mlflow Platform for Complete Machine Learning Lifecycle

Jules S. Damji
@2twitme

San Francisco | May 6, 2020: Part 2 of 3 Series

databricks

$ whoami

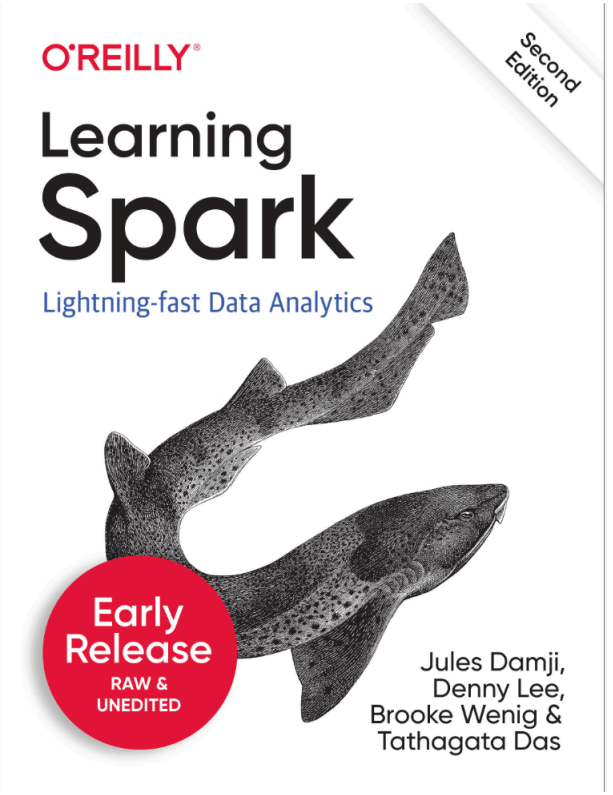
Apache Spark Developer & Community Advocate @ Databricks

Developer Advocate @ Hortonworks

Software engineering @ Sun Microsystems, Netscape, @Home, Excite@Home, VeriSign, Scalix, Centrify, LoudCloud/Opsware, ProQuest

Program Co-chair Spark + AI Summit

https://www.linkedin.com/in/dmatrix
@2twitme



databricks

Who is Databricks?

- Databricks is the data and AI company.

What do we do?

- We simplify data and AI so data teams can innovate faster.

How do we do it?

- We provide an open and unified platform for data engineering, machine learning and analytics

What is our mission?

- Our mission is to help data teams solve the world's toughest problems

APACHE Spark™

databricks

DELTA LAKE

mlflow™

Koalas

Outline – Part 2

- Recap Part 1
- Concepts and Motivations
- MLFlow Components
  - MLflow Projects & Models
  - Use MLflow on localhost
  - MLflow CLI & API on localhost
  - MLflow UI on localhost
- Q & A

Traditional Software vs. Machine Learning

Traditional Software

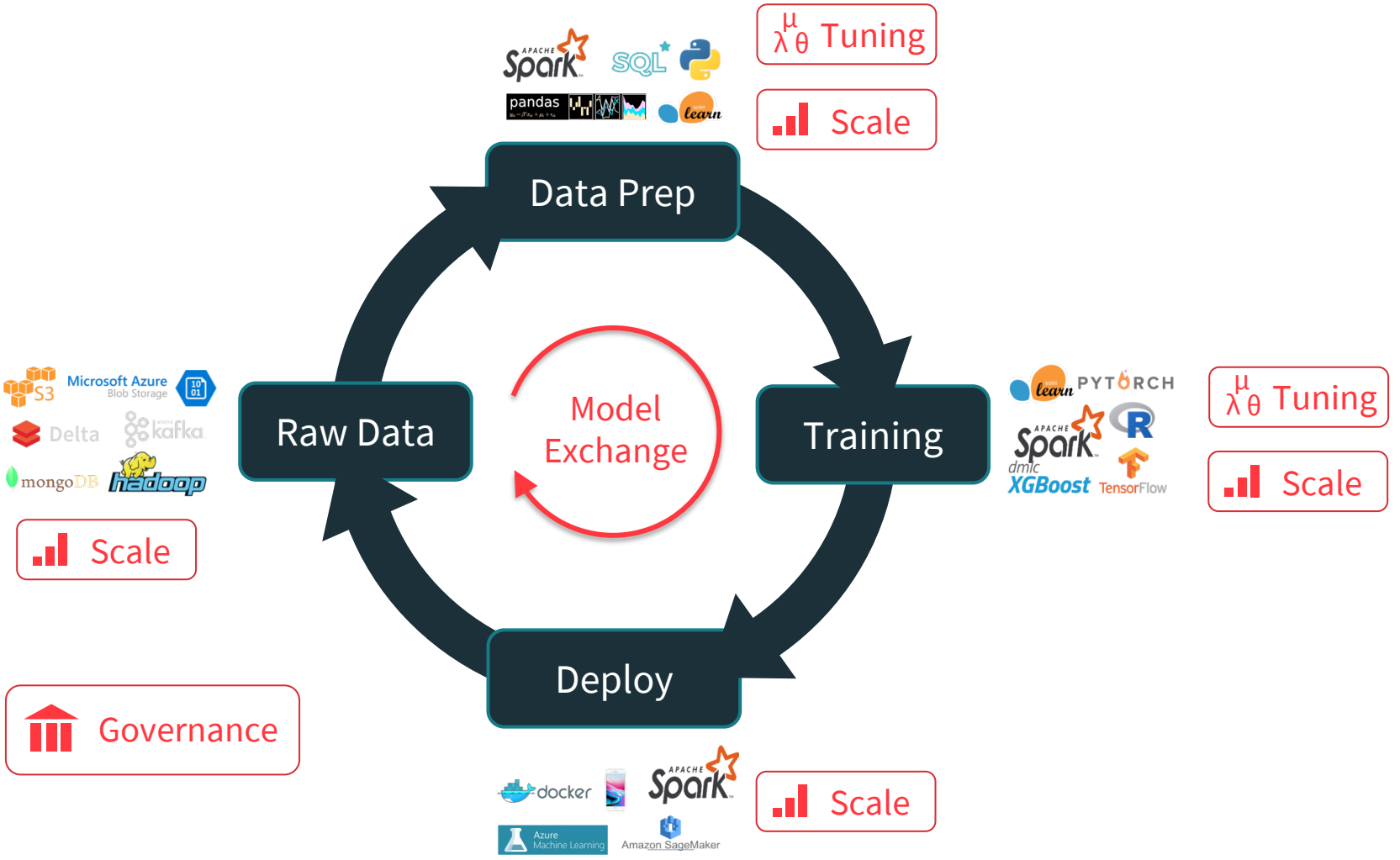
- Goal: Meet a functional specification
- Quality depends only on code
- Typically pick one software stack w/ fewer libraries and tools

Machine Learning

- Goal: Optimize metric(e.g., accuracy. Constantly experiment to improve it
- Quality depends on input data and tuning parameters

- Compare + combine many libraries, model

Machine Learning Lifecycle



databricks

MLflow Design Philosophy

API-First

- Submit runs, log models, metrics, etc. from popular library & language
- Abstract “model” lambda function that MLflow can then deploy in many places (Docker, Azure ML, Spark UDF)
- Open interface allows easy integration from the community

Key enabler: built around Programmatic APIs, REST APIs & CLI

Modular Design

- Allow different components individually (e.g., use MLflow’s project format but not its deployment tools
- Not monolithic
- But Distinctive and Selective

Key enabler: distinct components (Tracking/Projects/Models/Registry)

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 any platform

mlflow

Models

Deploy machine learning models in diverse serving environments environments

new

mlflow

Model Registry

Store, annotate and manage models in a central repository

databricks.com/mlflow

mlflow.org

github.com/mlflow

twitter.com/MLflow

databricks

Key Concepts in MLflow Tracking

Parameters: key-value inputs to your code

Metrics: numeric values (can update over time)

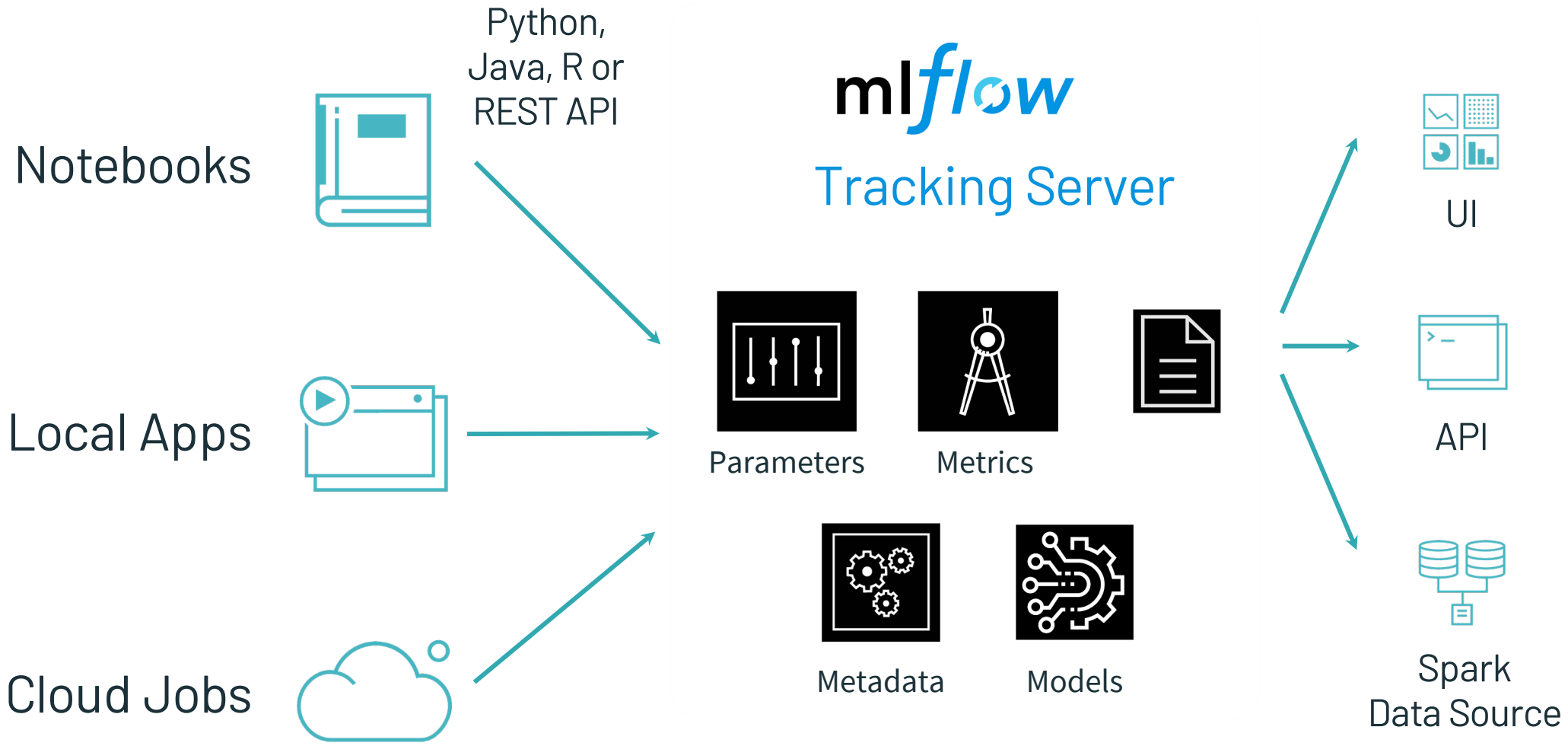
Tags and Notes: information about a run

Artifacts: files, data, and models

Source: what code ran?

Version: what of the code?

MLflow Tracking




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

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 any platform

mlflow

Models

Deploy machine learning models in diverse serving environments environments

new

mlflow

Model Registry

Store, annotate and manage models in a central repository

databricks.com/mlflow

mlflow.org

github.com/mlflow

twitter.com/MLflow

databricks

MLflow Projects Motivation

Diverse set of tools

PYTORCH

dmlc XGBoost

TensorFlow

scikit learn

APACHE Spark™

R

Diverse set of environments

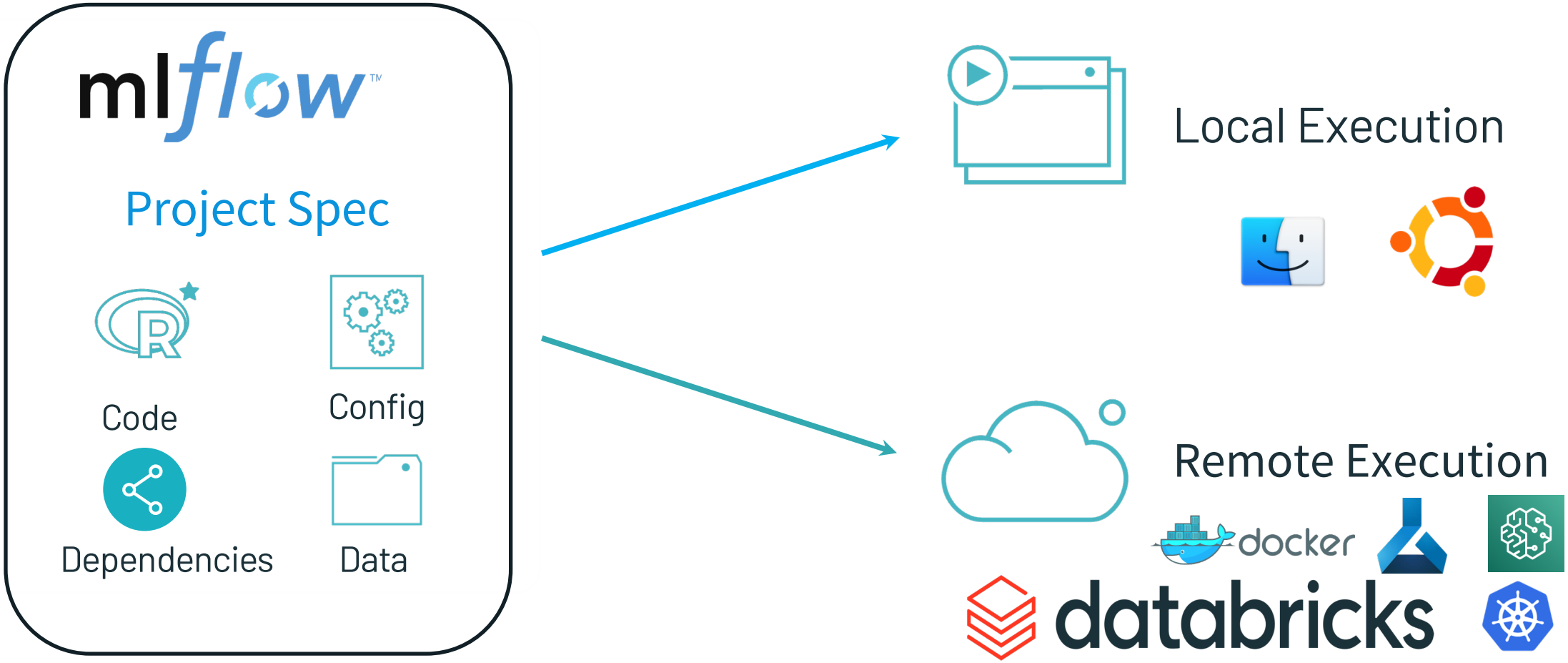
docker

Azure Machine Learning

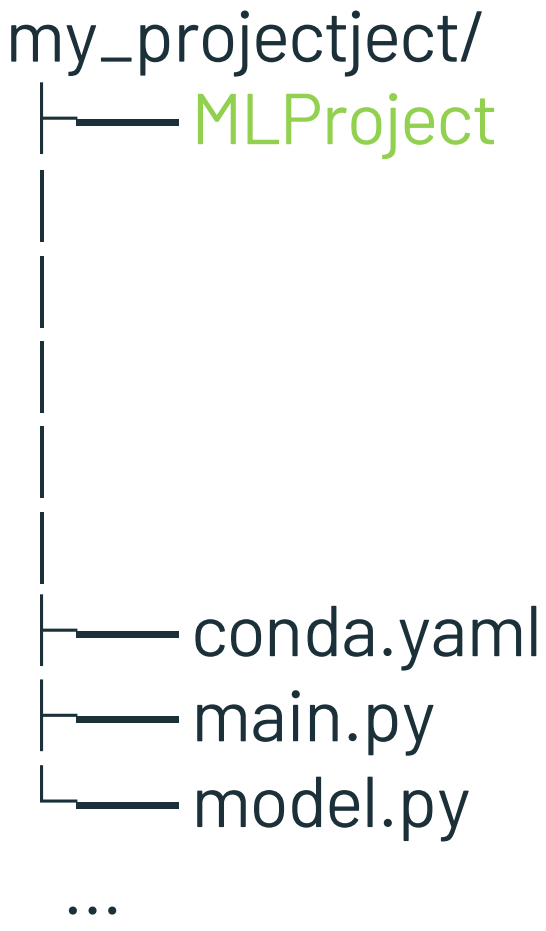
databricks

Challenge: ML results difficult to reproduce

MLflow Projects



Example MLflow Project




```
conda_env: conda.yaml

entry_points:
  main:
    parameters:
      training_data: path
      lambda: {type: float, default: 0.1}
    command: python main.py {training_data} {lambda}
```

```
$ mlflow run git://<my_project> -P lambda=0.2
```

```
mlflow.run("git://<my_project>", ...)
```

```
mlflow run . -e main -P lambda=0.2
```

Example

```
my_project/
├── MLproject
│
│
│
│
│
│
├── conda.yaml
├── main.py
└── model.py
  ...
```

```
channels:
- defaults
dependencies:
- python=3.7.3
- scikit-learn=0.20.3
- pip:
  - mlflow
  - cloudpickle==0.8.0
name: mlflow-env
```

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 any platform

mlflow

Models

Deploy machine learning models in diverse serving environments environments

new

mlflow

Model Registry

Store, annotate and manage models in a central repository

databricks.com
/mlflow

mlflow.org

github.com/mlflow

twitter.com/MLflow

databricks

MLflow Model Motivations

ML Frameworks

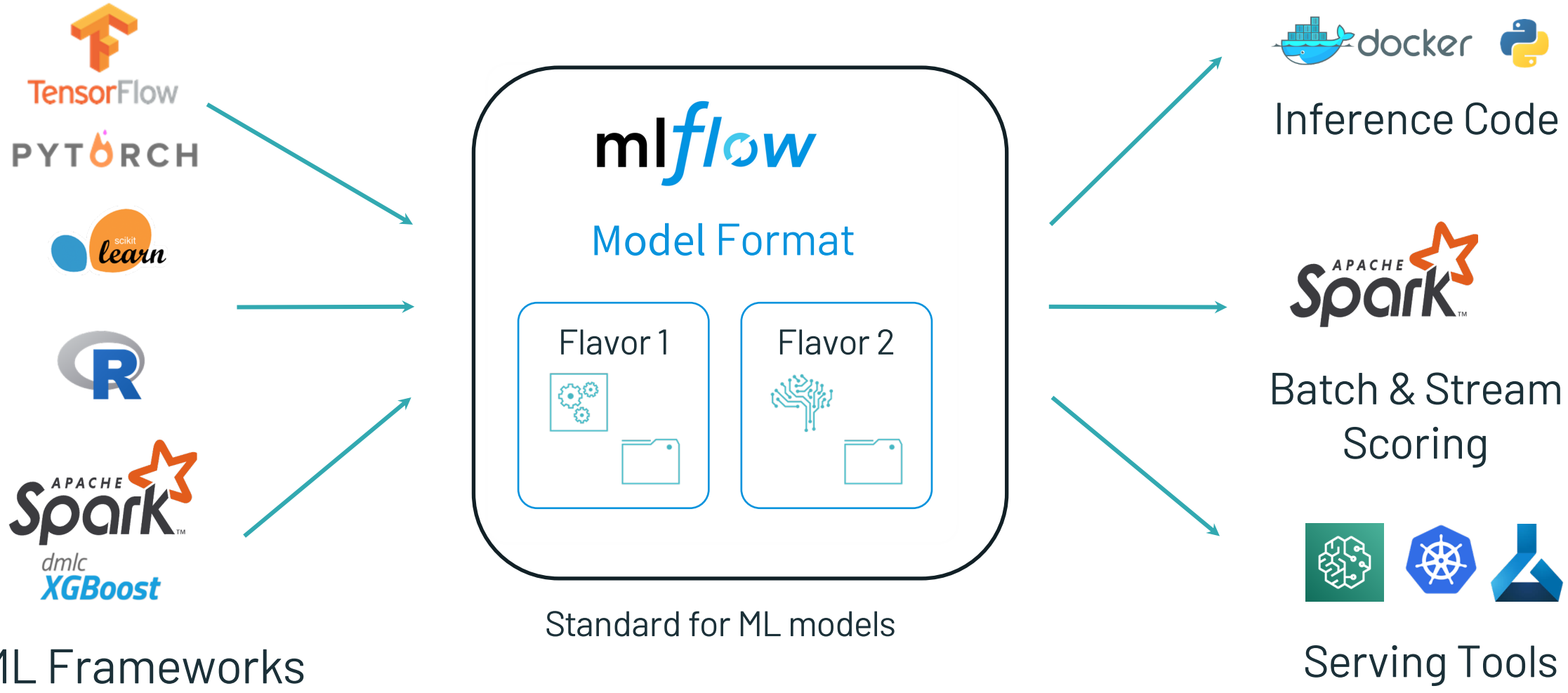
N x M
Combination of
Model support for
all Serving tools

Inference Code

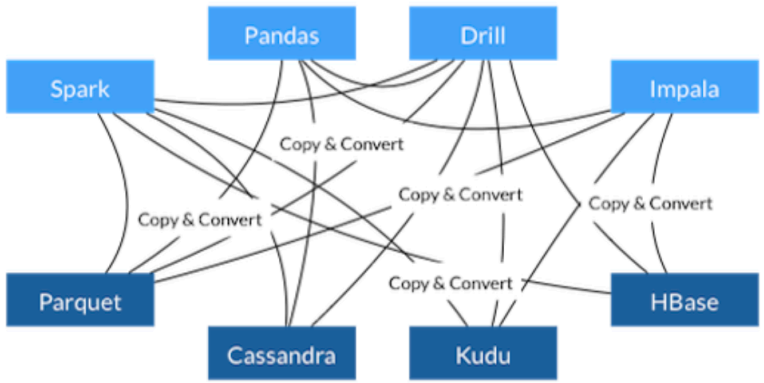
Batch & Stream Scoring

Serving Tools

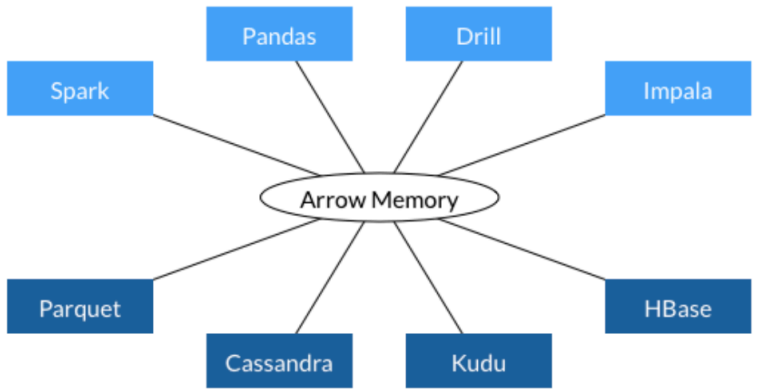
MLflow Models



Advantages of a Common Data Layer



- Each system has its own internal memory format
- 70-80% computation wasted on serialization and deserialization
- Similar functionality implemented in multiple projects



- All systems utilize the same memory format
- No overhead for cross-system communication
- Projects can share functionality (eg, Parquet-to-Arrow reader)

Example MLflow Model

```
mlflow.tensorflow.log_model(...)
```

```
my_model/
├── MLmodel
└── estimator/
    ├── saved_model.pb
    └── variables/
        ...
```

```
run_id: 769915006efd4c4bbd662461
time_created: 2018-06-28T12:34
flavors:
  tensorflow:
    saved_model_dir: estimator
    signature_def_key: predict
  python_function:
    loader_module: mlflow.tensorflow
```

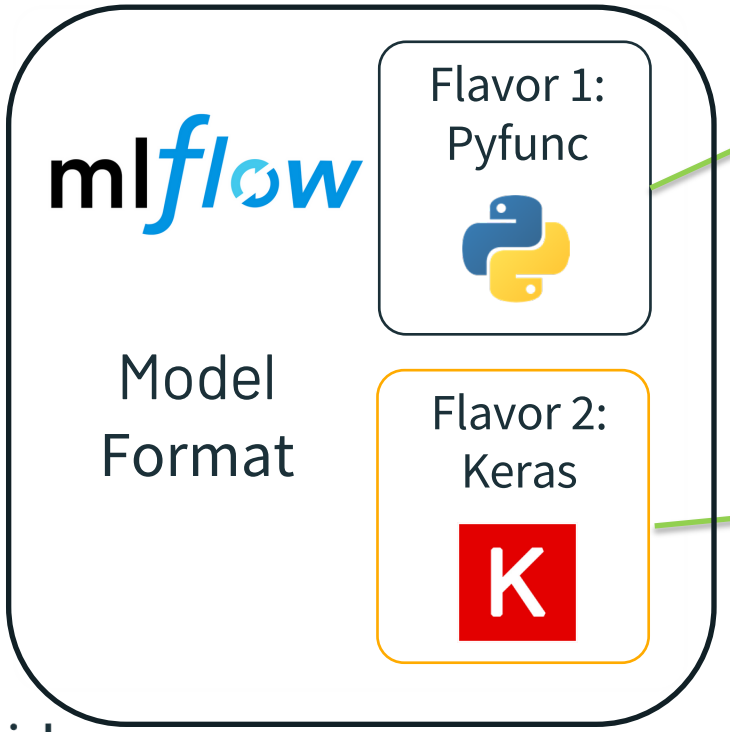
Usable by tools that understand TensorFlow model format

Usable by any tool that can run Python (Docker, Spark, etc!)

Model Flavors Example

Train a model

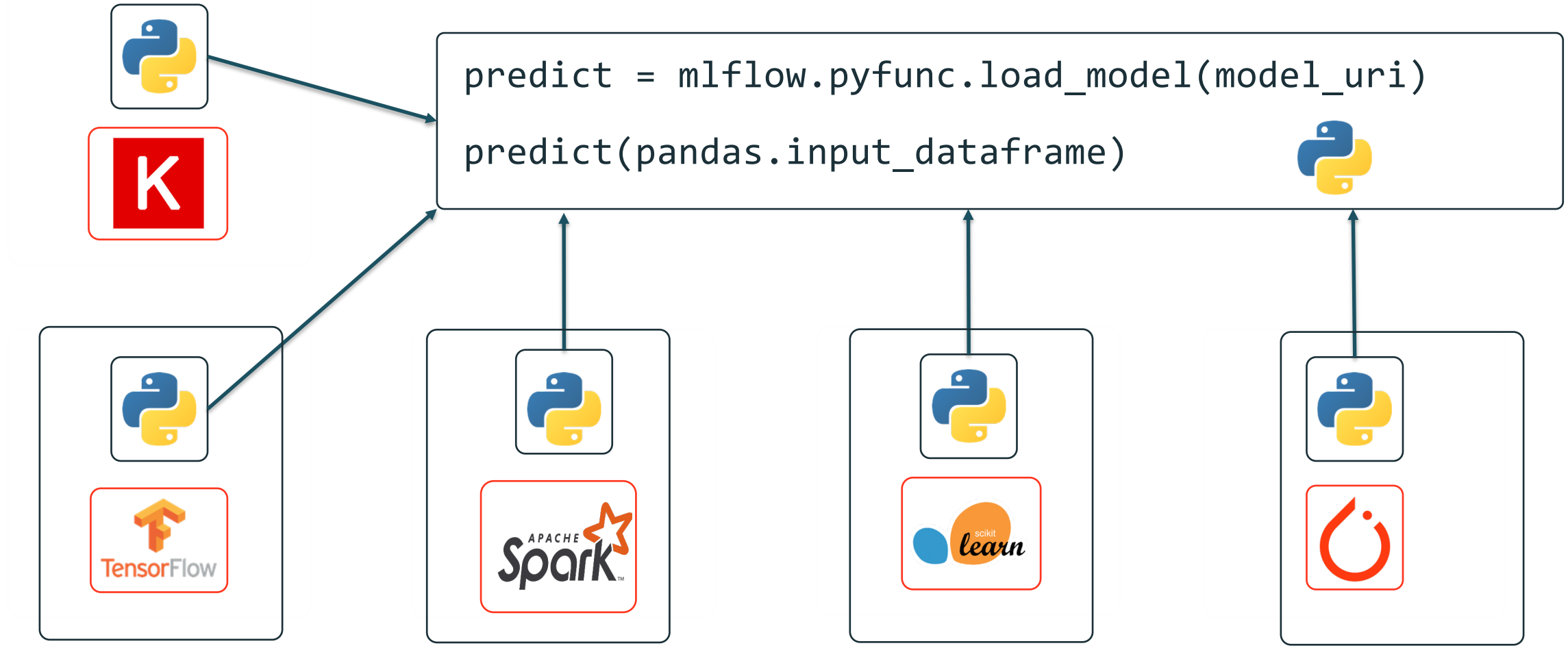
`mlflow.keras.log_model(…)`




```
predict = mlflow.pyfunc.load_model(…)
predict(pandas.input_dataframe)
```

```
model = mlflow.keras.load_model(…)
model.predict(keras.Input(…))
```

Model Flavors Example



Learning More About MLflow

- pip install mlflow to get started
- Find docs & examples at mlflow.org
- Peruse code at Mlflow Github
- Join the Slack channel
- More MLflow tutorials

MLflow Project & Models Tutorials

https://github.com/dmatrix/mlflow-workshop-part-2

Thank you! ☺

Q & A

jules@databricks.com
@2twitme
https://www.linkedin.com/in/dmatrix/