MLOps CD/CI for AI Models

How to manage the lifecycle of your models using MLFlow + Streamlit + FastAPI

MLOps workshop →

This workshop is presented for the audience of School of AI Algiers with ❤

February 8th, 2023

A bit about me ?

Final year CS engineering student

- ESI-ALGER (Algiers, Algeria)
- Computer systems
- Masters and state-engineering degrees at preparation

AI R&D research assistant

- LMCS-INFOLOGIC Engineering (Lyon, France)
- Working on predicting diffrent failures in datacenters and cloud systems using AI

Entrepreneurial kiddo

- Ex. dev team leader at ETIC Club
- Candidate for several engineer-entrepreneur trainings

esi ECOLE NATIONALE SUPÉRIEURE D'INFORMATIQUE

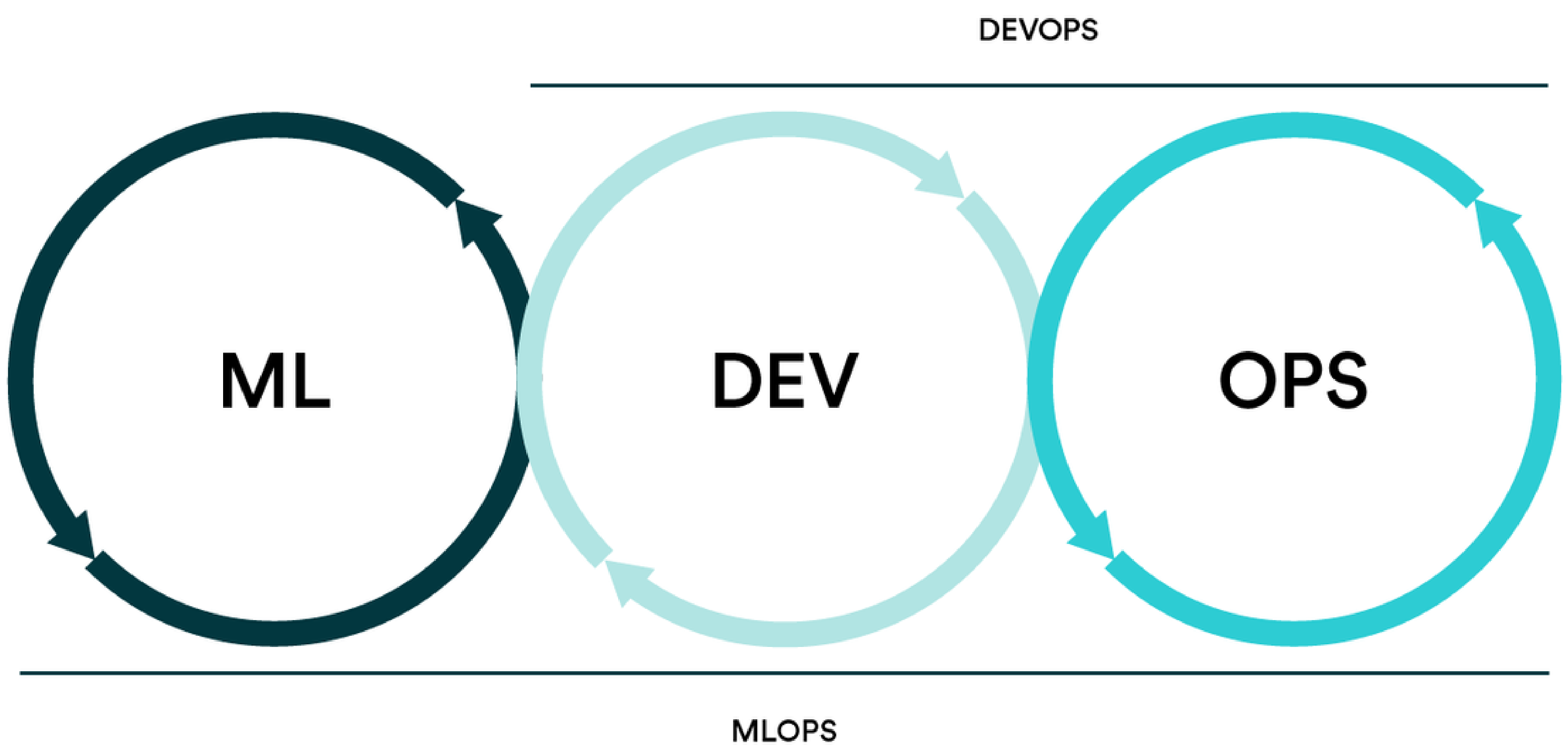
INFOLOGIC ENGINEERING

ETIC

Software dev is different from ML/AI dev ?!

Machine learning lifecycle managements is different from traditional software dev in terms of :

- Functional requirements
- Continuous integration goals
- Deployement strategies



ML DEV vs SOFT DEV

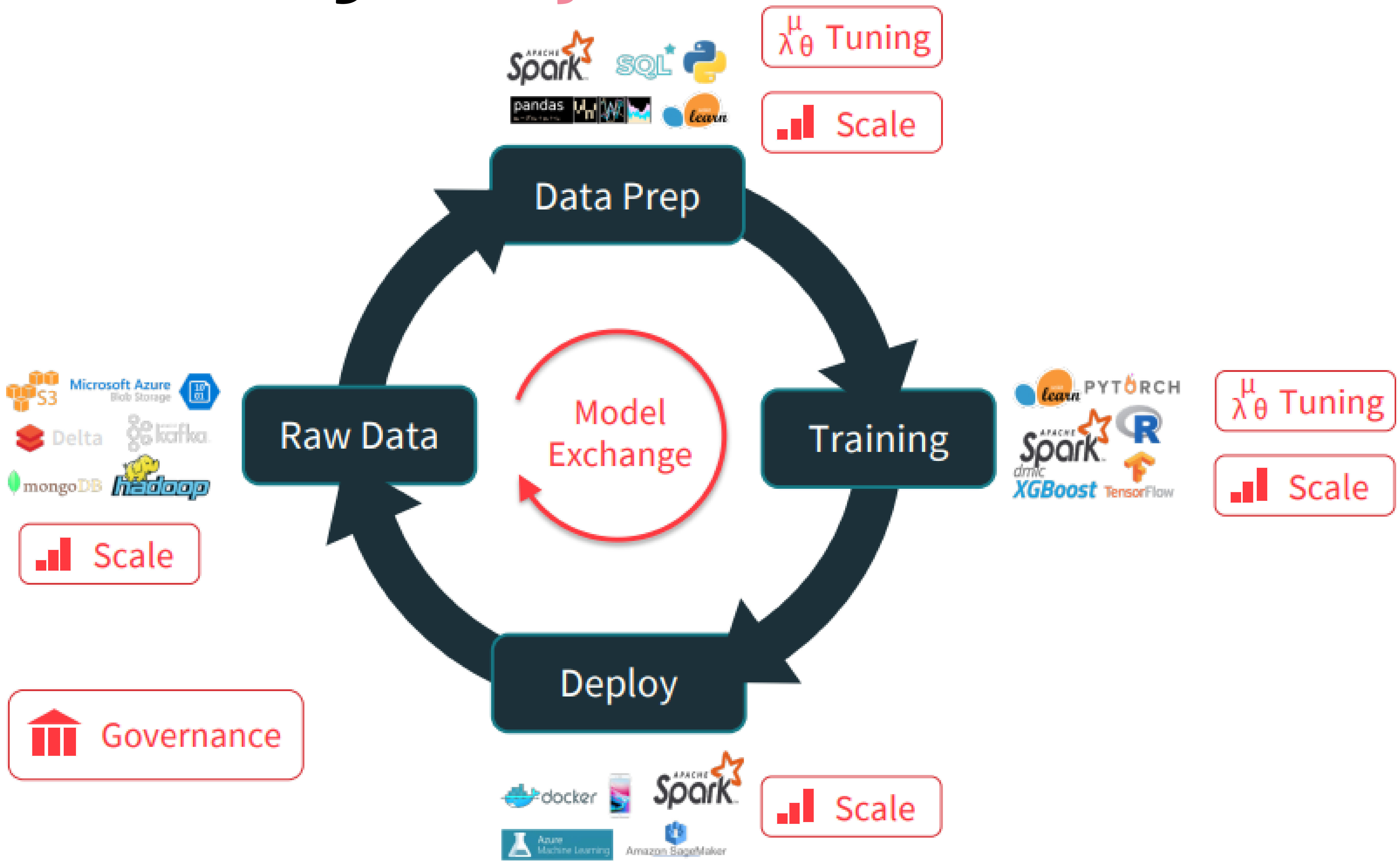
SOFT

- Goal: Meet a functional specification
- Quality depends only on code
- Typically pick one software stack w/ fewer libraries and tools
- Limited deployment environments

ML

- Goal: Optimize metric(e.g., accuracy. Constantly experiment to improve it
- Quality depends on input data and tuning parameters
- Compare + combine many libraries, model
- Diverse deployment environments

Machine learnifng Lifecycle



Can we do these **CD/CI tasks** without using a big company's paying product ?

mlflow

- Open machine learning platform
- Works with popular ML library & language
- Runs the same way anywhere (e.g., any cloud or locally)
- Designed to be useful for 1 or 1000+ person orgs
- Simple. Modular.Easy-to-use.
- Offers positive developer experience to get started!

How it works – Modular design

Allows different components	• You can use for example the tracking tools but not its deployement tools
Not monolithic	• Different components can run separatly and on different machines

How it works – API-first

Submit	• log models, metrics, runs, ...etc
Abstract lambda functions	• Enabling deployement on a variety of envs (Docker, Azure ML, Spark UDF, ...)
Open interface	• Easy to integrate within programmatic APIs, REST APIs & CLI

MLflow components

mlflow

Tracking

Record and query experiments: code, data, config, and results

mlflow

Projects

Package data science code in a format that enables reproducible runs on many platform

mlflow

Models

Deploy machine learning models in diverse serving environments

new

mlflow

Model Registry

Store, annotate and manage models in a central repository

MLflow **concepts**

Parameters: key-value inputs to your code

Metrics: numeric values (can update over time)

Tags and Notes: information about a run

Artifacts: files, data, and models

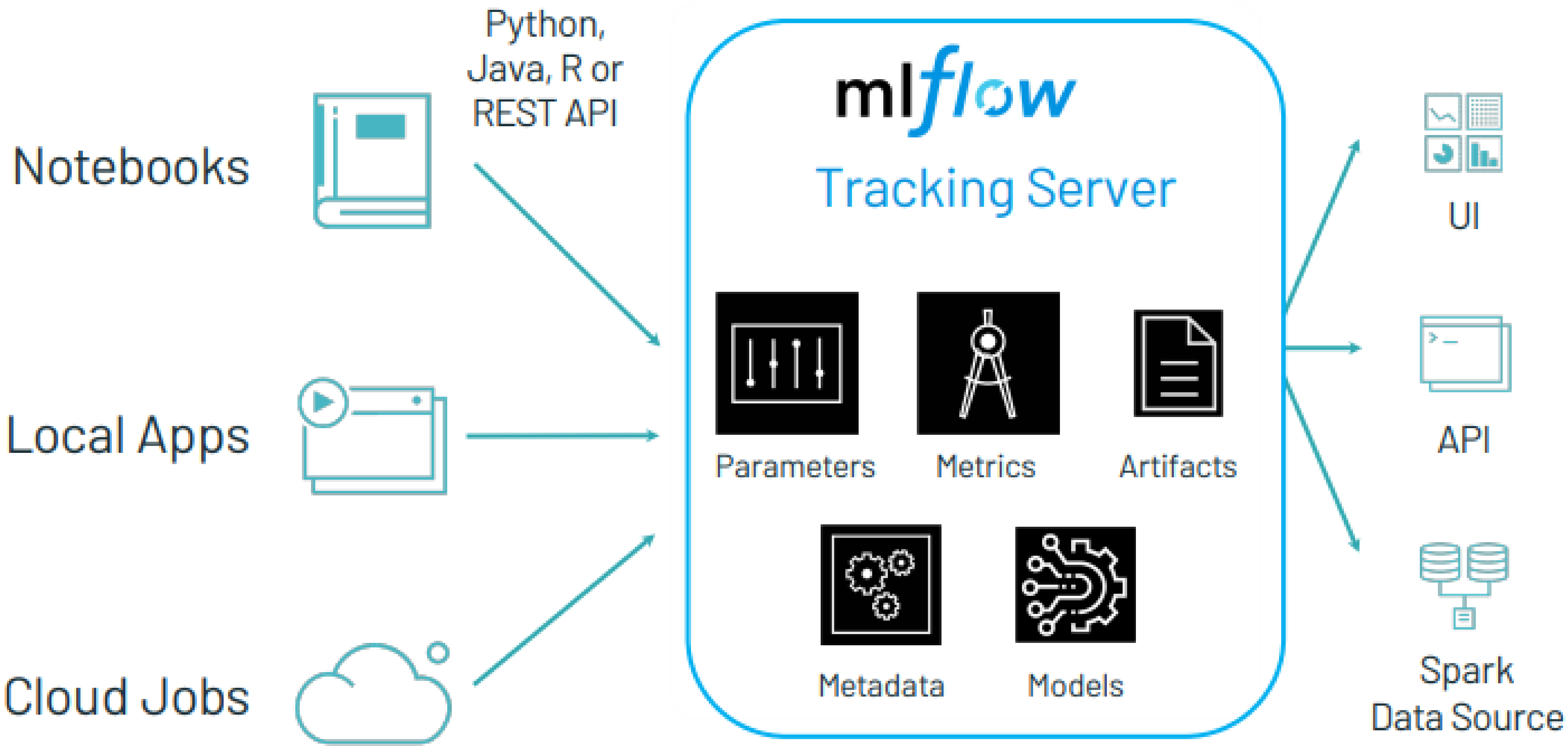
Source: what code ran?

Version: what of the code?

Run: an instance of code that runs by MLflow

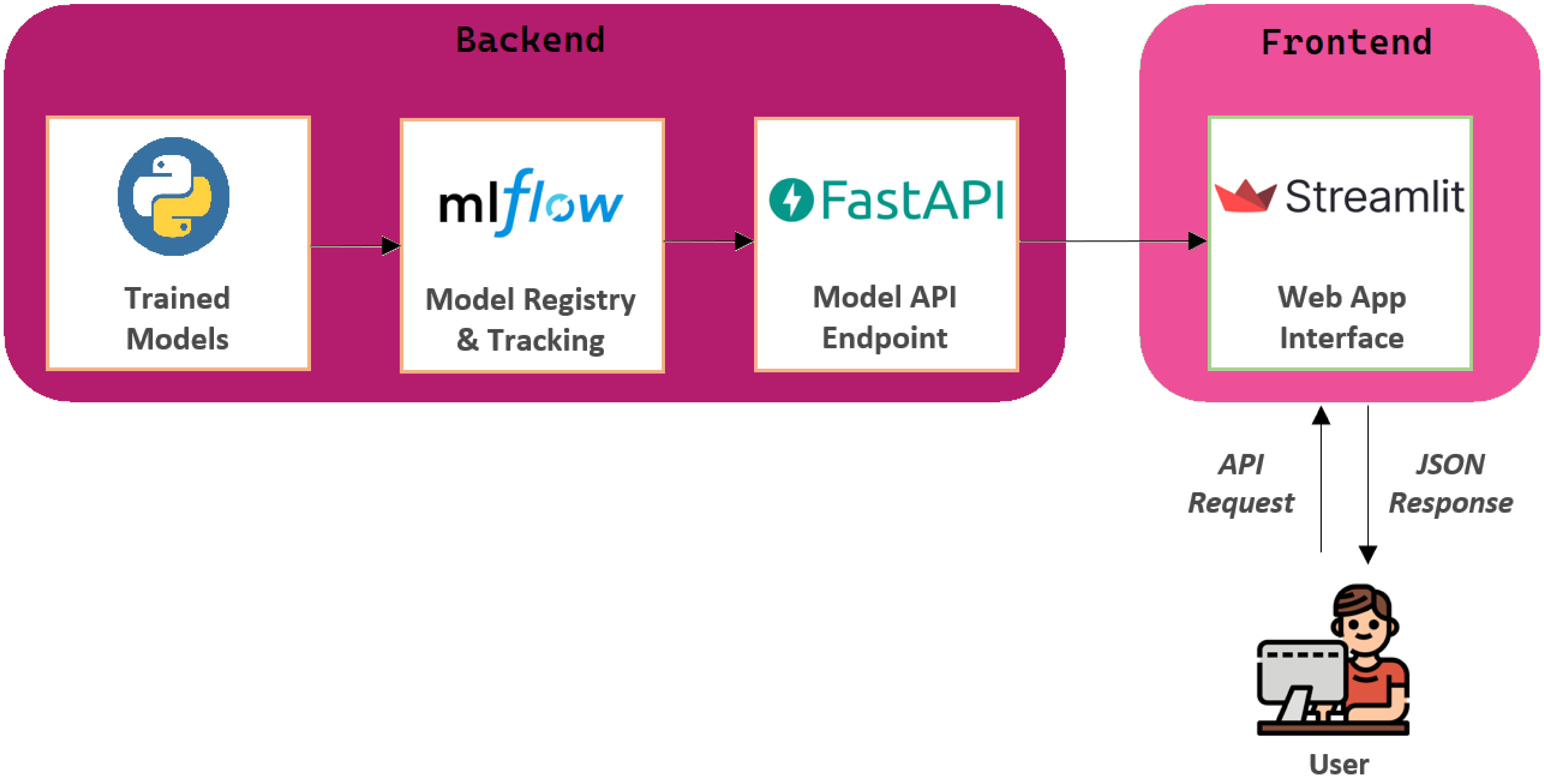
Experiment: {Run, ... Run}

MLflow tracking component




```
$ export MLFLOW_TRACKING_URI <URI>
mlflow.set_tracking_uri(URI)
```

MLops stack



Backend
Trained Models
mlflow
Model Registry & Tracking
FastAPI
Model API Endpoint
Frontend
Streamlit
Web App Interface
API Request
JSON Response
User

Classic ml models vs MLflow model

```
data = load_text(file)
ngrams = extract_ngrams(data, N=n)
model = train_model(ngrams,
learning_rate=lr)
score = compute_accuracy(model)
print(“For n=%d, lr=%f: accuracy=%f”
% (n, lr, score))
pickle.dump(model, open(“model.pkl”))
```

```
import mlflow
data = load_text(file)
ngrams = extract_ngrams(data, N=n)
model = train_model(ngrams,
learning_rate=lr)
score = compute_accuracy(model)
with mlflow.start_run():
mlflow.log_param(“data_file”, file)
mlflow.log_param(“n”, n)
mlflow.log_param(“learn_rate”, lr)
mlflow.log_metric(“score”, score)
mlflow.sklearn.log_model(model)
```

Thanks!

Questions ?

CODE TIME
FELLAS !