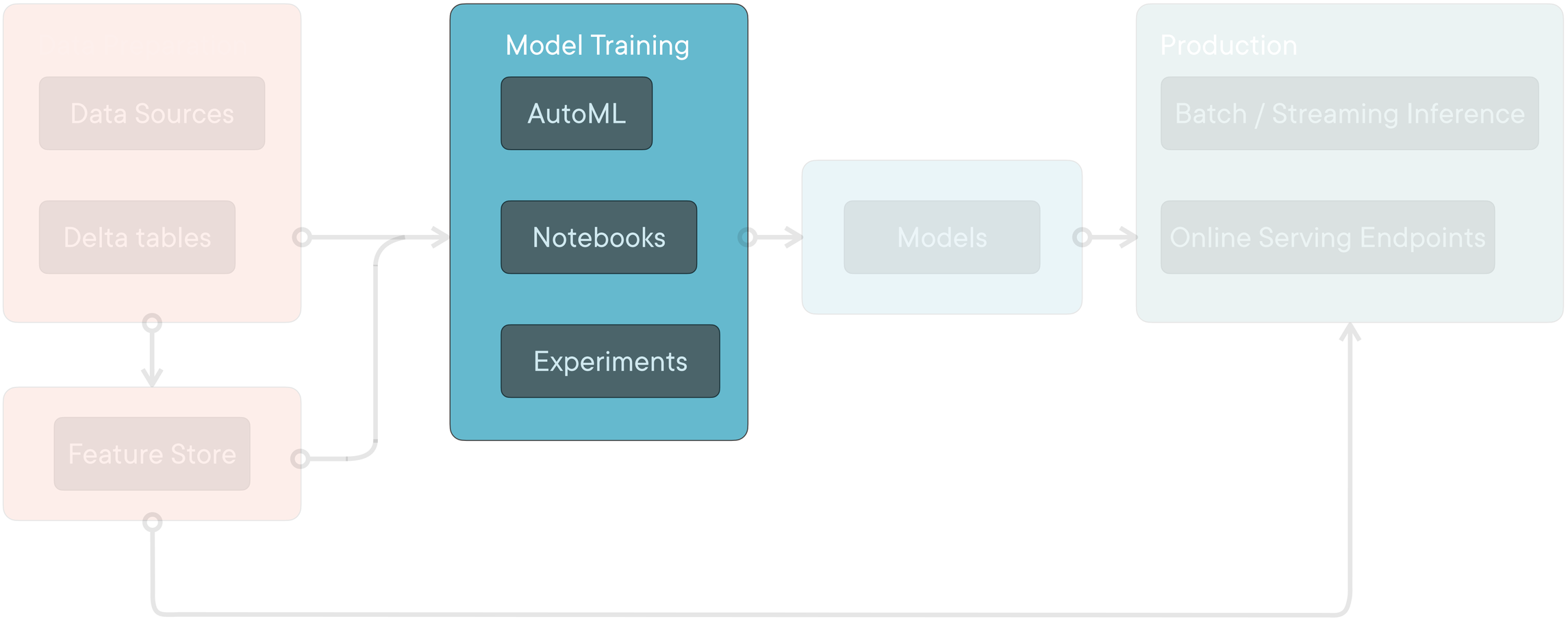
Feature tables in the feature store allows you to store processed features for model training and inference

Model Training



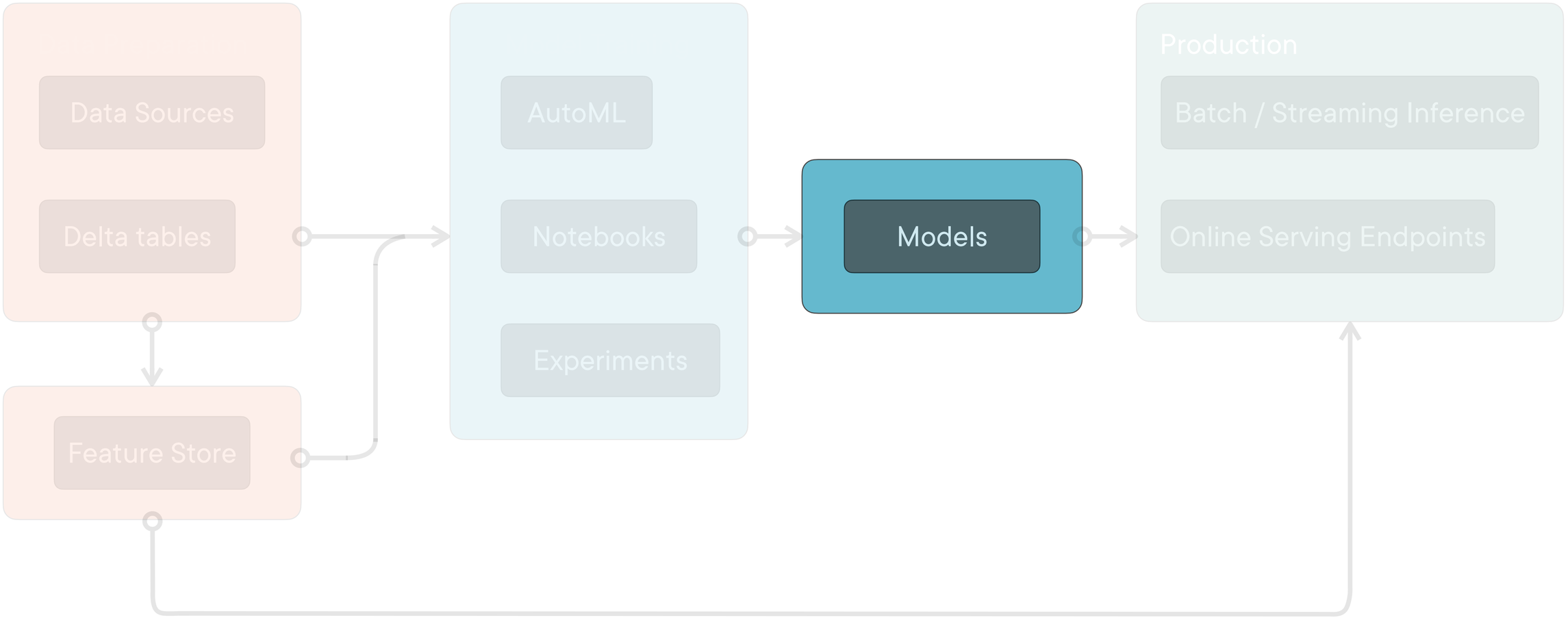
Model training can be performed using custom code or with AutoML

Model Training



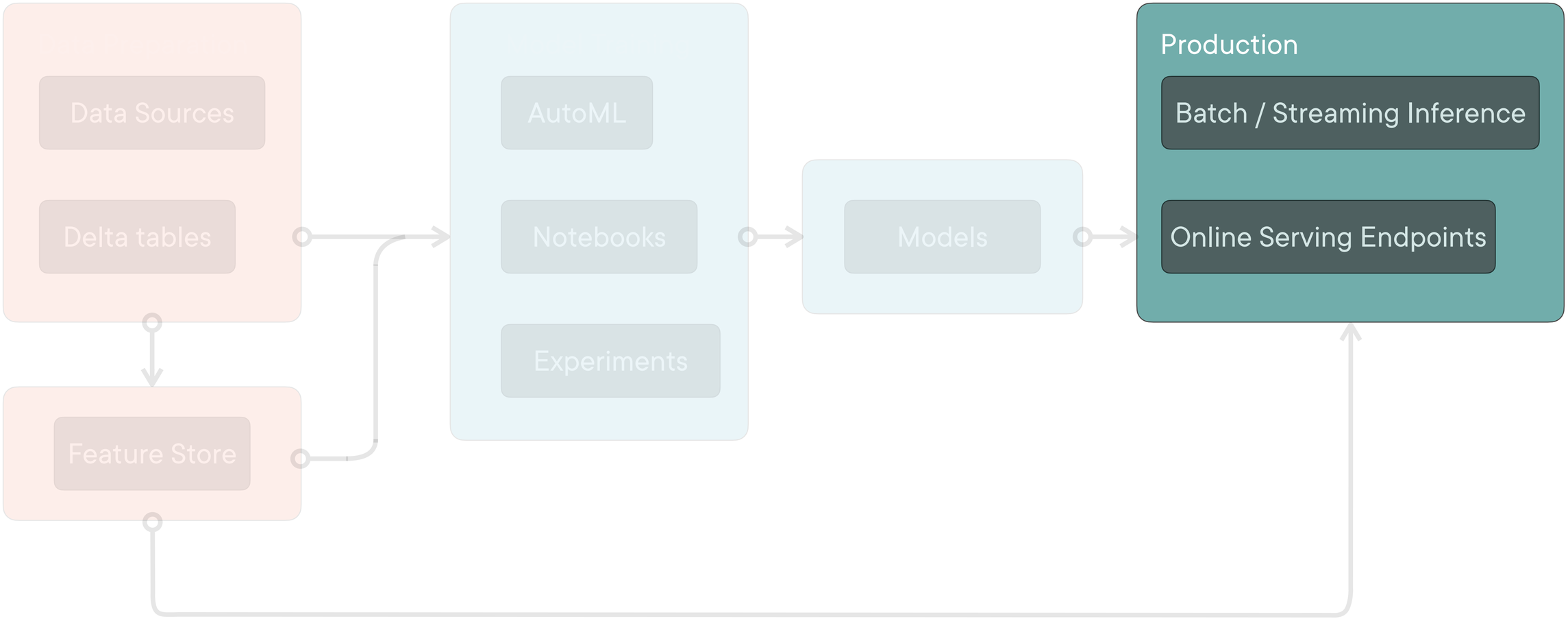
Track parameters and models using experiments with MLflow tracking

Model Registry



Share, manage, and serve models using the Model Registry

Model Inference



Deploy models to production and perform inference on batch as well as streaming data

Introducing MLflow

MLflow

Open-source platform for managing the end-to-end machine learning lifecycle which includes model tracking, a model registry, model serving and inference.

https://docs.databricks.com/mlflow/index.html

MLflow Components

Tracking

Models

Projects

Model Registry

Model Serving

MLflow Components

Tracking

Models

Projects

Model Registry

Model Serving

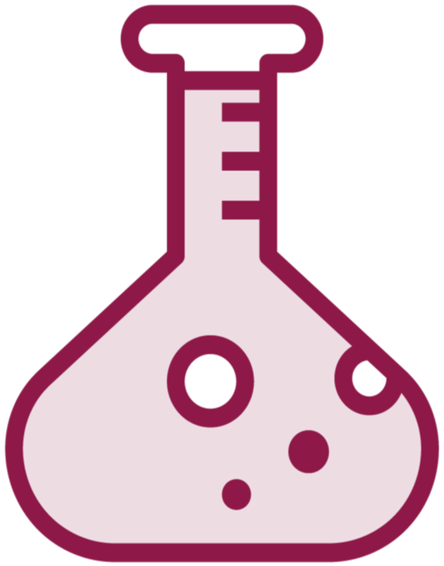
Model Tracking

API and UI for logging parameters, code versions, metrics, and output files

Tracking lets you log and query experiments

Supported technologies and languages include Python, REST, R API, and Java APIs

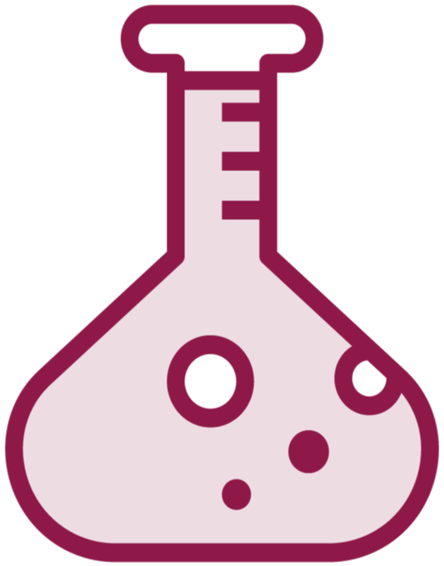
MLflow Tracking Concepts



Experiments

Runs

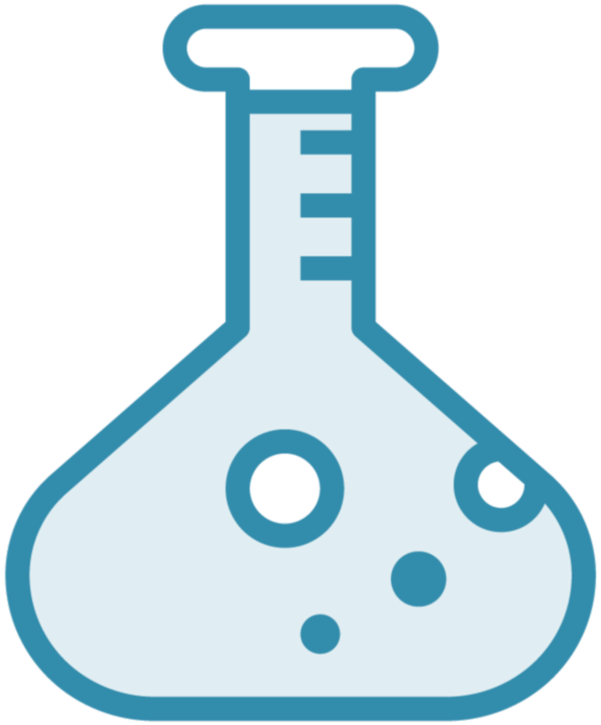
MLflow Tracking Concepts



Experiments

Runs

Experiments



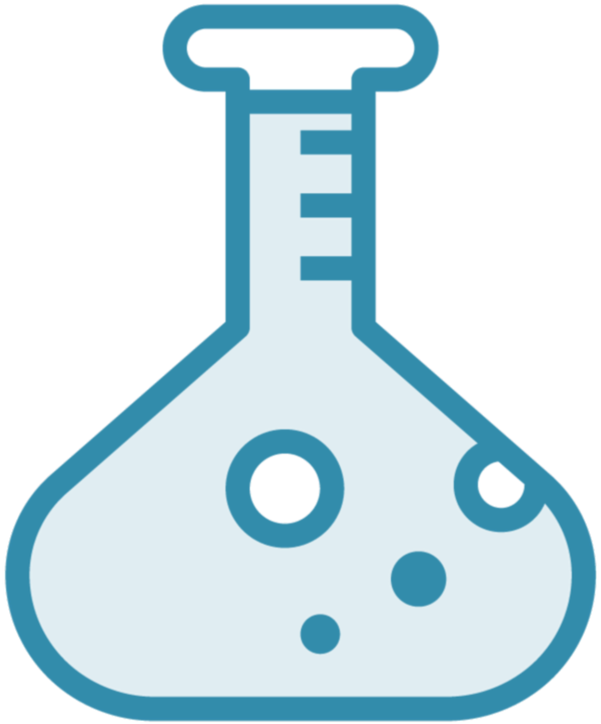
Primary unit of organization and access control for runs

Allows you to visualize, search for, and compare runs

Two types of experiments:

- Workspace experiment
- Notebook experiment

Experiments



Workspace experiment:

- Belongs to the workspace
- Not associated with a notebook
- Runs in any notebook can log to them

Notebook experiment:

- Associated with a specific notebook
- Automatically created if no active experiment for a run

MLflow Tracking Concepts

Experiments

Runs

Runs

Single execution of model code

Contains the following information:

- Notebook
- Version
- Start and end time
- Metrics
- Tags
- Artifacts

MLflow Components

Tracking

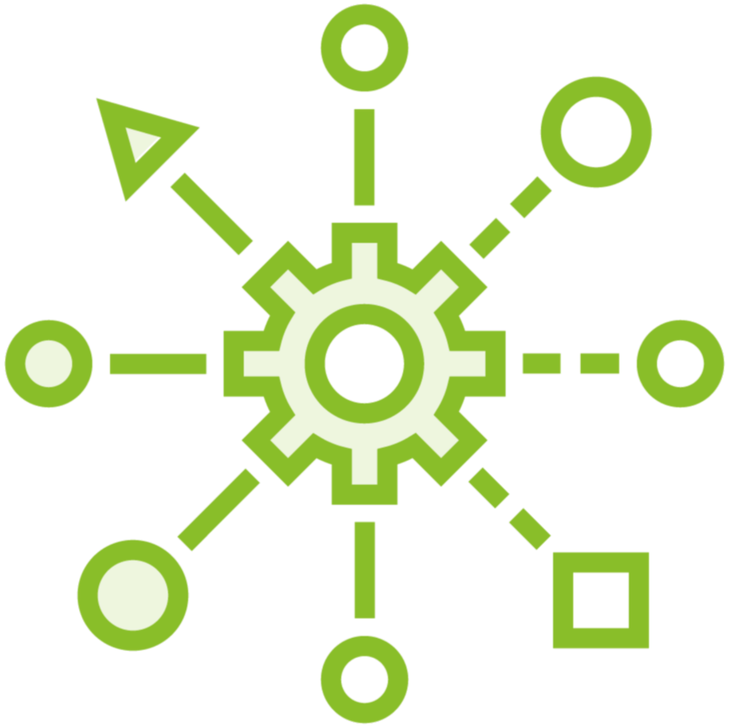
Models

Projects

Model Registry

Model Serving

Models and Model Serving



Manage and deploy models built using popular machine learning libraries

Supports multiple model serving and inference platforms

- Classic MLflow model serving using REST endpoints
- Serverless real-time inferencing allows scalable REST endpoints

MLflow Components

Tracking

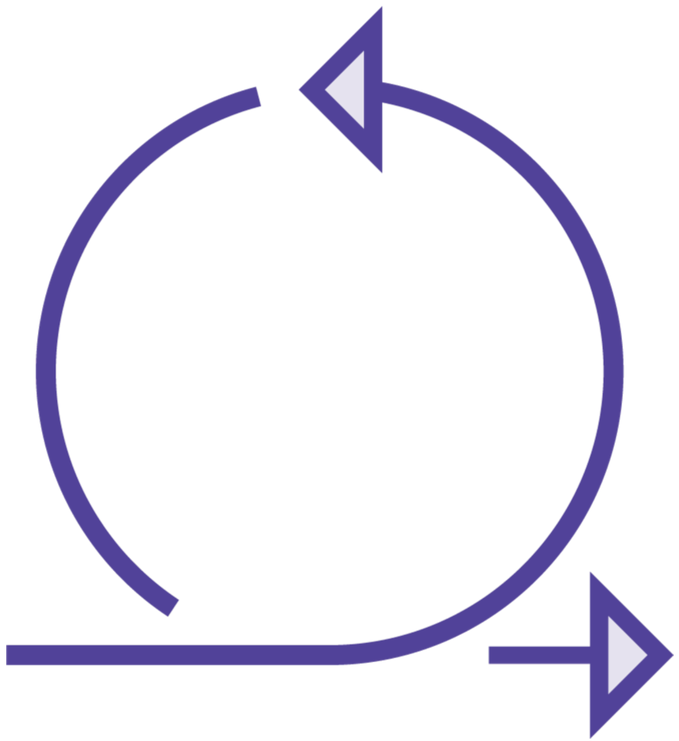
Models

Projects

Model Registry

Model Serving

Projects



Package ML code in a reusable and reproducible form

Allows sharing with other data scientists or transfers to production

MLflow Components

Tracking

Models

Projects

Model Registry

Model Serving

Model Registry

Centralized model store where models can be registered and managed

Manage lifecycle stage transitions from staging to production

Allows versioning and annotation of models

Summary

The Databricks ML runtime

MLflow to manage the ML workflow

Up Next:
Implementing Deep Learning Models Using TensorFlow and Keras