

MLflow: Platform for Machine Learning Lifecycle

Machine Learning Architects Basel

Bassem Ben Hamed
December 2022





Agenda

- Introduction to MLflow
- MLflow Components
- MLflow Tracking
- MLflow Projects
- MLflow Models
- MLflow Model Registry



Introduction to MLflow

Open machine learning platform

- Works with popular ML library and language
- Runs the same way anywhere (e.g. any cloud or locally)
- Designed to be useful for all persons on your organisation
- Simple. Modular. Easy-to-use
- Offers positive developer experience to get started!

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

mlflow Model Registry

Store, annotate and manage models in a central repository

Key Concepts in MLflow Tracking

- Parameters: Key-value inputs to your code
- Metrics: numerics 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}

Model Development with MLflow is Simple

```
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)
```

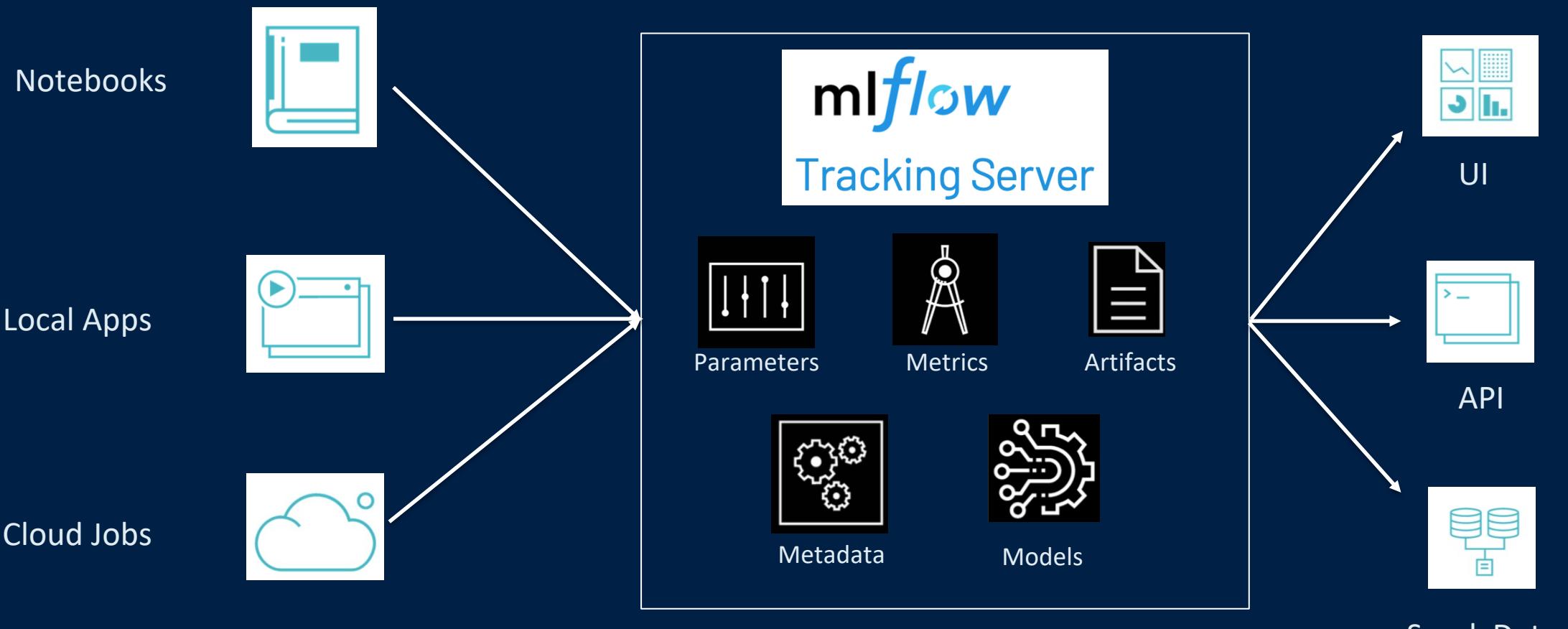
\$ mlflow ui

The screenshot shows the MLflow UI interface. At the top, there's a search bar with the query "metrics.rmse < 1 and params.model == 'tree'". Below it, there are filter parameters for "alpha, lr" and a "Filter Metrics" dropdown set to "rmse, r2". A "Search" button is also present. The main area displays a table of 36 matching runs. The columns in the table are Date, User, Source, Version, Parameters, and Metrics. The Parameters column includes alpha, l1_ratio, mae, r2, and rmse. The Metrics column includes rmse and r2. The table shows multiple runs from July 17, 2018, with various parameter values and metric scores.

Date	User	Source	Version	Parameters	Metrics
2018-07-19 03:26:53	root	azure-demo1	0.01	0.55 0.596 0.25 0.762	rmse r2 0.25 0.762
2018-07-19 03:26:39	root	azure-demo	0.01	0.55 0.596 0.25 0.762	rmse r2 0.25 0.762
2018-07-19 03:26:14	root	azure-demo	0.01	0.55 0.596 0.25 0.762	rmse r2 0.25 0.762
2018-07-19 03:25:51	root	azure-demo	0.01	0.75 0.597 0.25 0.762	rmse r2 0.25 0.762
2018-07-19 03:25:42	root	azure-demo	0.01	0.04 0.591 0.256 0.759	rmse r2 0.256 0.759
2018-07-18 02:09:54	root	azure-demo	0.01	1.0 0.597 0.249 0.762	rmse r2 0.249 0.762
2018-07-18 02:09:29	root	azure-demo	0.01	0.75 0.597 0.25 0.762	rmse r2 0.25 0.762
2018-07-18 02:08:52	root	azure-demo	0.01	0.01 0.591 0.257 0.759	rmse r2 0.257 0.759
2018-07-17 08:13:37	root	azure-demo	0.01	0.01 0.591 0.257 0.759	rmse r2 0.257 0.759
2018-07-17 08:13:34	root	azure-demo	0.01	1.0 0.597 0.249 0.762	rmse r2 0.249 0.762
2018-07-17 08:13:30	root	azure-demo	0.01	0.75 0.597 0.25 0.762	rmse r2 0.25 0.762
2018-07-17 08:13:27	root	azure-demo	0.01	0.01 0.591 0.257 0.759	rmse r2 0.257 0.759
2018-07-17 08:08:05	root	azure-demo	0.01	0.01 0.591 0.257 0.759	rmse r2 0.257 0.759

Track parameters, metrics, output files and code version

MLflow Tracking



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

MLflow Projects Motivation

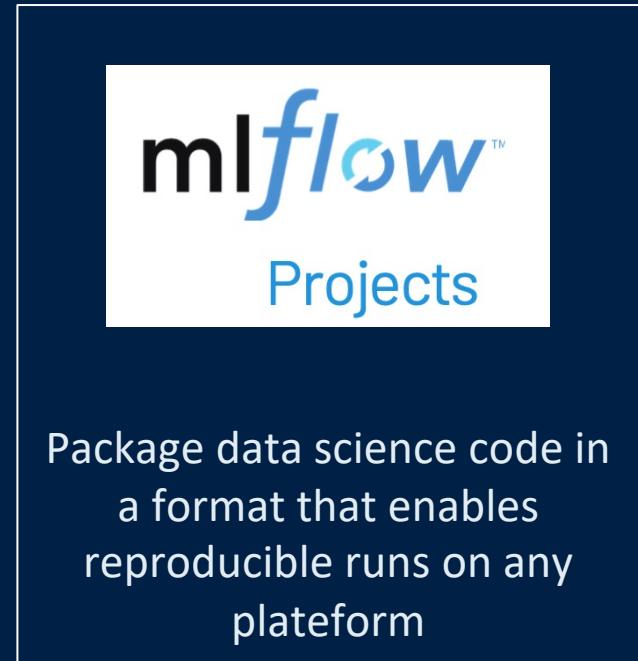
Diverse set of tools



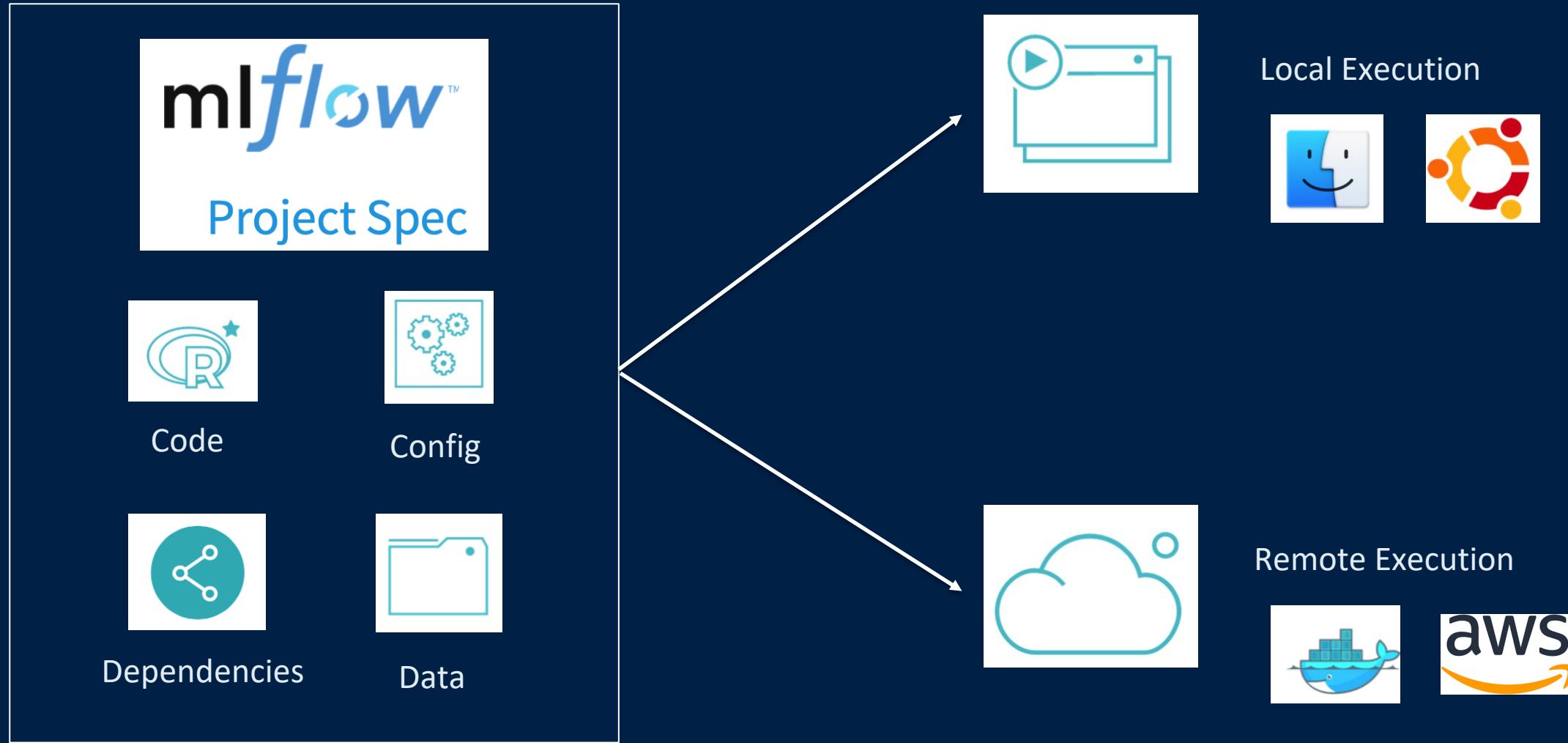
Diverse set of environments



Challenge:
ML results difficult to reproduce



MLflow Projects



MLflow Projects

Packaging format for reproducible ML runs

- Any code folder or Dagshub repository MLProject file with project configuration

Defines dependencies for reproducibility

- Conda dependencies can be specified in MLProject
- Reproducible in any environment

Execution API for running projects

- Python / R
- Supports local and remote execution
 - `mlflow run -help`
 - `mlflow run https://dagshub.com/mlflow-project`
 - `mlflow run (<project_uri>, parameters = {})`

MLflow Models



ML Frameworks



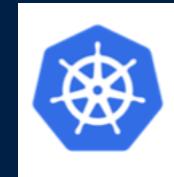
N x M Combination of
Model support for all
Serving tools



Inference Code



Batch and Stream Scoring



Serving Tools

MLflow Models

Packaging format for ML Models

- Any directory with MLmodel file

Defines dependencies for reproducibility

- Conda environment can be specified in MLmodel configuration

Model creation and loading utilities

- mlflow.<model_flavor>.save_model(...) or log_model(...)
- mlflow.<model_flavor>.load_model(...)

Deployment APIs

- Python / R
- mlflow models [OPTIONS] COMMAND [ARGS] ...
 - mlflow models serve [OPTIONS [ARGS] ...]
 - mlflow models predict [OPTIONS [ARGS] ...]

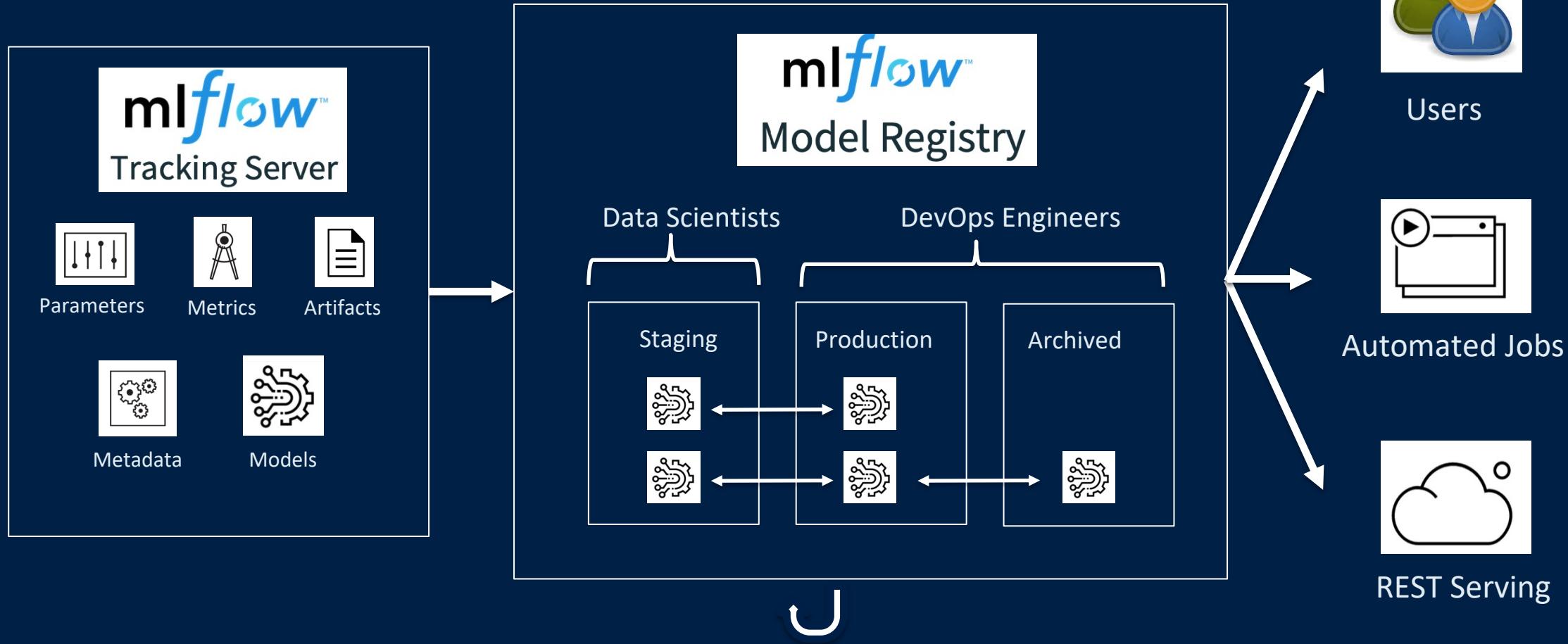
Model Management Problem

When you work in a large organization with many models, many data teams, management becomes a major challenge:

- Where can I find the best version of this model?
- How was this model trained?
- How can I track docs for each model?
- How can I review models?
- How can I integrate with CI/CD?

MLflow Model Registry

Vision: Centralized and collaborative model lifecycle management



MLflow Model Registry

- Repository of named, versioned models with comments and tags
- Track each model's stage: none, staging, production, or archived
- Easily inspect a specific version and its run info
- Easily load a specific version
- Provides model description, lineage and activities

Learning More About MLflow

- pip install mlflow – to get started
- Find documents and examples at mlflow.org
- Peruse code and contribute at github.com/mlflow/mlflow