
AWS
re:Invent
NOV. 28 – DEC. 2, 2022 | LAS VEGAS, NV

STP303-R

Deploying a complete machine learning fraud detection solution

Manish Chugh (he/him)

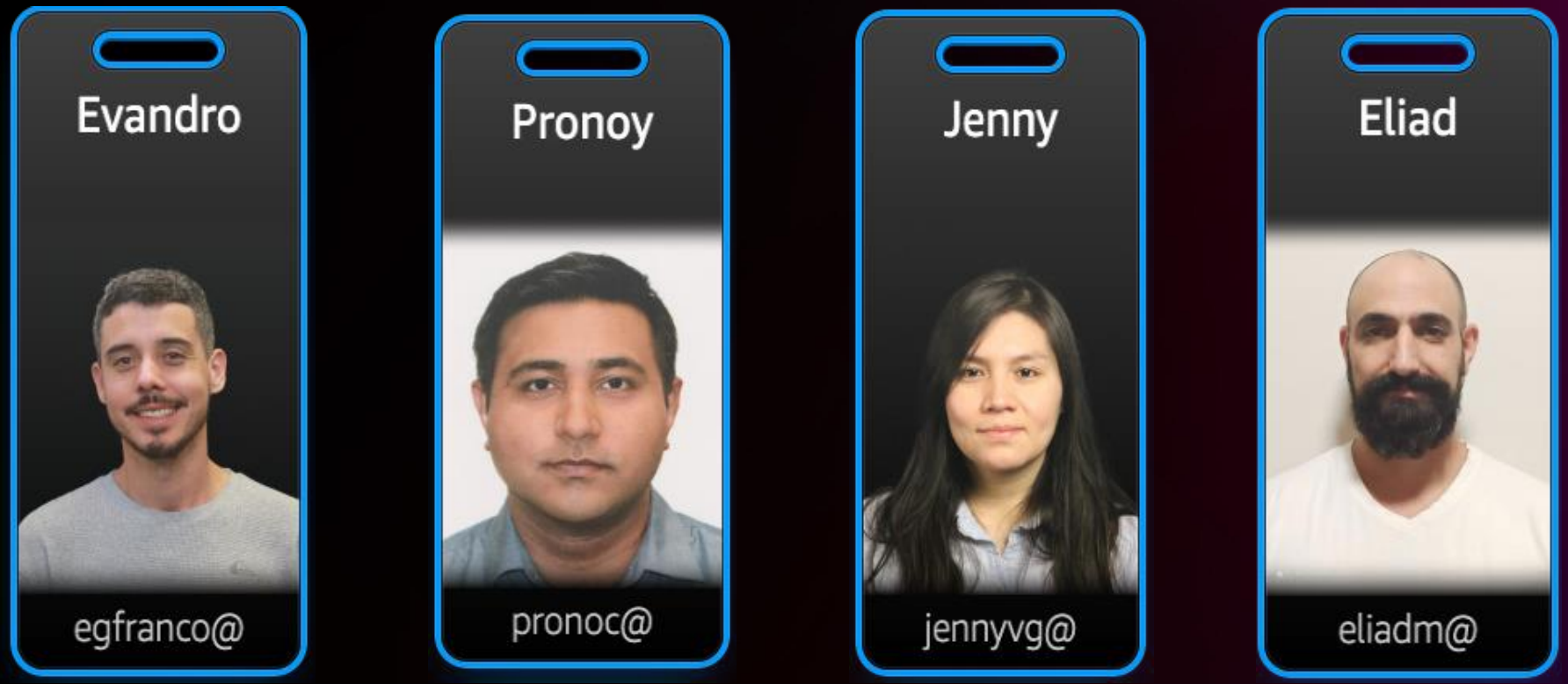
Principal Solutions Architect
Amazon Web Services

Pallavi Nargund (she/her)

Senior Solutions Architect
Amazon Web Services

Speakers

Workshop Core Team



Agenda

Auto insurance fraud

Machine learning with Amazon SageMaker

Hands-on labs

Problem statement

31%[1]

$40B
$400 - $700[2]

72%[3]

[1] Personal auto research: Application integrity down, fraud up - https://vrsk.co/3UBQRhm
[2] Federal Bureau of Investigation (FBI) Publication - https://bit.ly/3hyHj80
[3] Coalition Against Insurance Fraud Stats - https://bit.ly/3g41LNU

Challenges

Traditional business policies, procedures and rules

Lack of meaningful insights in data collected

Slow and time consuming, impacts customer experience

Use of technology to combat fraud

Predictive modeling and reduce false positives

Advance analytics and Machine learning to rescue

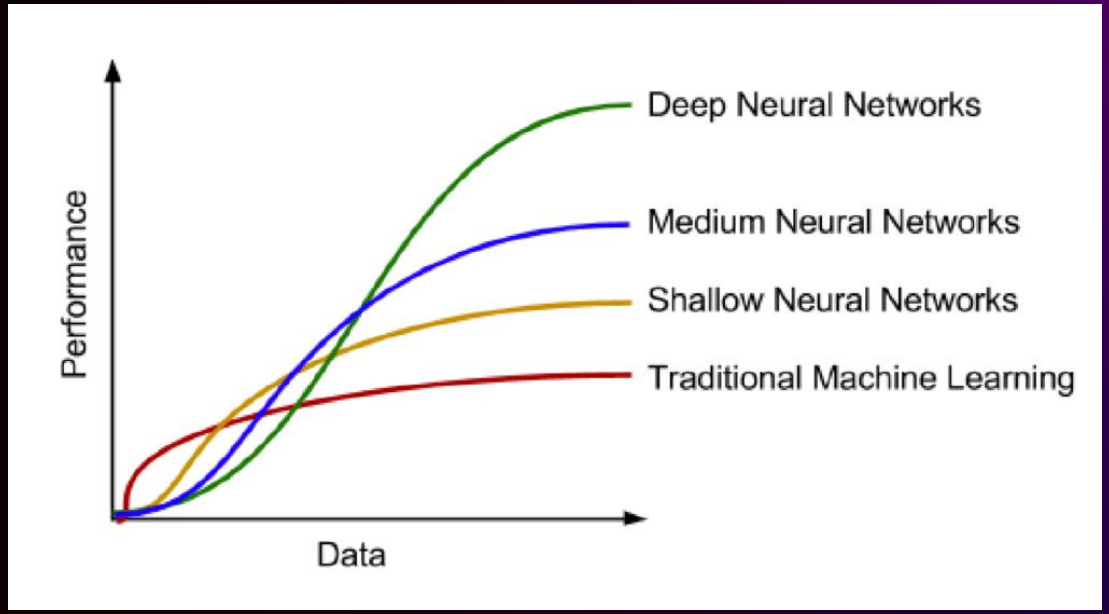
Cloud computing accelerate innovation and reduces cost

Machine learning challenges

- Extremely unbalanced data (aka “one-class” dataset)
- Often unlabeled data
- Requires low-latency inference; models can’t be too complex

Machine learning solution option

- Labeled data/supervised learning
  - Build a Classifier
- Unlabeled data/unsupervised learning
  - Build an anomaly detector
    - Isolation Forest, Random Cut Forest
    - Build deep learning autoencoders for semi-supervised learning



The AWS AI/ML stack

Broadest and most complete set of machine learning capabilities

AI SERVICES

SPECIALIZED

Business processes
- Amazon Personalize
- Amazon Forecast
- Amazon Fraud Detector
- Amazon Lookout for Metrics

Search
- Amazon Kendra

Code + DevOps
- Amazon CodeGuru
- Amazon DevOps Guru

Industrial
- Amazon Monitron
- Amazon Lookout for Equipment
- Amazon Lookout for Vision

Healthcare
- Amazon HealthLake
- Amazon Comprehend Medical
- Amazon Transcribe Medical

CORE

Text & Documents
- Amazon Translate
- Amazon Comprehend
- Amazon Textract

Chatbots
- Amazon Lex

Speech
- Amazon Polly
- Amazon Transcribe
- Amazon Transcribe Call Analytics

Vision
- Amazon Rekognition
- AWS Panorama

ML SERVICES

SAGEMAKER CANVAS

No-code ML for business analysts

SAGEMAKER STUDIO LAB

Learn ML

AMAZON SAGEMAKER

Label data

Prepare data | Store features | Detect bias | Build with notebooks | Train models | Tune parameters | Deploy in production | Explain predictions | Manage & monitor

CI/CD

Manage edge devices

ML FRAMEWORKS & INFRASTRUCTURE

PyTorch, Apache MXNet, TensorFlow

Amazon EC2

CPUs

GPUs

AWS Inferentia

AWS Trainium

Habana Gaudi

FPGA

Elastic inference

Amazon SageMaker

PREPARE DATA AND BUILD, TRAIN, AND DEPLOY ML MODEL FOR ANY USE CASE

PREPARE

Ground Truth
Create high quality datasets for ML

Data Wrangler
Aggregate and prepare data for ML

Processing
Built-in Python, BYO R/Spark

Feature Store
Store, catalog, search, and reuse features

Clarify
Detect bias and understand model predictions

BUILD

Studio Notebooks & Notebook Instances
Fully managed Jupyter notebooks with elastic compute

Studio Lab
Free ML development environment

Built-in Algorithms
Integrated tabular, NLP, and vision algorithms

JumpStart
UI based discovery, training, and deployment of models, solutions, and examples

Autopilot
Automatically create ML models with full visibility

Bring Your Own
Bring your own container and algorithms

Local Mode
Test and prototype on your local machine

TRAIN & TUNE

Fully Managed Training
Broad hardware options, easy to setup and scale

Distributed Training Libraries
High performance training for large datasets and models

Training Compiler
Faster deep learning model training

Automatic Model Tuning
Hyperparameter optimization

Managed Spot Training
Reduce training cost by up to 90%

Debugger and Profiler
Debug and profile training runs

Experiments
Track, visualize, and share model artifacts across teams

Customization Support
Integrate with popular open-source frameworks and libraries

DEPLOY & MANAGE

Fully Managed Deployment
Ultra low latency, high throughput inference

Real-Time Inference
For steady traffic patterns

Serverless Inference
For intermittent traffic patterns

Asynchronous Inference
For large payloads or long processing times

Batch Transform
For offline inference on batches of large datasets

Multi-Model Endpoints
Reduce cost by hosting multiple models per instance

Multi-Container Endpoints
Reduce cost by hosting multiple containers per instance

Inference Recommender
Automatically select compute instance and configuration

Model Monitor
Maintain accuracy of deployed models

Kubernetes & Kubeflow Integration
Simplify Kubernetes-based ML

Edge Manager
Manage and monitor models on edge devices

Studio | RStudio
Integrated development environment (IDE) for ML

MLOps: Pipelines | Projects | Model Registry
Workflow automation, CI/CD for ML, central model catalog

Canvas
Generate accurate machine learning predictions—no code required

Workshop use case – Labeled data set

Binary classification problem

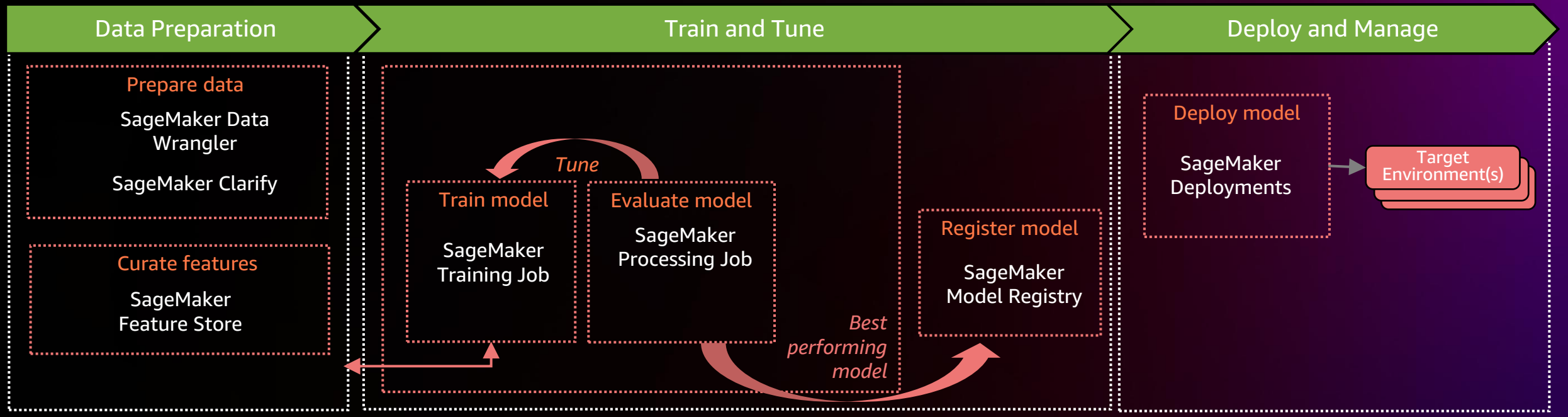
Improve predictive accuracy using boosting

Gradient boosting

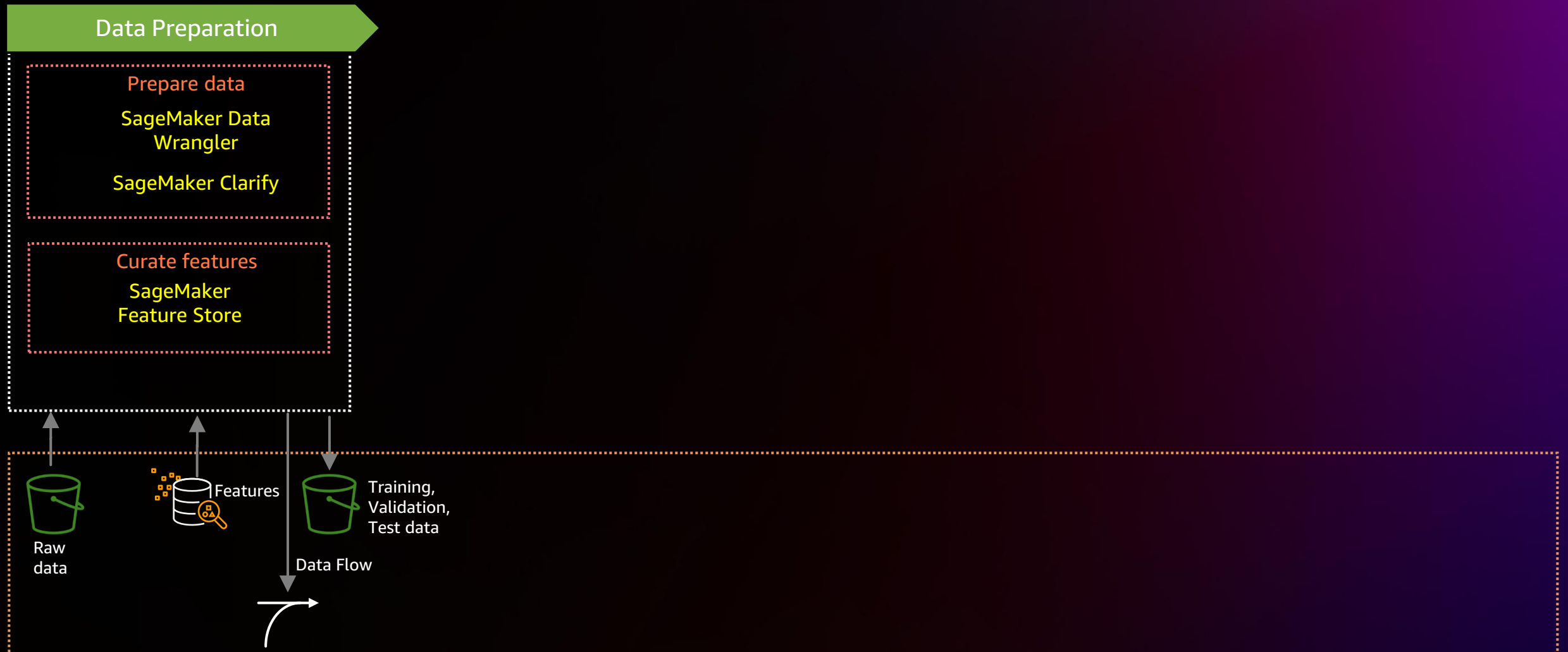
Extreme gradient boosting

Building, training, and deploying models

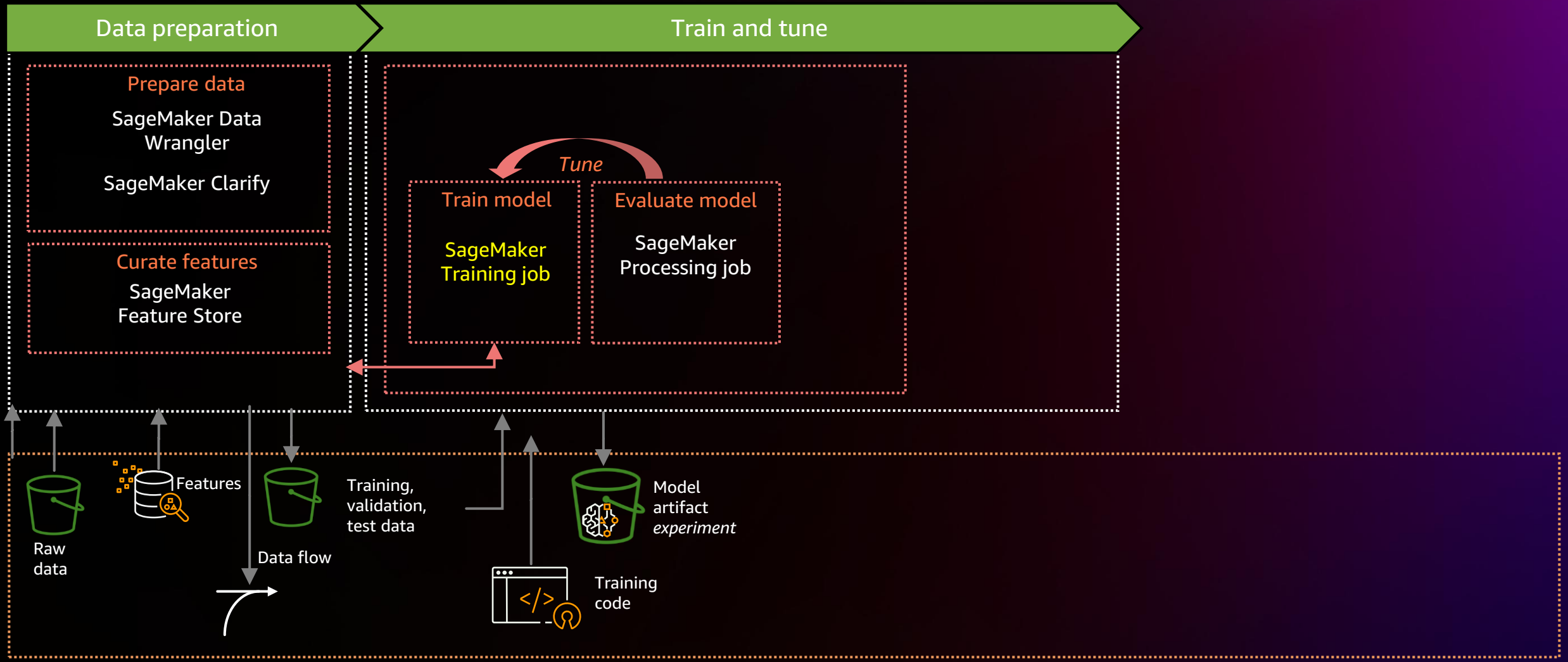
Experimentation to Production



Data preparation



Training and tuning



Model options

Training code

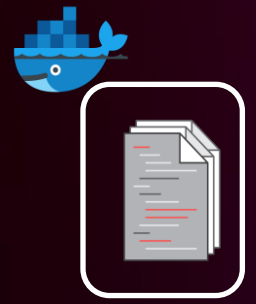
Built-in algorithms (17)
No ML coding required

- XGBoost - Gradient boosted trees
- Matrix factorization
- Regression
- Principal component analysis
- K-means clustering
- And more!

Bring your own script
Amazon SageMaker builds the container
Open-source containers

GLUON, mxnet, Keras, TensorFlow, HOROVOD, dmlc XGBoost, PyTorch, Apache Spark, Chainer, scikit learn

Bring your own container
Full control, you build the container
R, C++, etc.



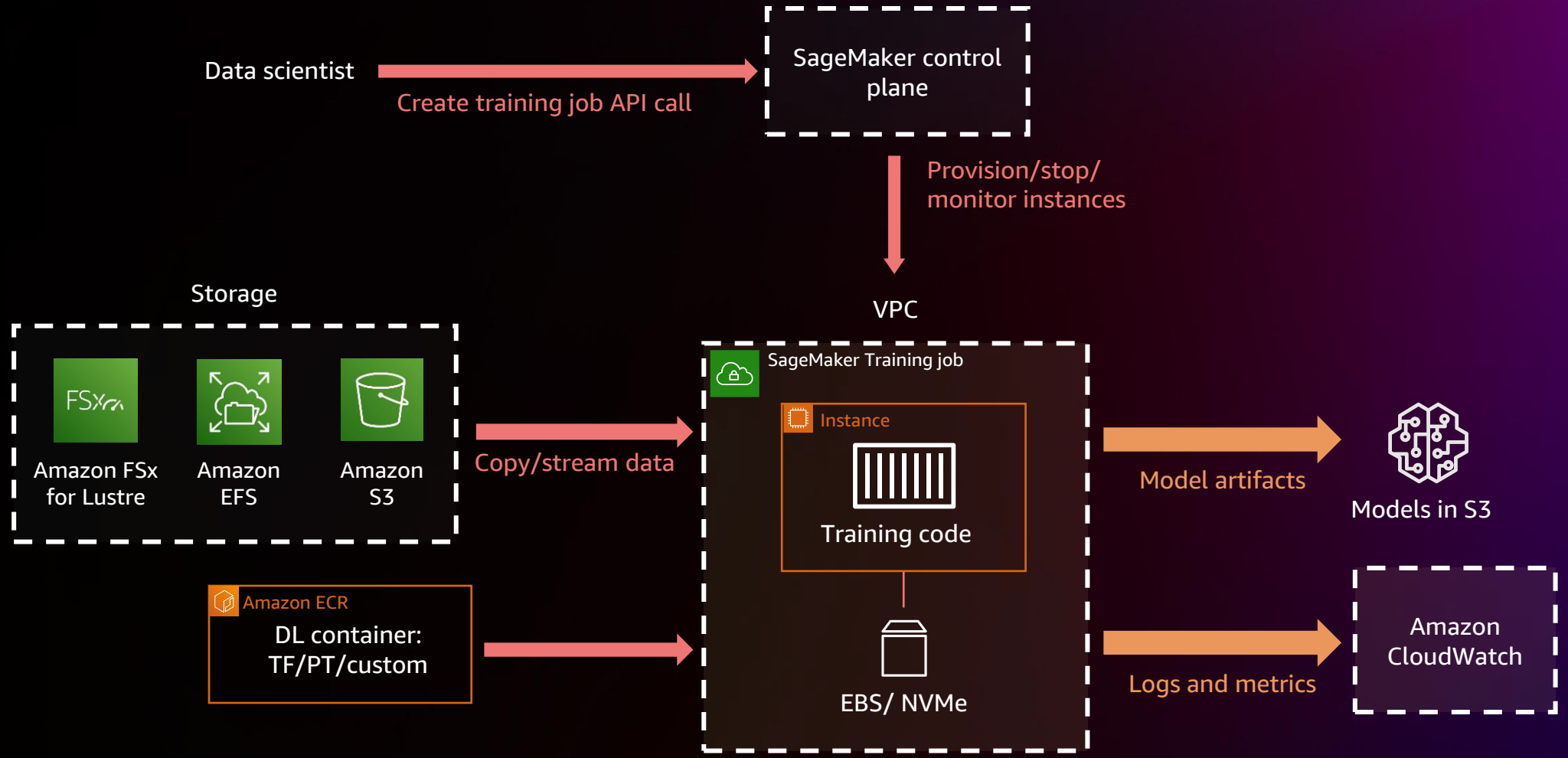
AWS Marketplace for machine learning

Fully managed, distributed, auto scaled, secured

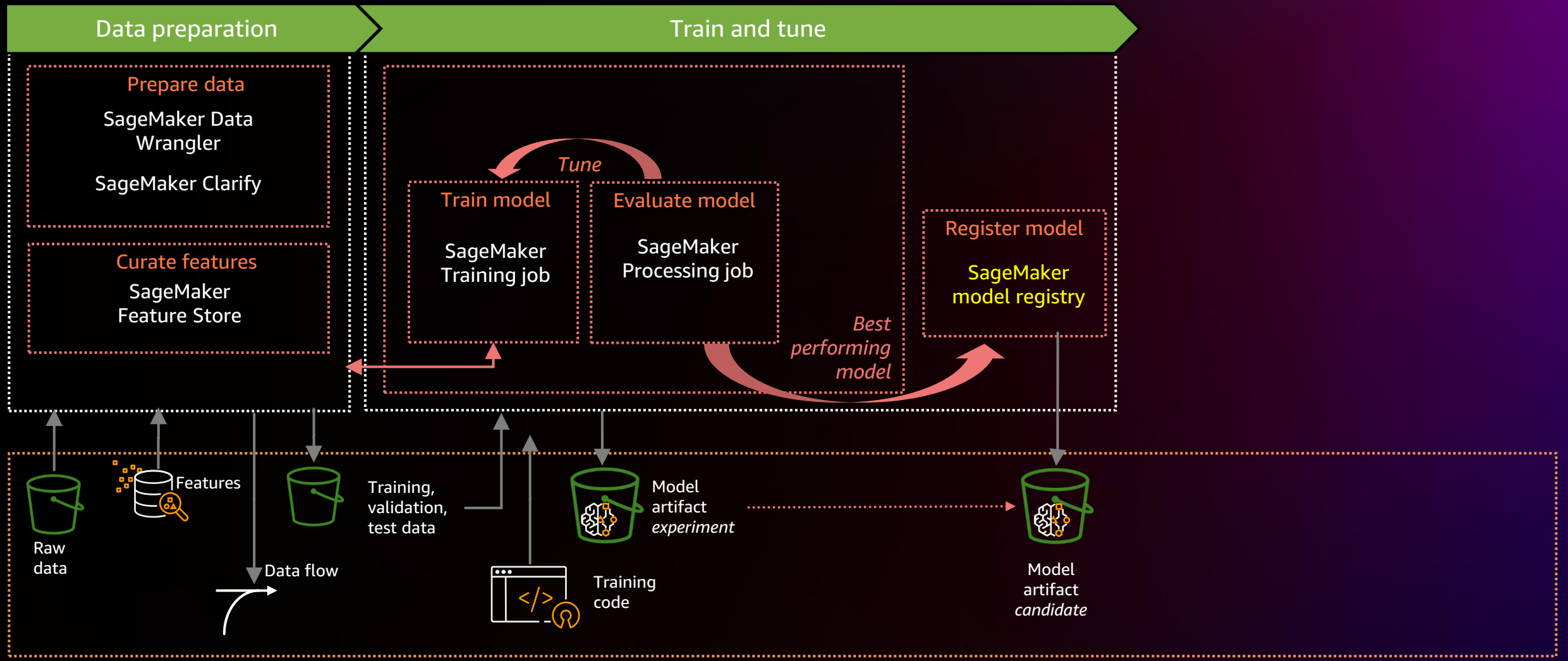
Training on SageMaker

```
# Input data from s3
inputs = {"train": s3_inputs_train, "test": s3_inputs_test}

metric_definitions = [
    {"Name": "loss", "Regex": "loss: ([0-9\\.]+)"},
    {"Name": "accuracy", "Regex": "accuracy: ([0-9\\.]+)"},
    {"Name": "val_loss", "Regex": "val_loss: ([0-9\\.]+)"},
    {"Name": "val_accuracy", "Regex": "val_accuracy: ([0-9\\.]+)"},
]

# Create a TensorFlow Estimator
tf2_california_housing_estimator = TensorFlow(
    entry_point="california_housing_tf2.py",
    source_dir="code",
    role=sagemaker.get_execution_role(),
    instance_count=1,
    instance_type="ml.m5.large",
    framework_version="2.4.1",
    hyperparameters={
        "learning_rate": 0.1,
        "epochs": 100
    },
    py_version="py37",
    metric_definitions=metric_definitions,
    enable_sagemaker_metrics=True,
    tags=[{"Key": "trial-desc", "Value": trial_desc}],
)

# Launch a training job
tf2_california_housing_estimator.fit( inputs, job_name=training_job_name)
```

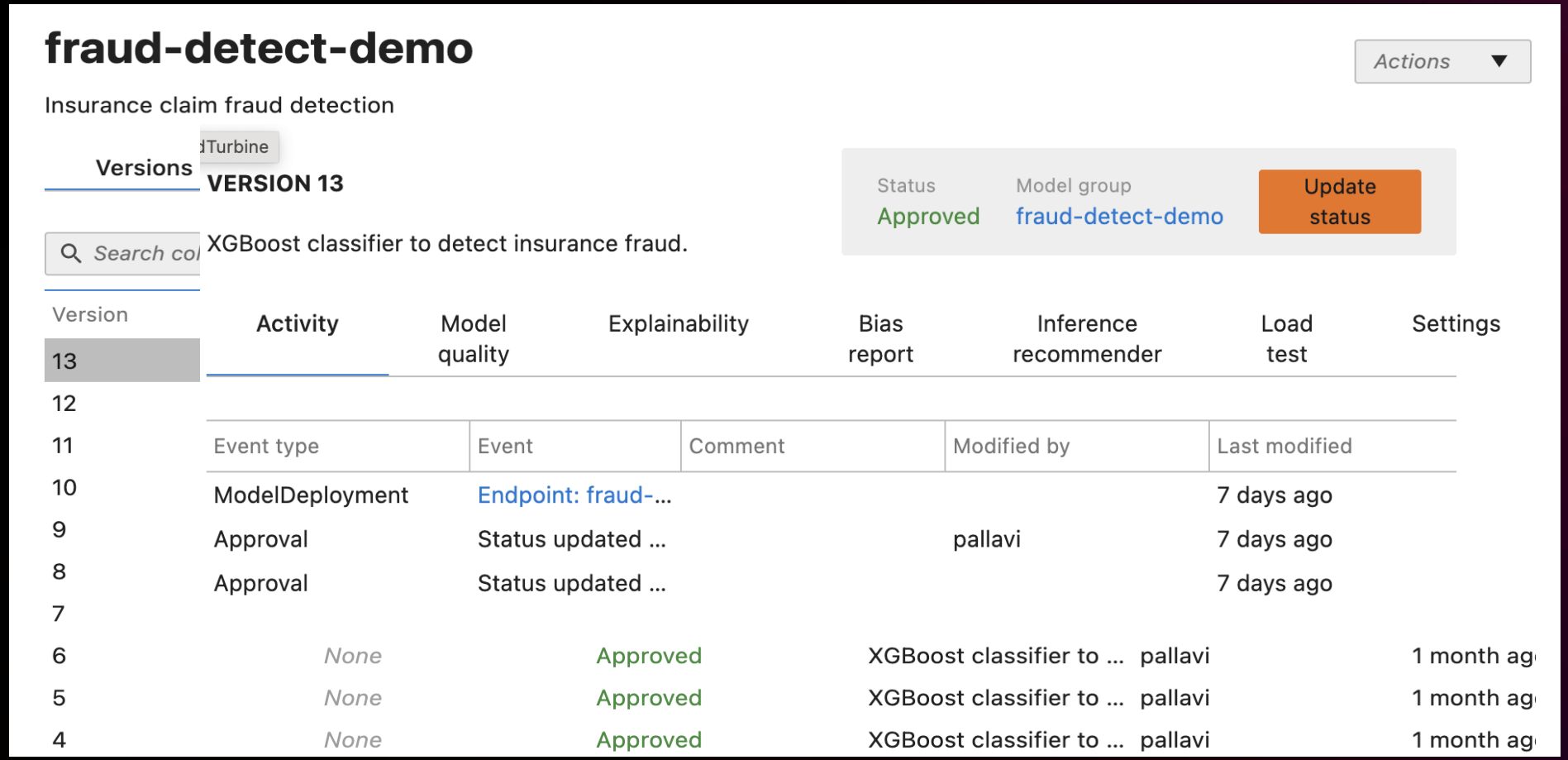
Training on SageMaker



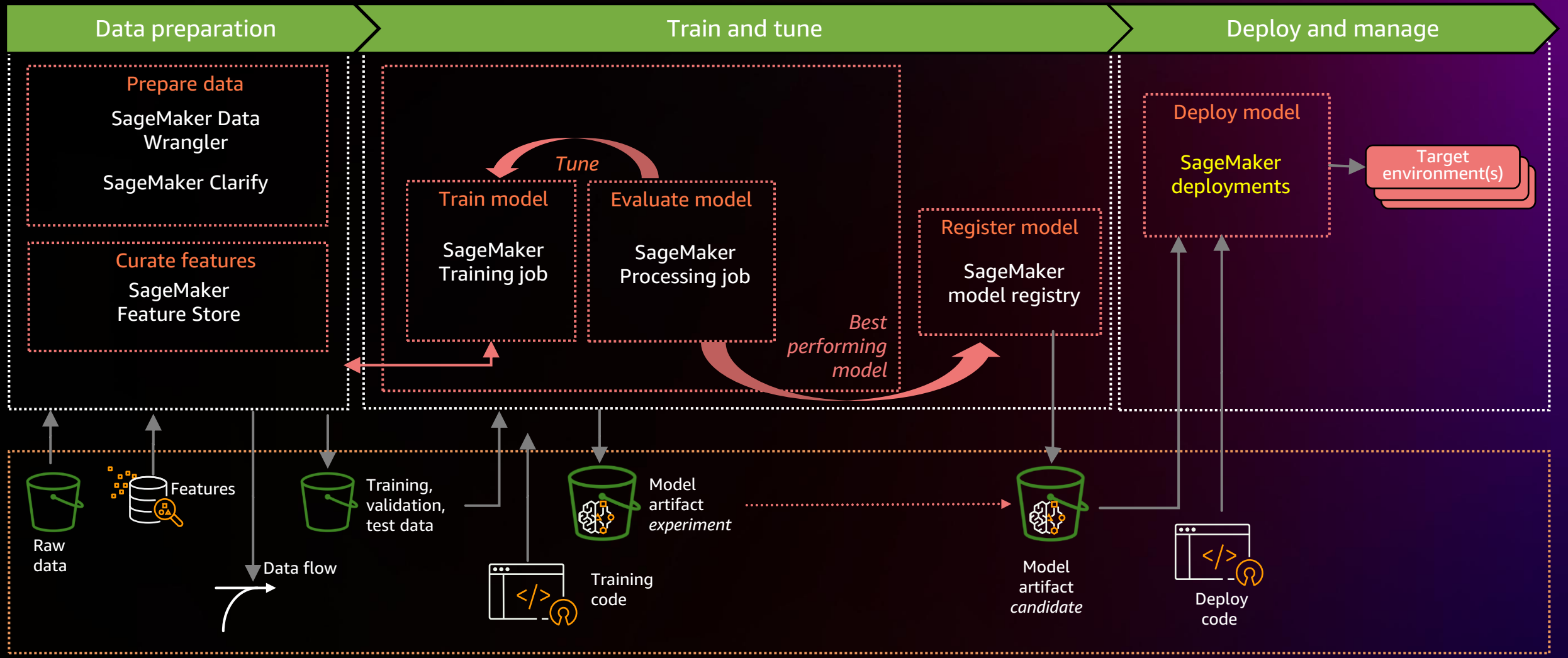
Register model



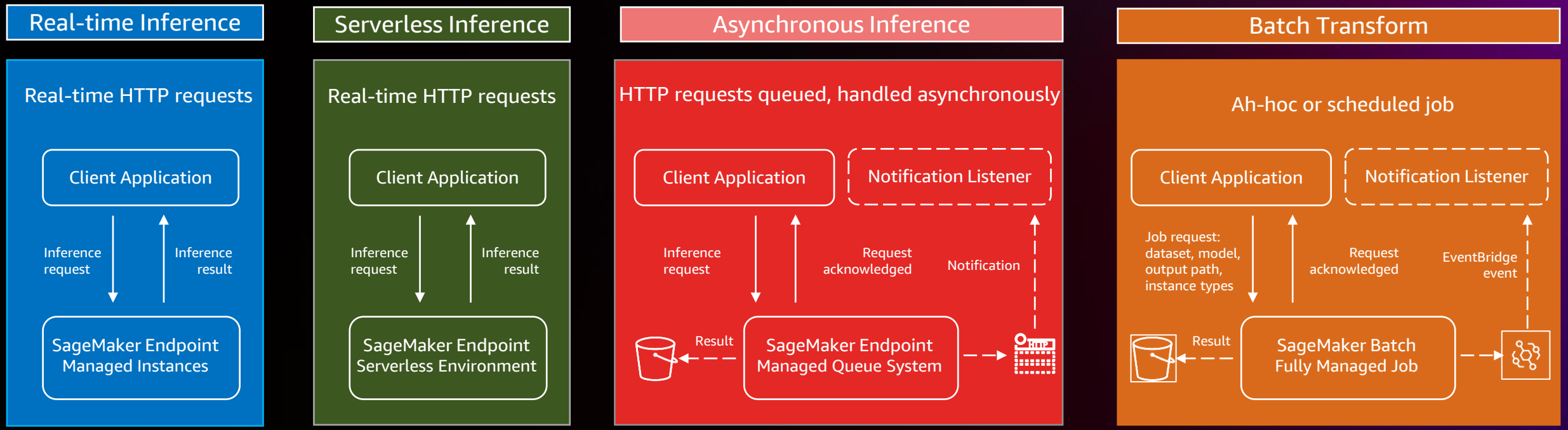
Amazon SageMaker Model Registry



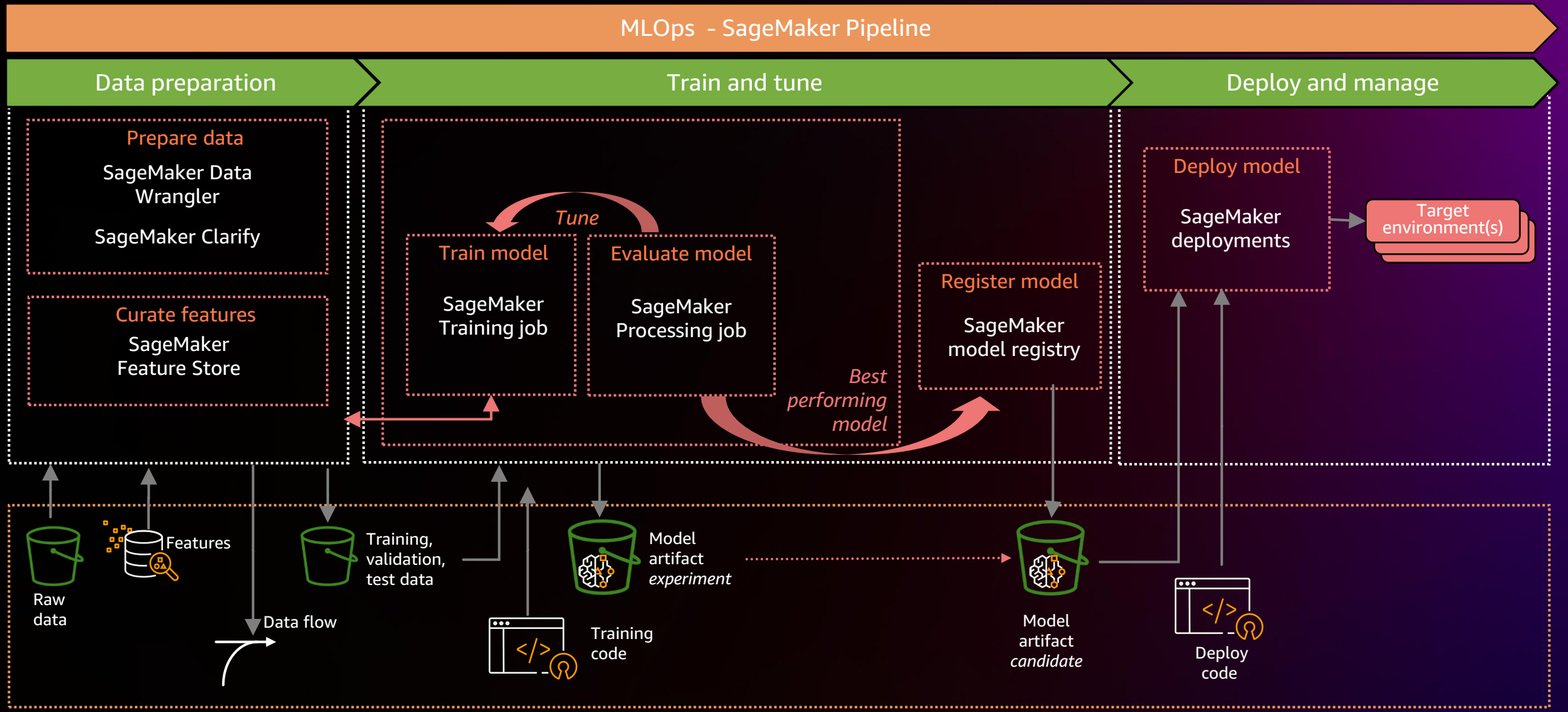
Deploy and manage



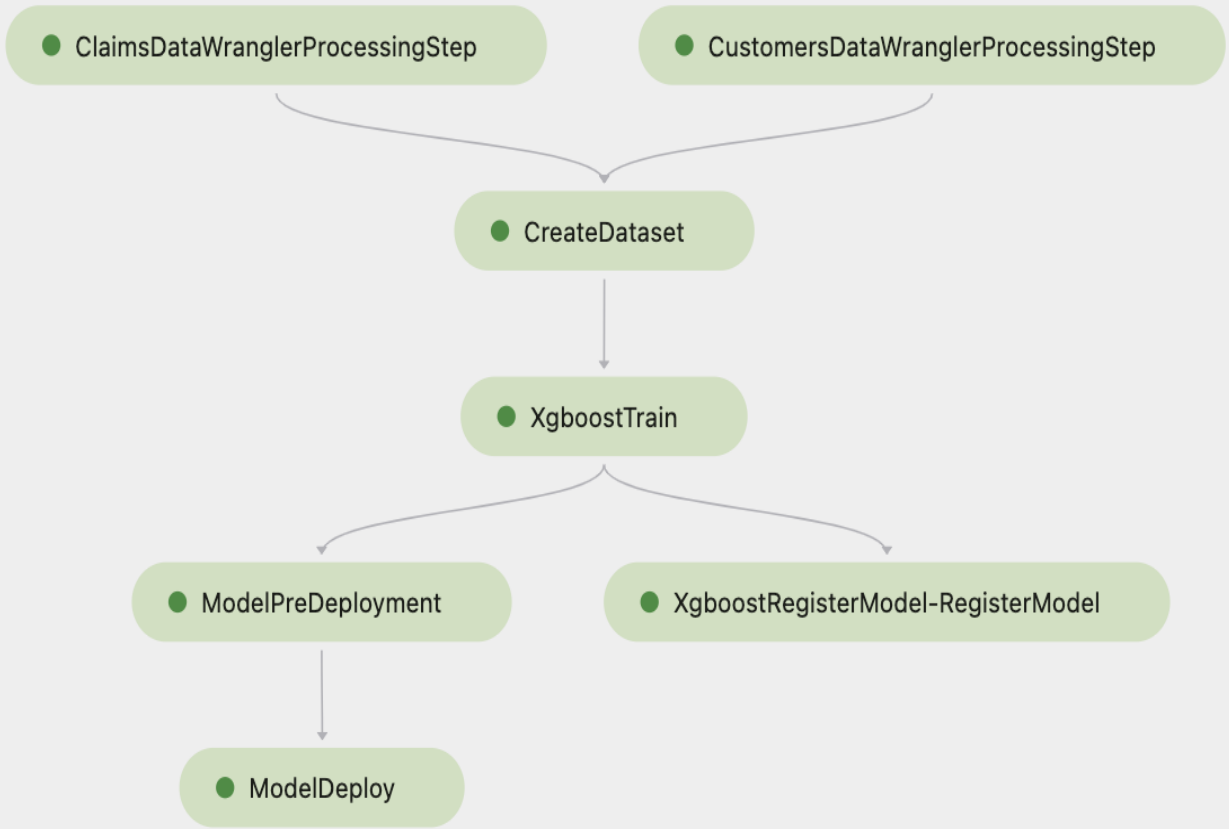
Amazon SageMaker model deployment modes



MLOps



SageMaker pipelines



Supported steps:

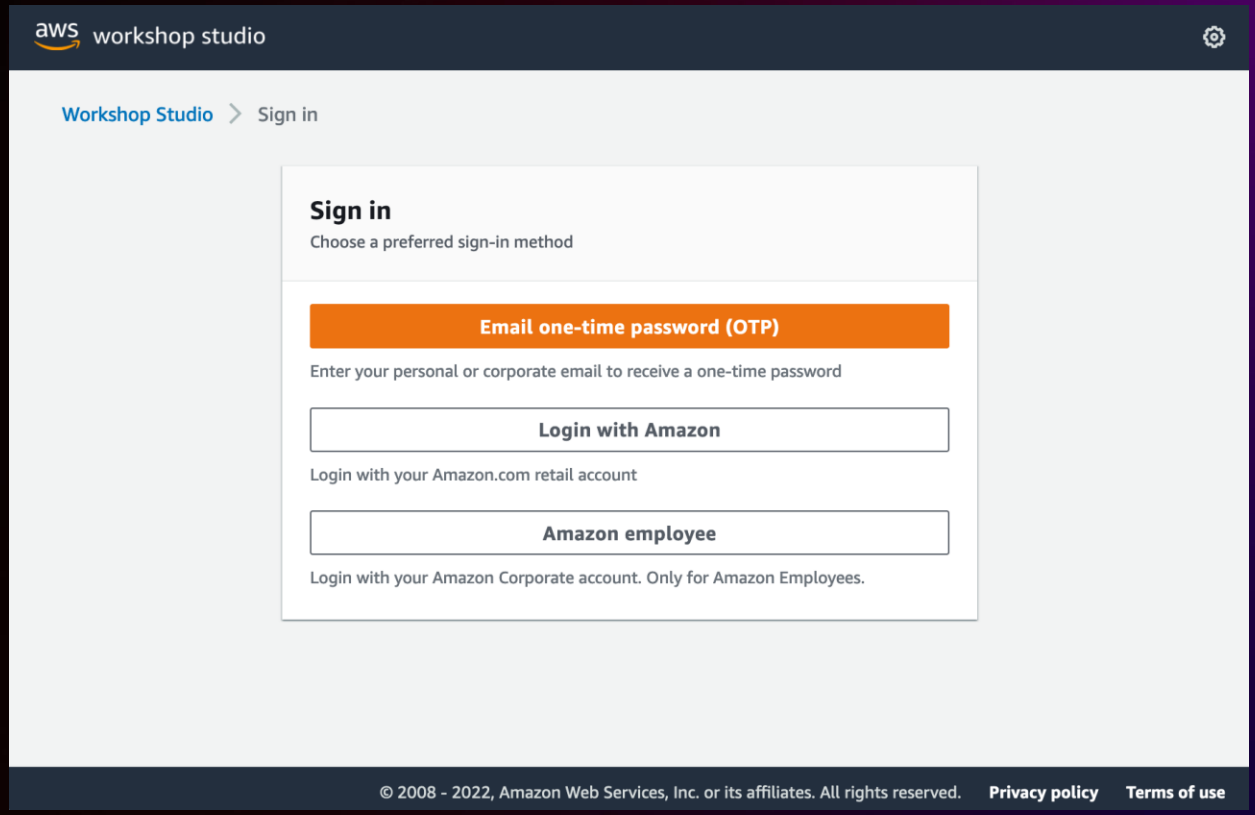
- **Processing:** Data processing and model evaluation
- **Training:** Model training using SageMaker Training jobs
- **Tuning:** Hyperparameter tuning jobs
- **Condition:** Conditional step execution
- **Transform:** Batch predictions
- **RegisterModel:** Create model package resource
- **Model/CreateModel:** Package model for deployment
- **Callback:** Incorporate additional tasks or AWS services into your workflow
- **AWS Lambda:** Run a Lambda function
- **ClarifyCheck:** Run baseline drift checks against previous baselines
- **QualityCheck:** Run baseline suggestions and drift checks against a previous baseline
- **Amazon EMR:** Run tasks on a running cluster

Getting started with this workshop

- As a participant, you will have access to an AWS account with any optional pre-provisioned infrastructure and IAM policies needed to complete this workshop.
- The AWS account will only be available for the duration of this workshop. You will lose access to the account thereafter.
- The optional pre-provisioned infrastructure will be deployed to a specific region. Check your workshop content to determine whether other regions will be used.
- Be sure to review the terms and conditions of the event. Do not upload any personal or confidential information in the account.

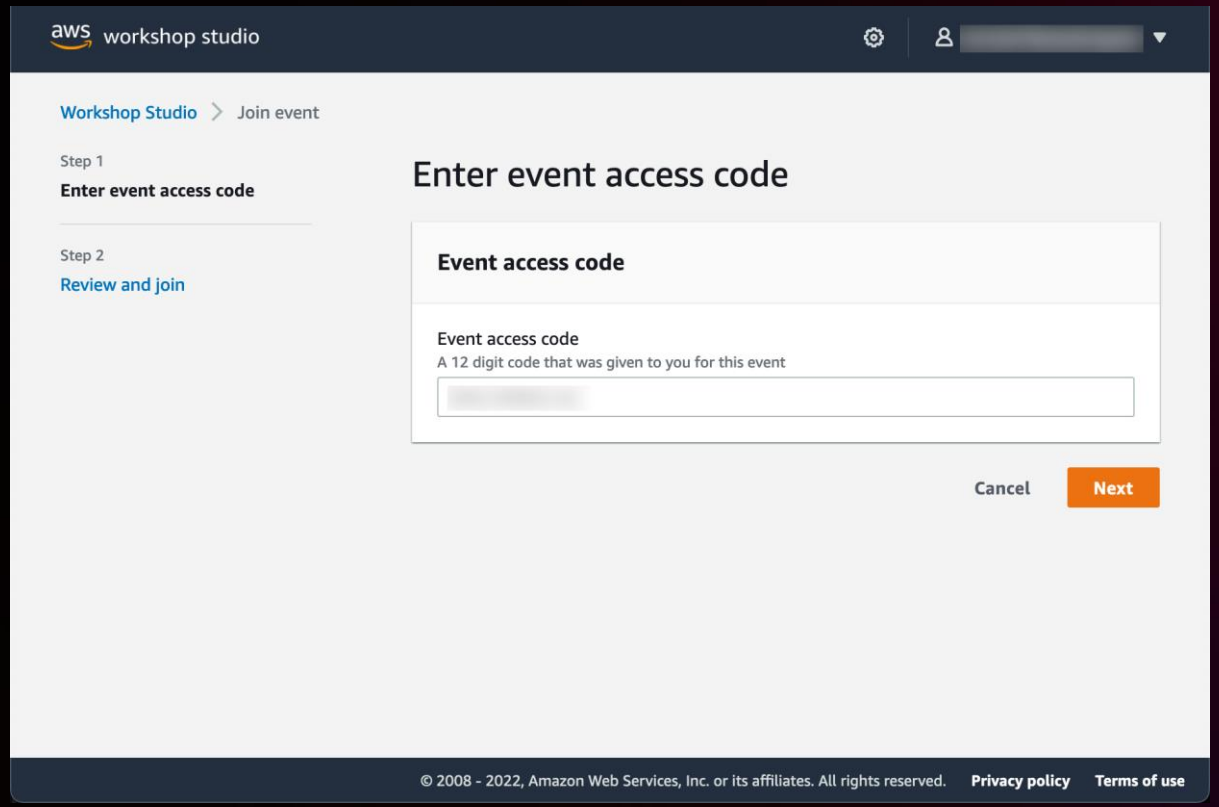
Step 1: Sign-In to AWS Account

bit.ly/STP303-2022

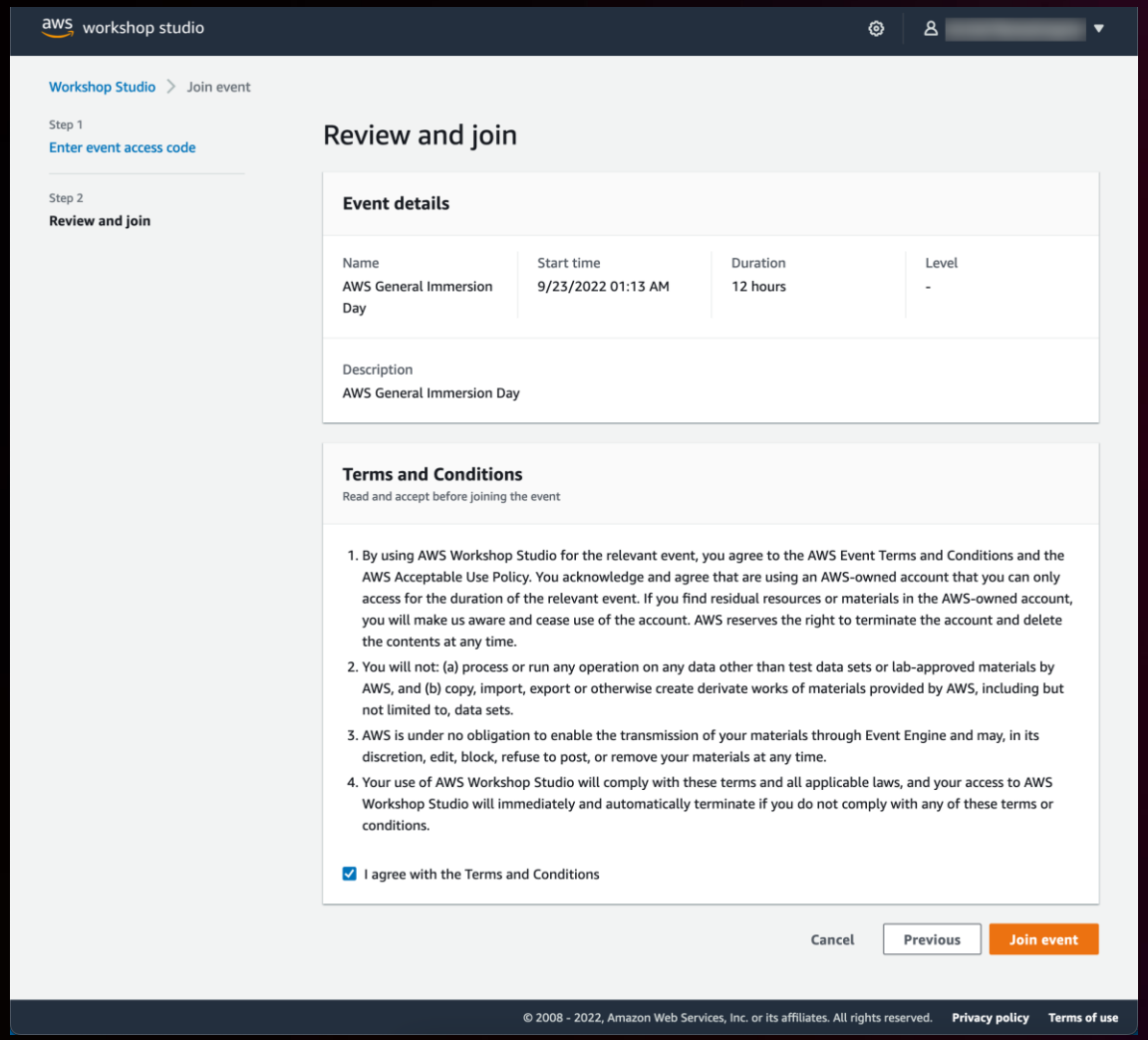


Step 2: Enter event access code

Enter 12 digit event access code.

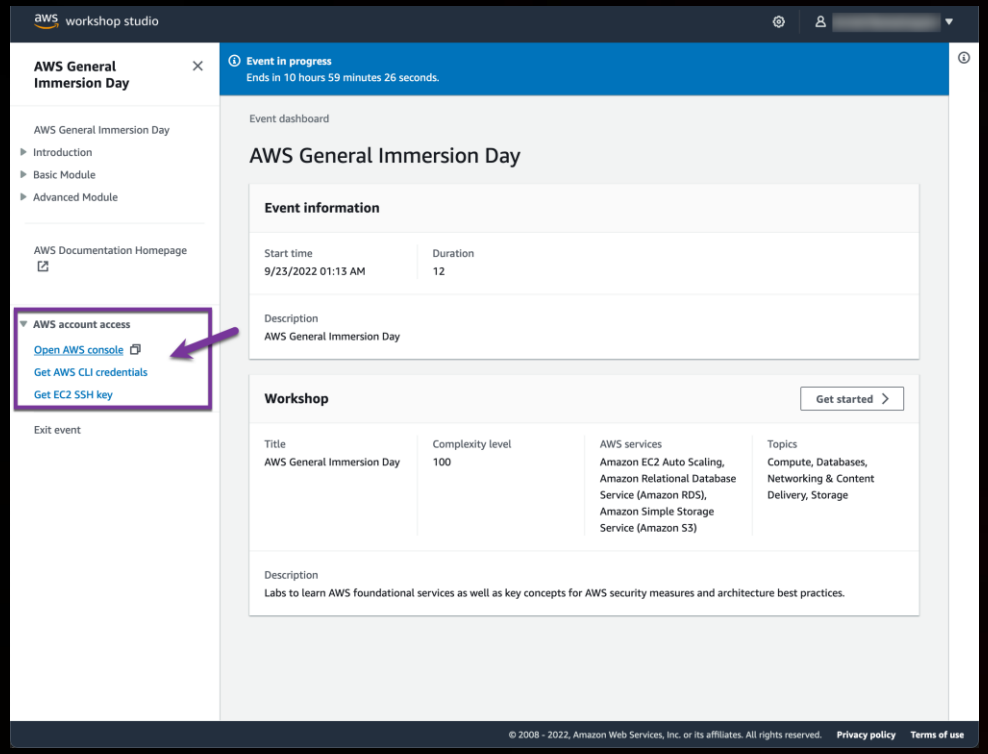


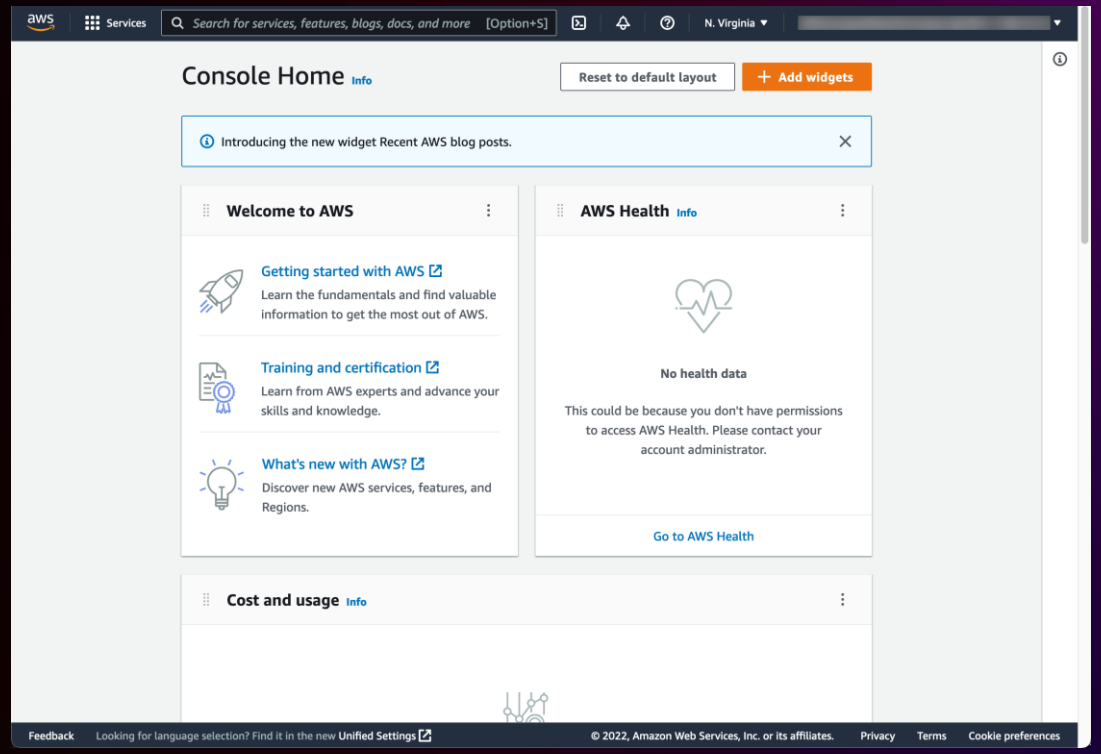
Step 3: Review terms and join event



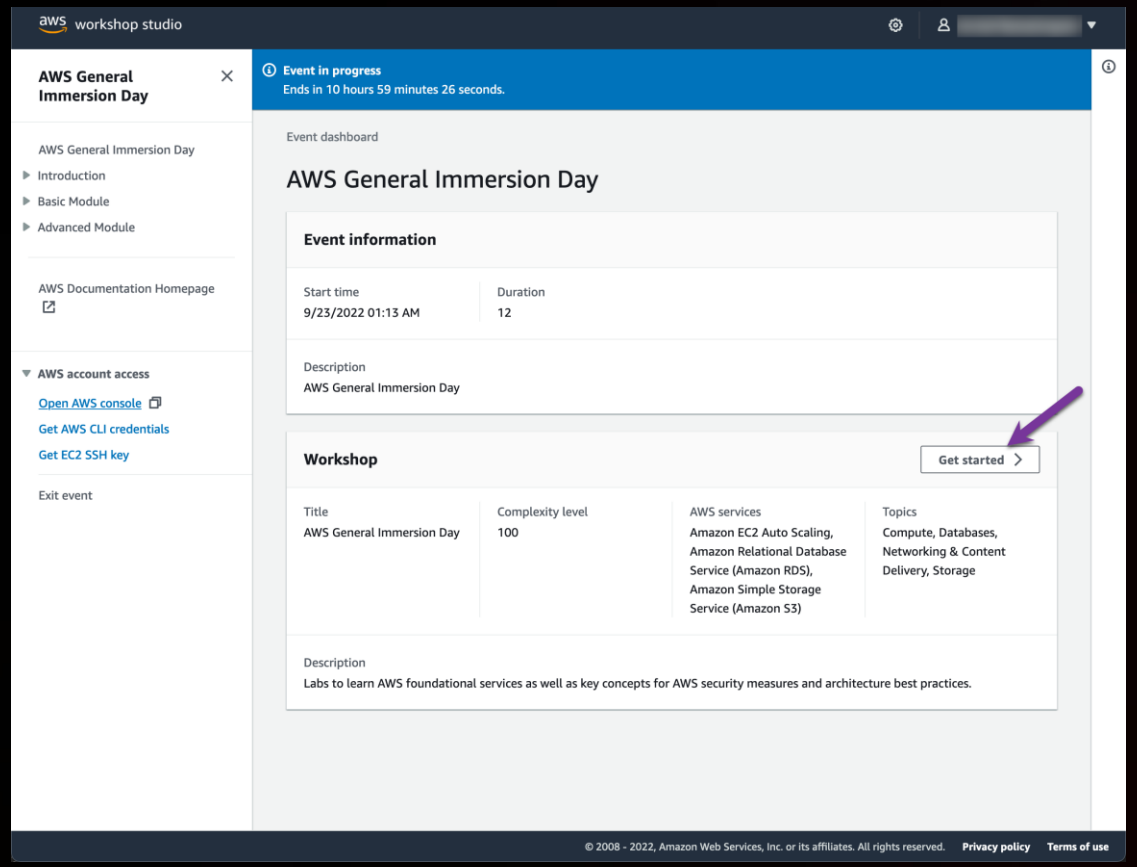
Step 4: Access AWS account

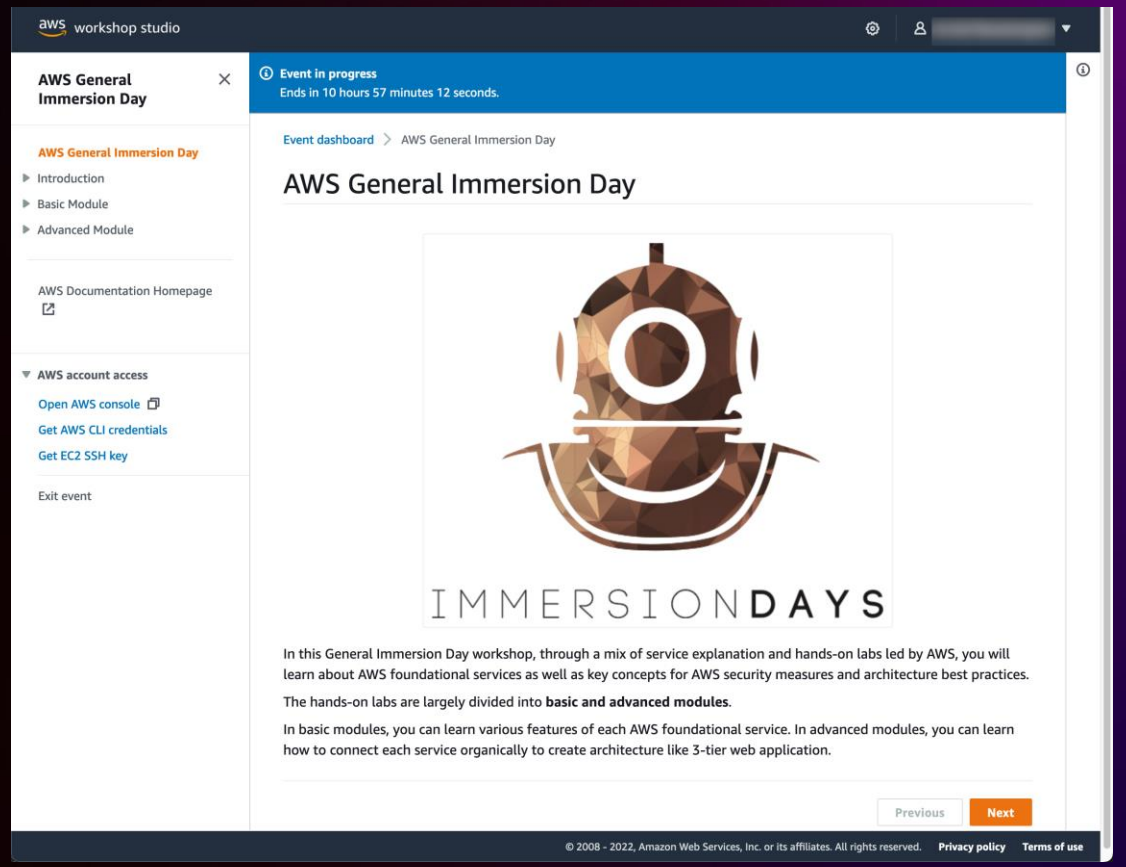
Access the AWS Console, or generate AWS CLI credentials as needed





Step 5: Get started with the workshop





Thank you!

Manish Chugh
manchugh@amazon.com

Pallavi Nargund
nargundp@amazon.com

Please complete the session survey in the **mobile app**