ML Operations Case Study

Kritharakis Emmanouil

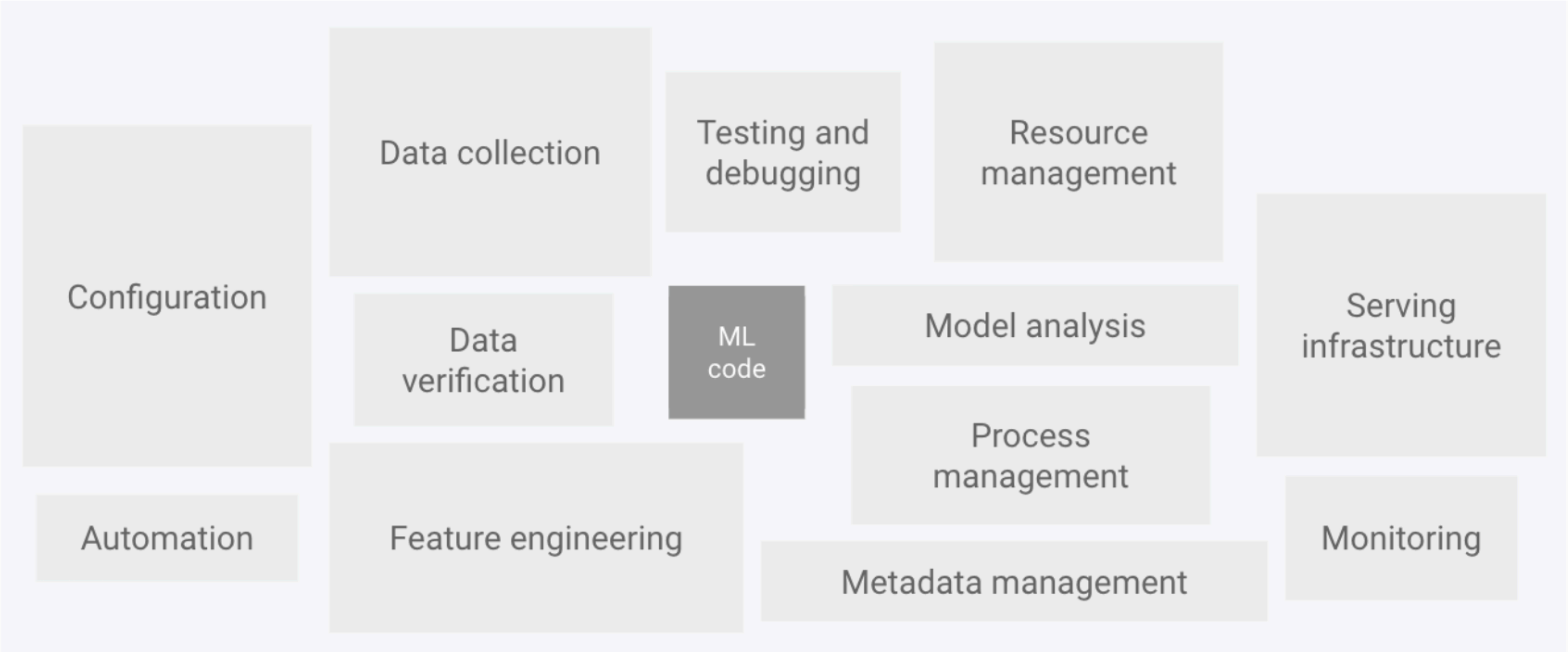
CIRCU
LI-ION

Hypothesis

Congratulations!
You build a Computer Vision ML model.

↓

Deploy to production.
How?



Configuration

Data collection

Testing and debugging

Resource management

Data verification

ML code

Model analysis

Serving infrastructure

Process management

Automation

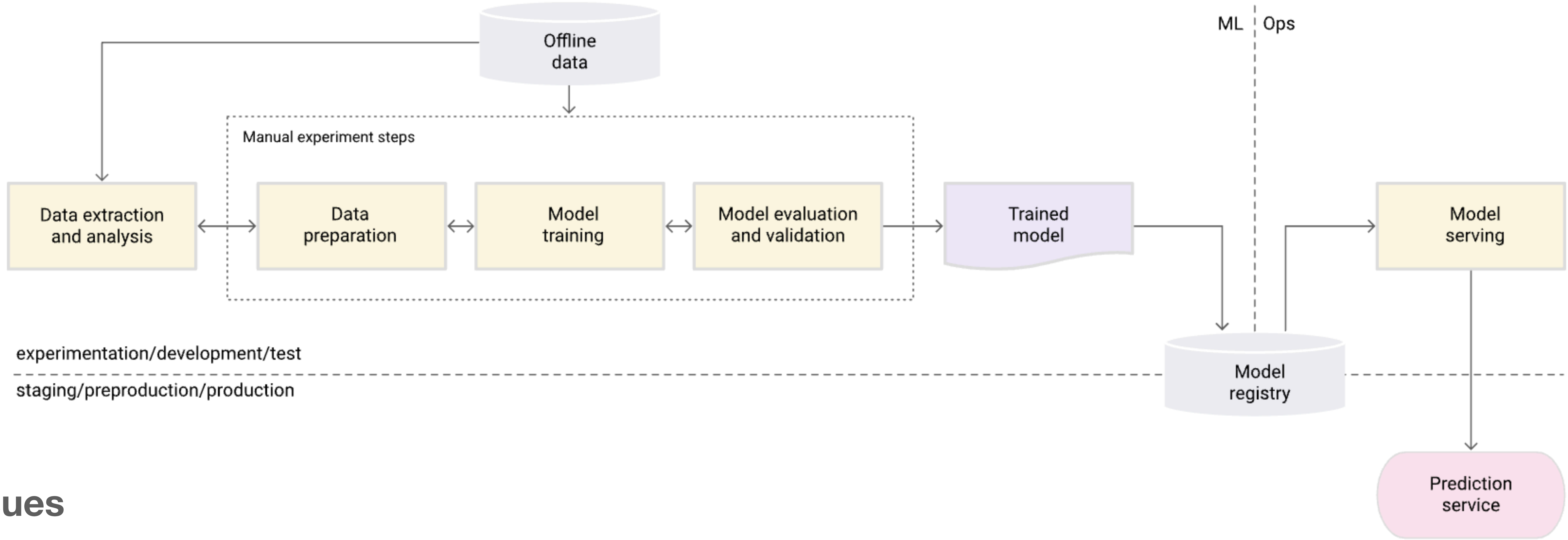
Feature engineering

Metadata management

Monitoring

Source: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning#mlops_level_2_cicd_pipeline_automation

First Approach MLOps Architecture

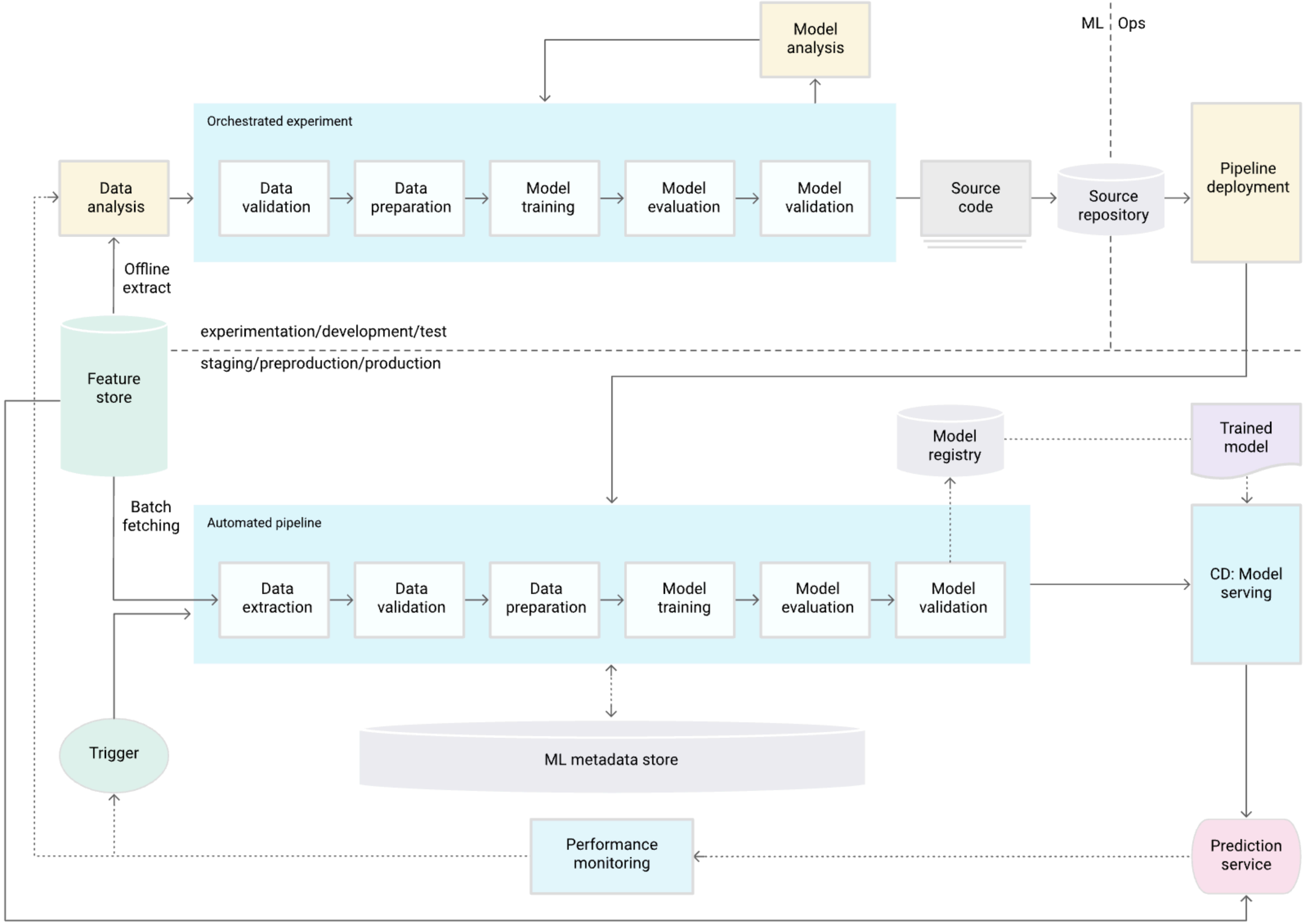


Rising Issues

- Manual, Script-driven, & Interactive process
- Training Serving Skew
- No Continuous Training Option
- No Monitoring

Source: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning#mlops_level_2_cicd_pipeline_automation

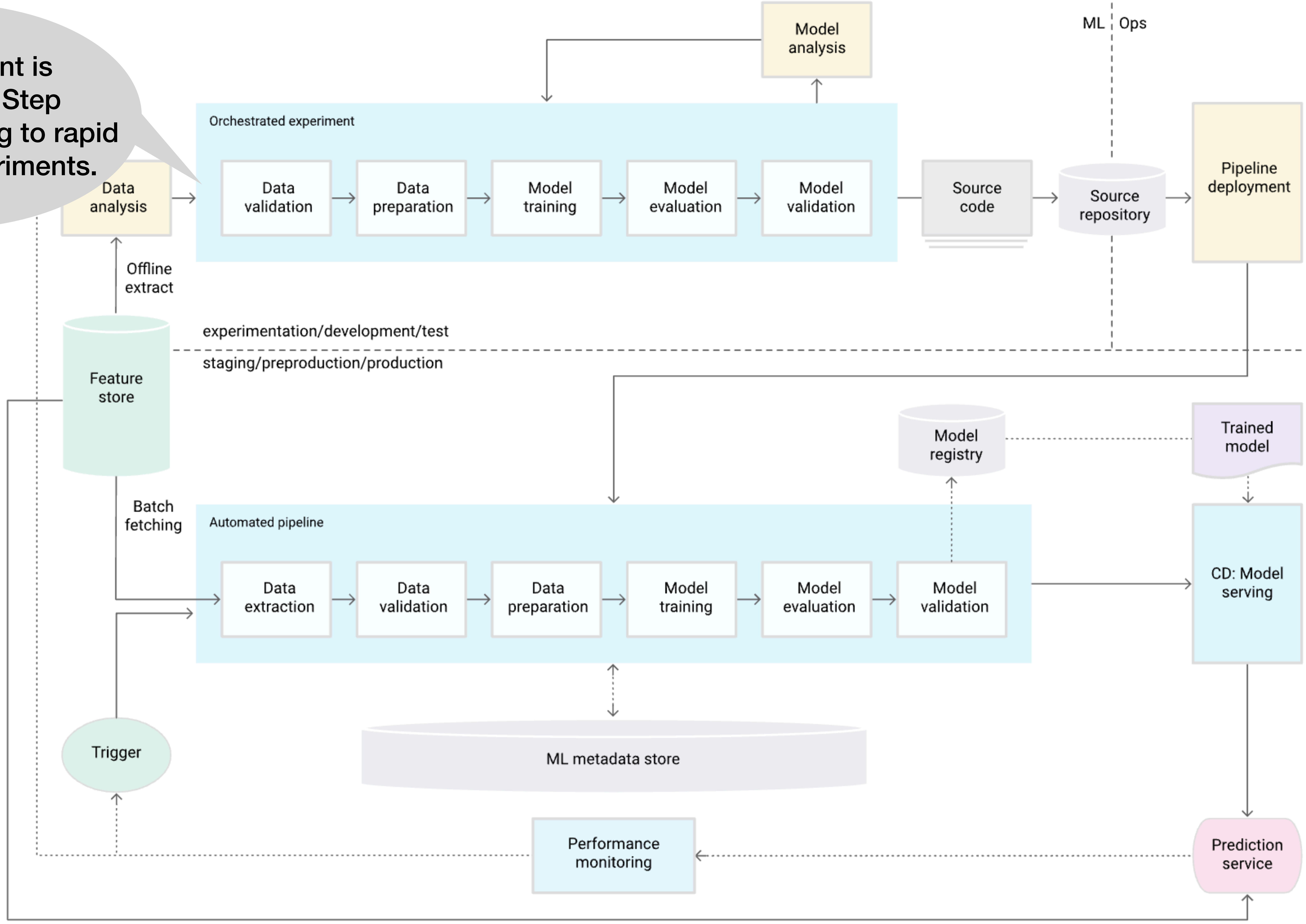
Improved Approach MLOps Architecture



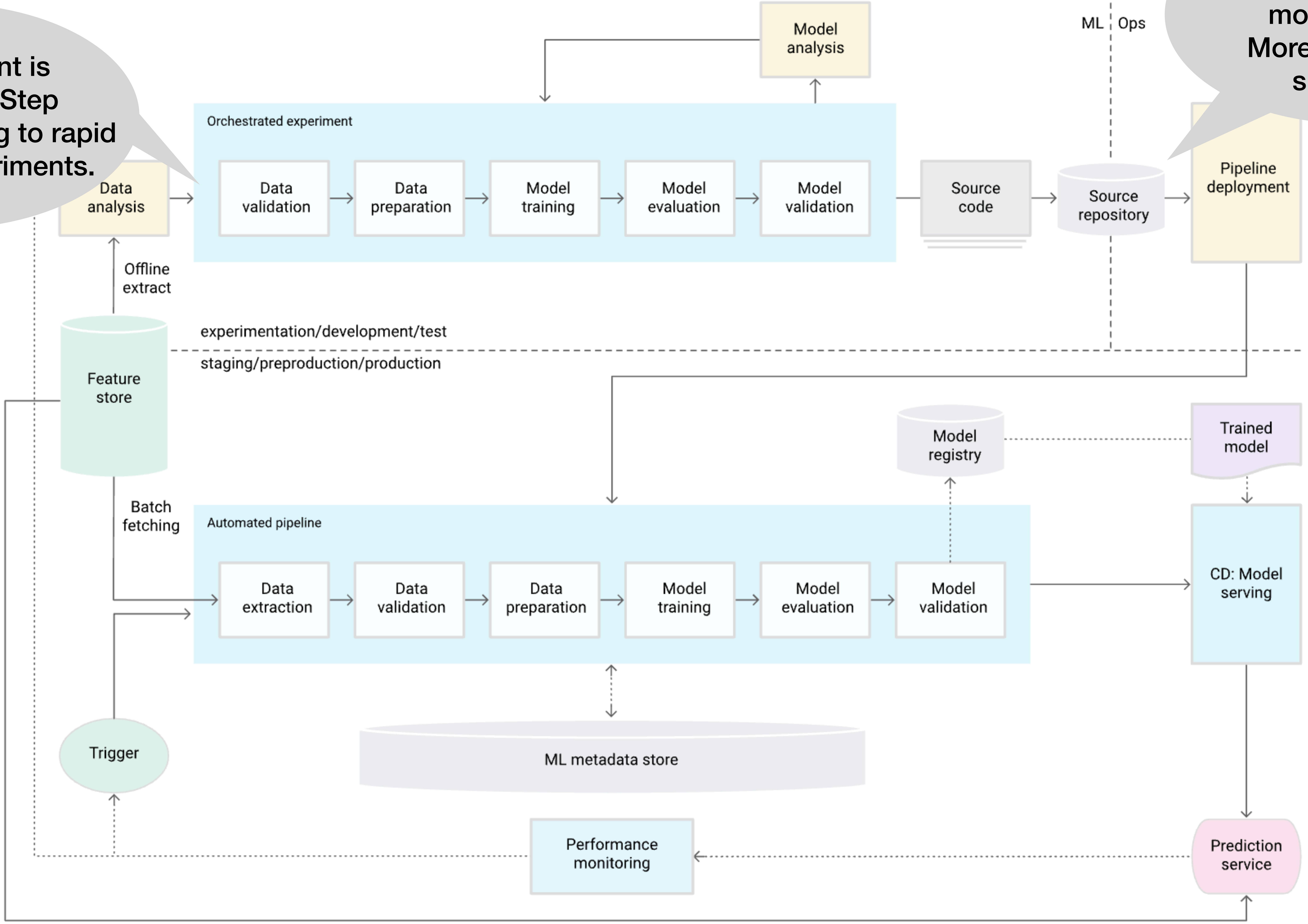
Source: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning#mlops_level_2_cicd_pipeline_automation

Improved Approach MLOps Architecture

ML experiment is orchestrated. Step automated, leading to rapid iteration of experiments.



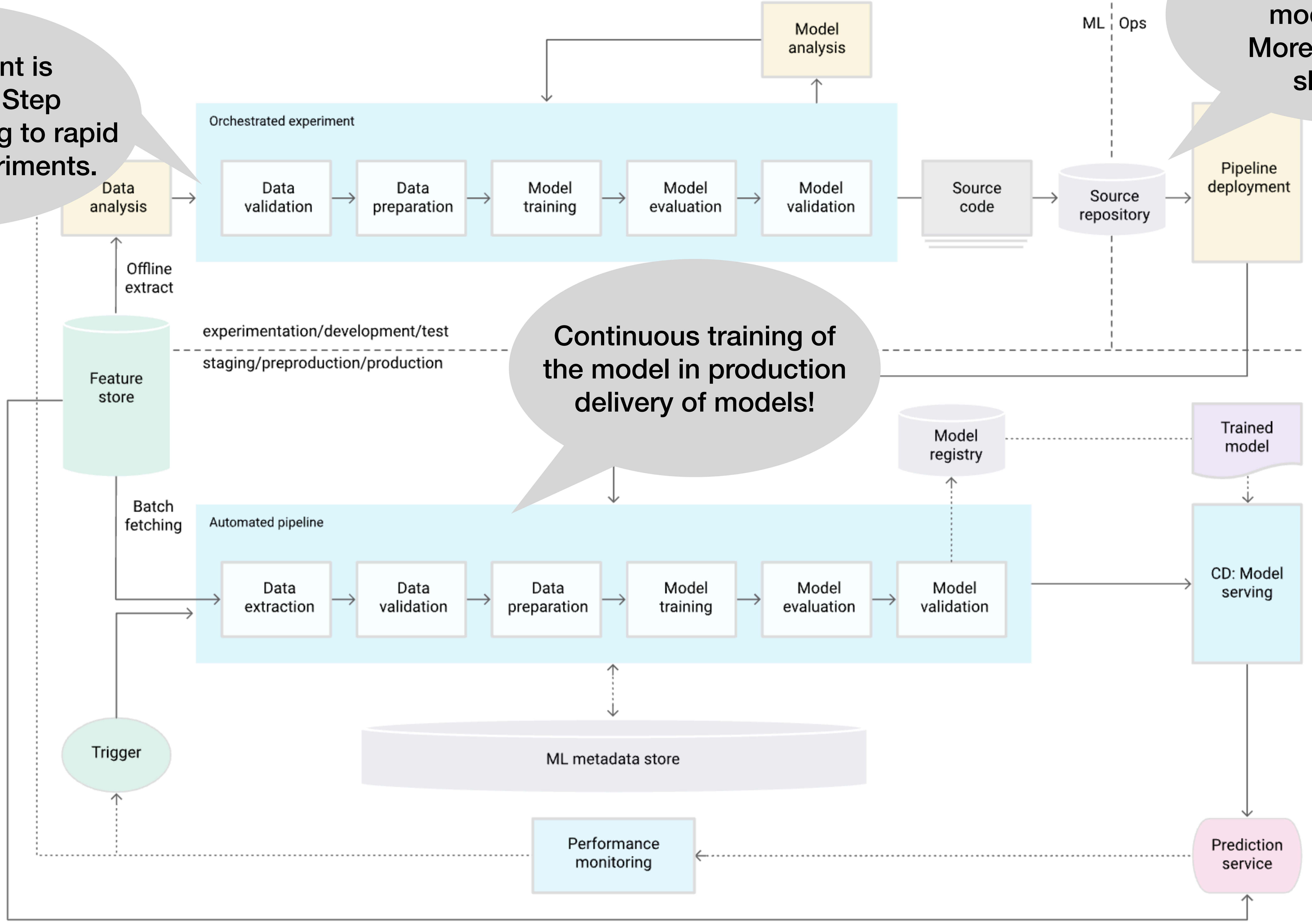
Improved Approach MLOps Architecture



ML experiment is orchestrated. Step automated, leading to rapid iteration of experiments.

Source code for components become modularized. More reusable & sharable!

Improved Approach MLOps Architecture

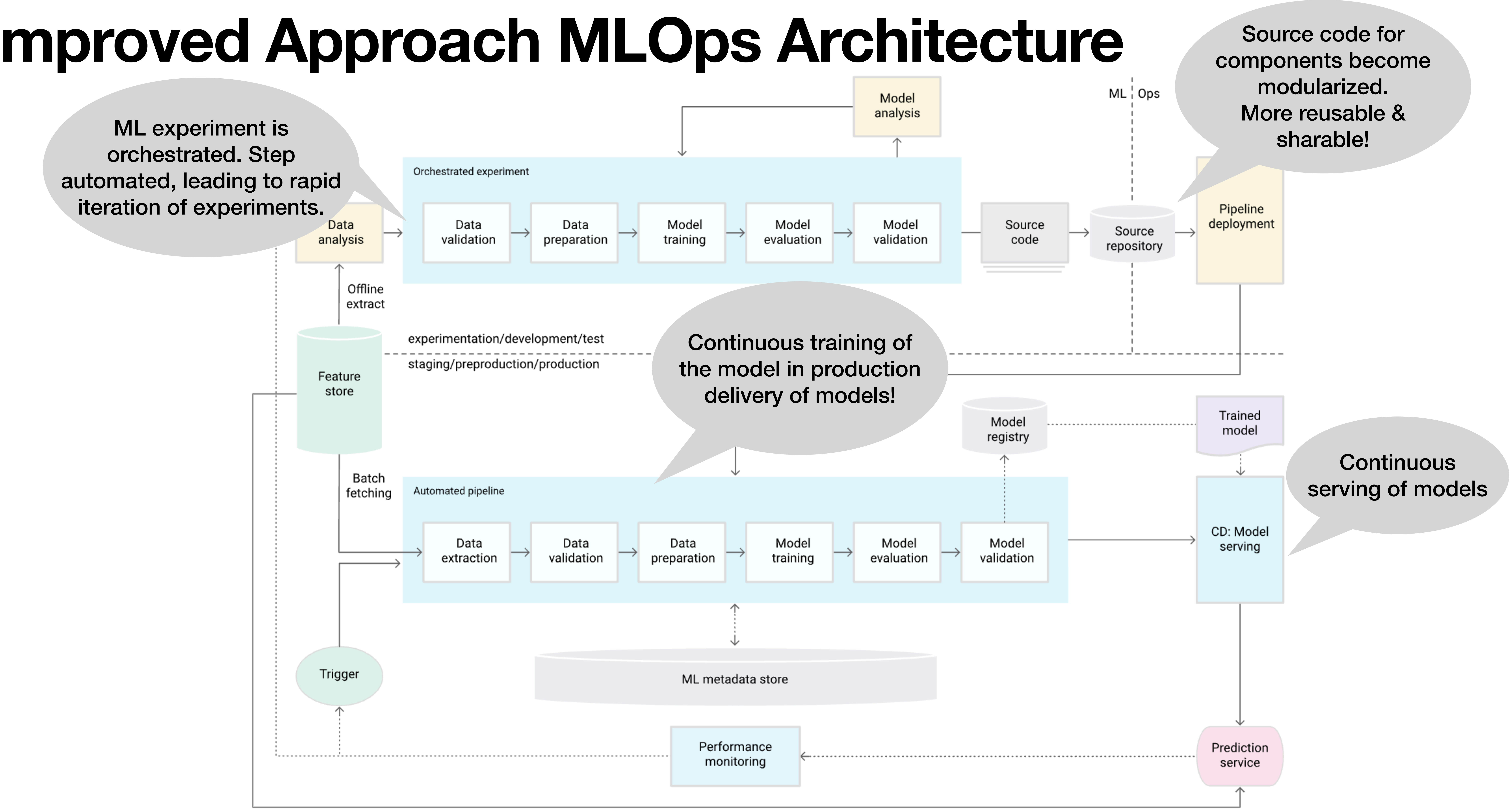


ML experiment is orchestrated. Step automated, leading to rapid iteration of experiments.

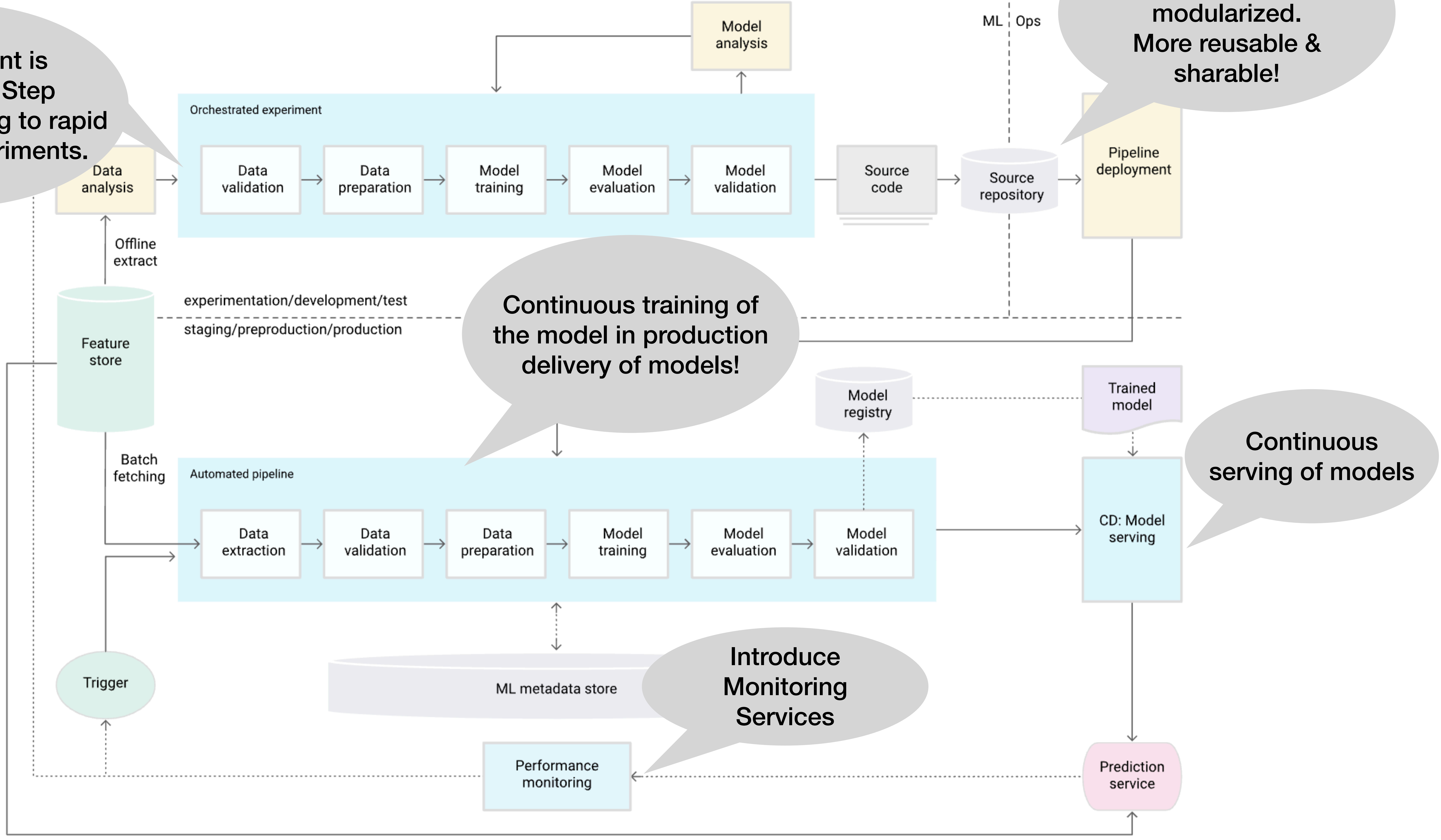
Source code for components become modularized. More reusable & sharable!

Continuous training of the model in production delivery of models!

Improved Approach MLOps Architecture



Improved Approach MLOps Architecture



ML experiment is orchestrated. Step automated, leading to rapid iteration of experiments.

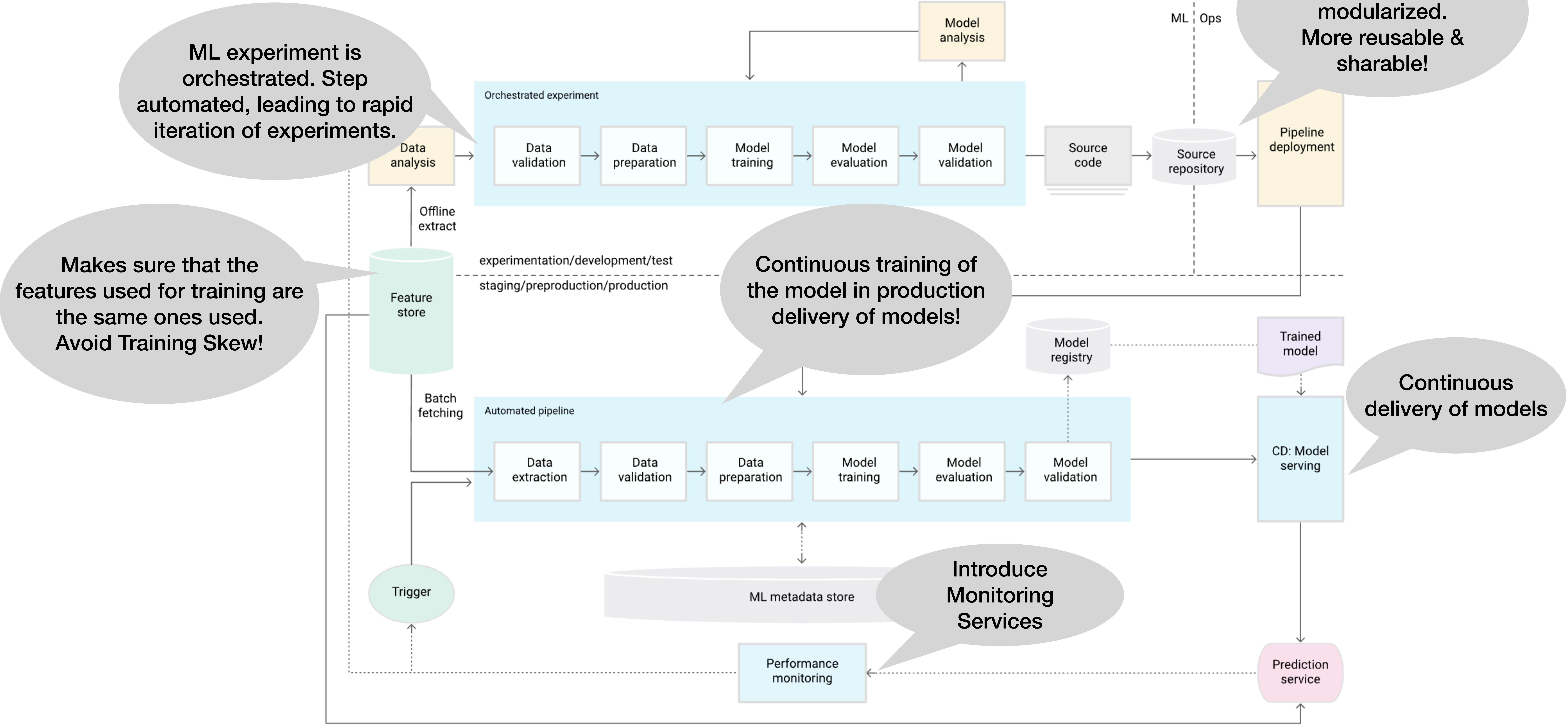
Source code for components become modularized. More reusable & sharable!

Continuous training of the model in production delivery of models!

Continuous serving of models

Introduce Monitoring Services

Improved Approach MLOps Architecture



ML experiment is orchestrated. Step automated, leading to rapid iteration of experiments.

Source code for components become modularized. More reusable & sharable!

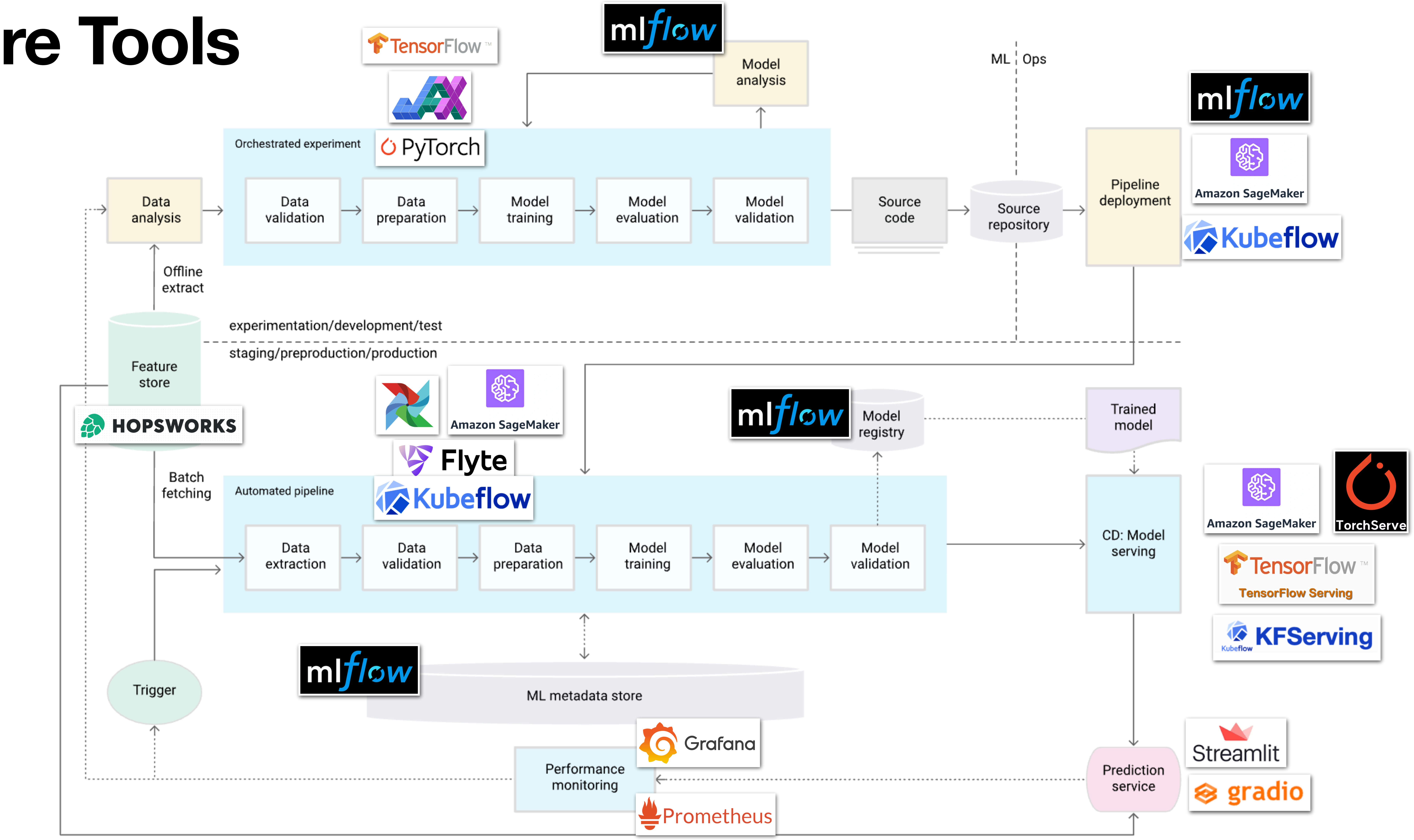
Makes sure that the features used for training are the same ones used. Avoid Training Skew!

Continuous training of the model in production delivery of models!

Continuous delivery of models

Introduce Monitoring Services

Software Tools



TensorFlow

mlflow

Model analysis

ML

Ops

PyTorch

Orchestrated experiment

Data analysis

Data validation

Data preparation

Model training

Model evaluation

Model validation

Source code

Source repository

Pipeline deployment

mlflow

Amazon SageMaker

Kubeflow

Offline extract

experimentation/development/test

staging/preproduction/production

Feature store

HOPSWORKS

Amazon SageMaker

Flyte

Kubeflow

mlflow

Model registry

Trained model

Batch fetching

Automated pipeline

Data extraction

Data validation

Data preparation

Model training

Model evaluation

Model validation

CD: Model serving

Amazon SageMaker

TorchServe

TensorFlow

TensorFlow Serving

Kubeflow

KFServing

Trigger

mlflow

ML metadata store

Grafana

Performance monitoring

Prometheus

Prediction service

Streamlit

gradio

Software Tools In Depth

Development - Deployment - Continuous Training (CT)

mlflow

- Experiment Tracking
- Packages data science source for reproducibility
- Deploys & manages ML models
- Model Registry

Flyte

- Versioned & Reproducible pipelines
- Deployment at scale
- Ergonomic SDKs in Python, Java & Scala

Kubeflow

- End-to-End MLOps Platform
- Deployment on Kubernetes portable & scalable.
- Data preparation
- Model training & optimization
- Prediction serving (KFServing)

Amazon SageMaker

- End-to-End MLOps Platform
- A collaborative environment for data science teams
- Automate ML training workflows
- CI/CD for automatic integration and deployment
- Continuous monitoring & Retaining models to maintain quality

- Pipeline orchestration tool
- Helpful for build, manage & deploy ML models
- Includes Data & Model Validation modules
- Scale horizontally as well as vertically.

Monitoring

Prometheus monitors metrics collection and alerting toolkit.

Grafana provides dashboard to query, visualize, & alert on monitoring metrics.

Serving

TensorFlow™ TensorFlow Serving can now experiment, train, deploy, & maintain ML model. However, it only works with Tensorflow models.

TorchServe is a performant, flexible and easy to use tool for serving PyTorch models in production.

Prediction Services

Streamlit provides a simple API that enables users to create intuitive and interactive ML applications.

gradio creates & share ML apps. In a few lines of code, create a Interface. Available as webpage or embedded into Python notebook.

Feature Store

HOPSWORKS is a powerful centralized storage for ML features. Allows the automation and management of feature engineering and serving at scale for stream and batch data.

CIRCU LI-ION

Thank you!

Any Questions?