DATA+AI SUMMIT 2022

mlflow™

MLflow Pipelines

Accelerating MLOps from Development to Production

Xiangrui Meng
Principal Software Engineer
Databricks

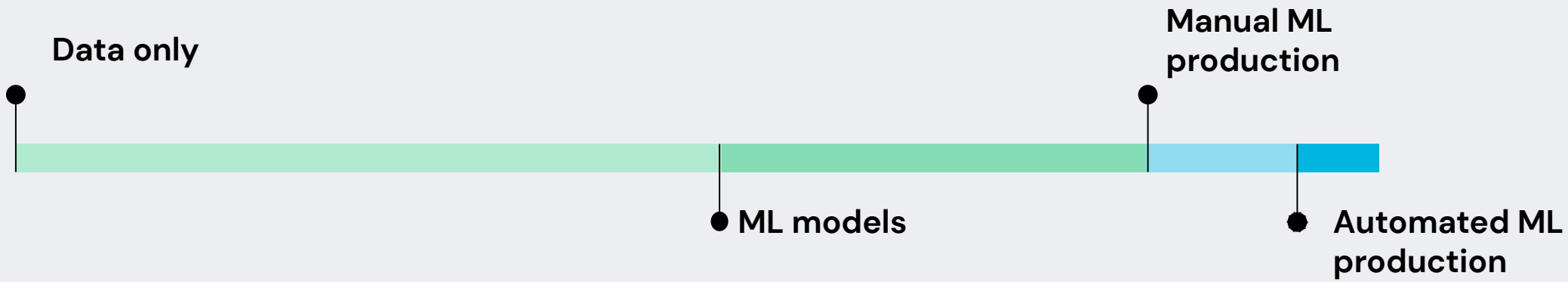
Jin Zhang
Staff Software Engineer
Databricks

ORGANIZED BY databricks

Problems

ML maturity levels

Almost every company wants to be a Data+AI company, but the majority are new to machine learning (ML).



3 common problems with scaling ML

Getting started

Mundane steps with lots of boilerplate

Iteration speed

Slow, redundant iterative development

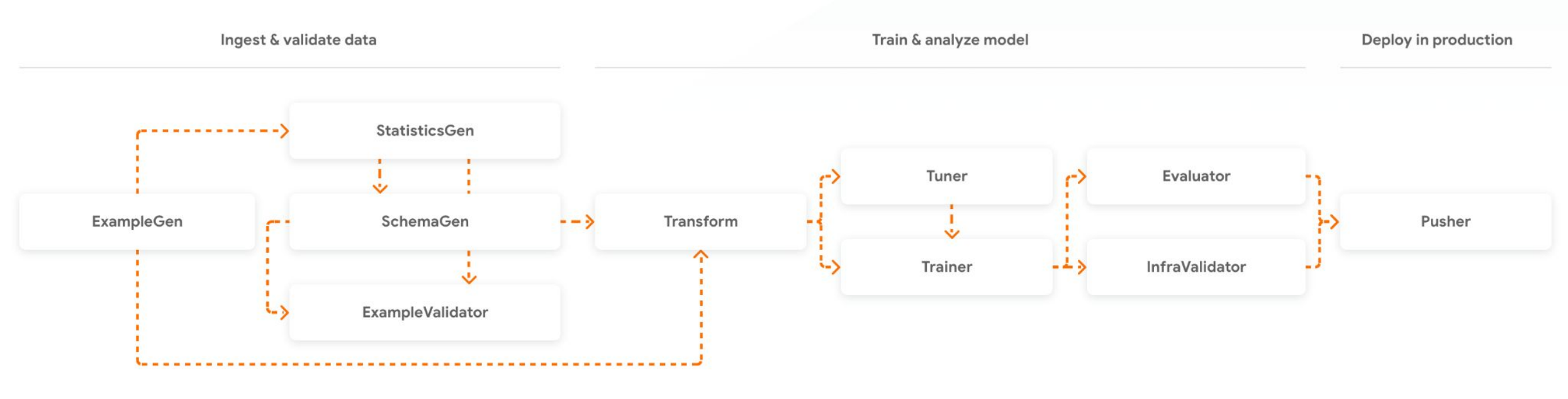
Productionization

Handoff to production is very manual and bottlenecked

Explore Solutions

TensorFlow Extended (TFX)

Google-production-scale ML pipelines



https://www.tensorflow.org/tfx

TensorFlow Extended (TFX)

What fits:

- ML pipeline abstraction
- predefined steps
- embedded best practices

What doesn't fit:

- Designed for ML engineers to solve complex and large-scale ML problems, hard for data scientists to get started.

```
trainer = tfx.components.Trainer(
    module_file=os.path.abspath(_taxi_trainer_module_file),
    examples=transform.outputs['transformed_examples'],
    transform_graph=transform.outputs['transform_graph'],
    schema=schema_gen.outputs['schema'],
    train_args=tfx.proto.TrainArgs(num_steps=10000),
    eval_args=tfx.proto.EvalArgs(num_steps=5000))
context.run(trainer, enable_cache=True)
```

Apache Maven

"Why is no one using make for Java?"

Project Object Model (POM)

```
<project>
  <modelVersion>4.0.0</modelVersion>

  <parent>
    <groupId>com.mycompany.app</groupId>
    <artifactId>my-app</artifactId>
    <version>1</version>
  </parent>

  <groupId>com.mycompany.app</groupId>
  <artifactId>my-module</artifactId>
  <version>1</version>
</project>
```

Build Lifecycle

- validate
- compile
- test
- package
- verify
- install
- deploy

Standard Directory Layout

- pom.xml
- src/main/java
- src/main/resources
- src/main/filters
- src/main/webapp
- src/test/java
- src/test/resources
- src/test/filters
- src/it
- src/assembly
- src/site
- LICENSE.txt
- NOTICE.txt
- README.txt

Apache Maven

What fits:

- Easy to get started with predefined build lifecycle and standard directory layout.
- Standardized on POM for collaboration and sharing.
- Build cache for incremental build.

What doesn't fit:

- One "build" lifecycle might not fit all ML problem types.
- Not designed to solve ML problems, where data is essential.

Notebook vs. IDE

Do we have to choose one?

Notebooks are good for:

- analyzing data
- interactive execution
- prototyping
- …

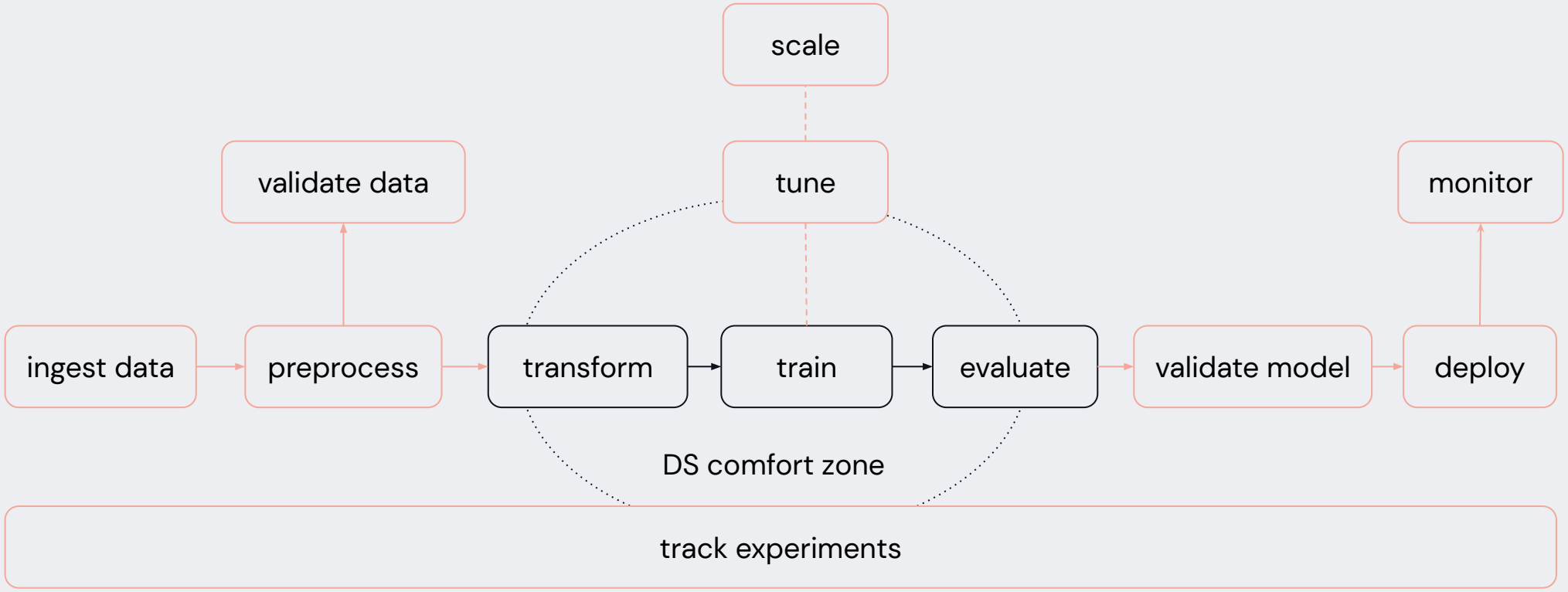
IDEs are good for:

- writing code
- batch execution
- productionization
- …

Production ML needs both data and code!

MLflow Pipelines

Start quickly with pipeline templates



Start quickly with pipeline templates

Clustering the 80% of ML problems

Built-in pipeline templates:

- regression
- classification
- …

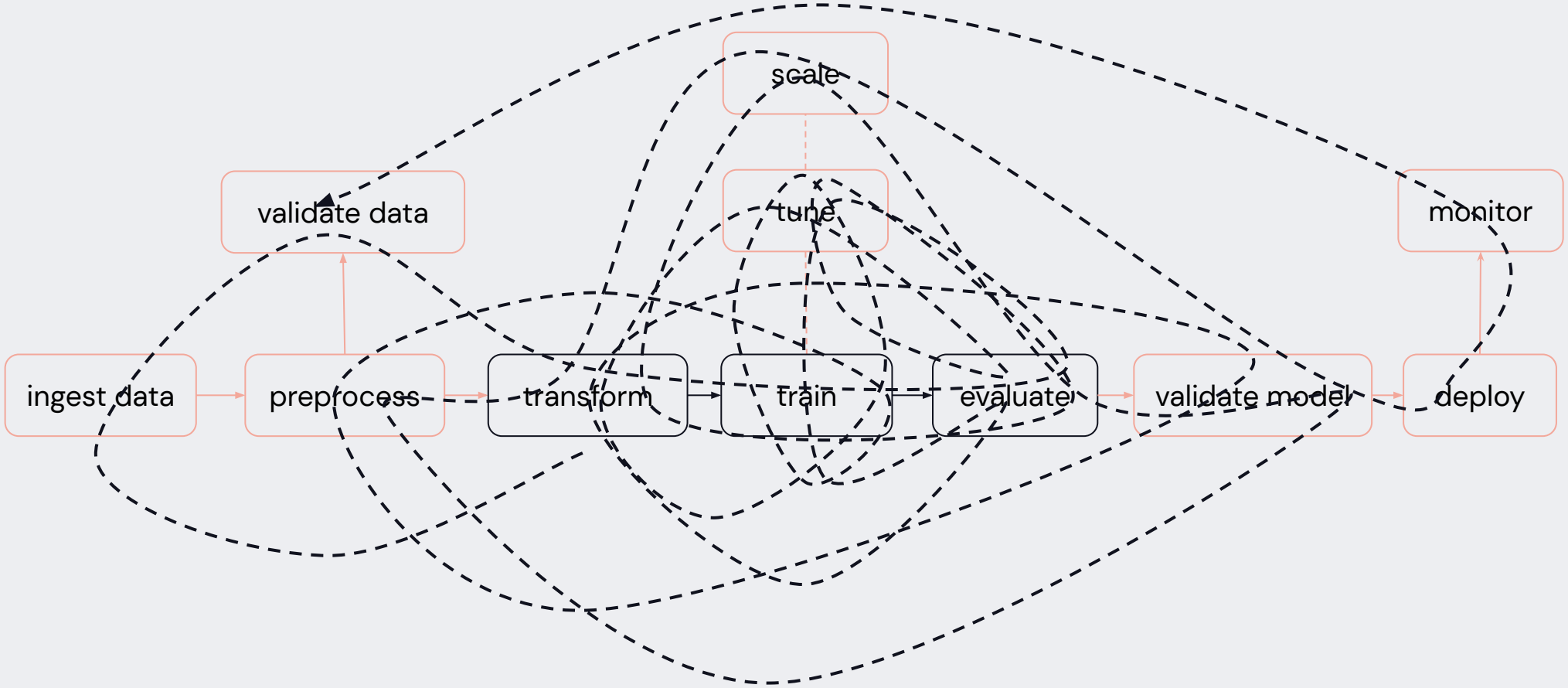
Simple to get started:

- standard directory layout
- mostly config-driven
- code within DS' comfort zone

pipeline.yaml

```
template: regression/v1

data:
  type: parquet
  path: datasets/taxi.parquet

target_col: fare_amount

metrics:
  primary: rmse

steps:
  transform:
    …
```

Iterate fast with the pipeline engine



Accelerate with opinionated dev workflows

Embedded best practices

- deterministic splits
- data profiles
- transformed feature names
- feature importance
- automatic MLflow tracking
- ...

Notebook + IDE

- Notebook to trigger pipeline execution.
- Notebook to view rich format output.
- Notebook to analyze data.

- IDE to develop modularized code.

Productionize without refactoring

- Notebook is only used for triggering execution, not on the critical path. Whenever one finds a good model, it is ready to ship.
- Default test suites to prevent unexpected changes.
- Standard command-line interface to integrate with CI/CD.

mlflow™ Pipelines

Pipeline templates to get started quickly

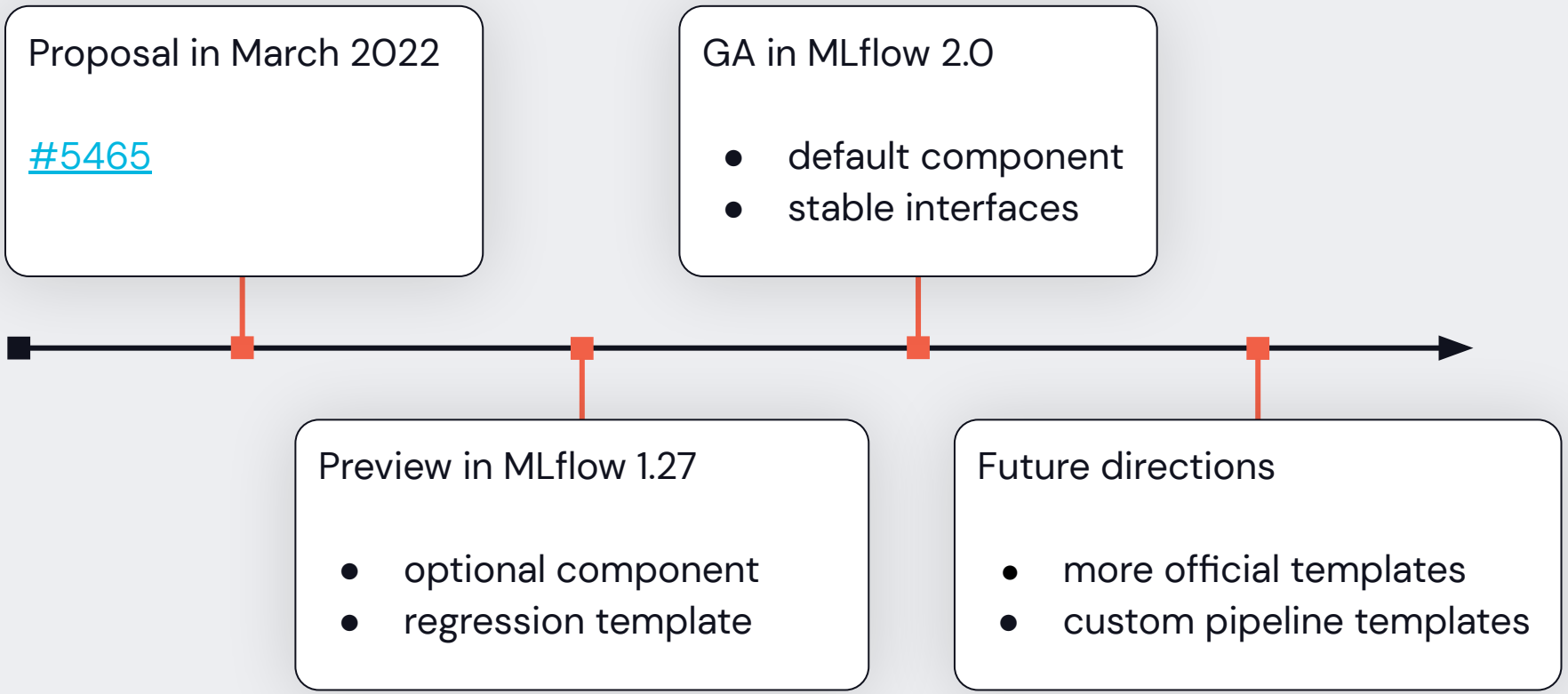
Pipeline engine for accelerated development

Opinionated structure to automate handoffs

Feature Spotlights

Timeline

Your subtitle here



MLflow Pipelines

Getting started

- Download MLP template (scan )
  - https://github.com/mlflow/mlp-regression-template
- Discuss improvement topics
  - https://github.com/mlflow/mlflow/discussions
- Report issues
  - https://github.com/mlflow/mlflow/issues

MLflow Pipelines