aws health

EMEA HEALTHCARE & LIFE SCIENCES WORKSHOPS

Machine Learning on AWS

Yegor Tokmakov
Sr. Solutions Architect
AWS Health

Dmitri Laptev
Sr. Solutions Architect
AWS Startups

Aamna Najmi
Data Scientist
AWS Professional Services

Agenda

- 13:00 - 13:15 Welcome and introductions
- 13:15 - 13:45 Amazon SageMaker Introduction
- 13:45 - 14:30 Lab - Train, Tune and Deploy model using SageMaker Built-in Algorithm
- 14:30 - 15:00 Create model, prediction and inference
- 15:00 - 15:45 Lab - Bring your custom model
- 15:45 - 16:00 Coffee break
- 16:00 - 16:30 Advanced ML: Transformers, LLMs, HuggingFace
- 16:30 - 17:15 Lab - Summarize Scientific Documents with Amazon SageMaker and HuggingFace
- 17:15 - 17:30 Conclusion, QA

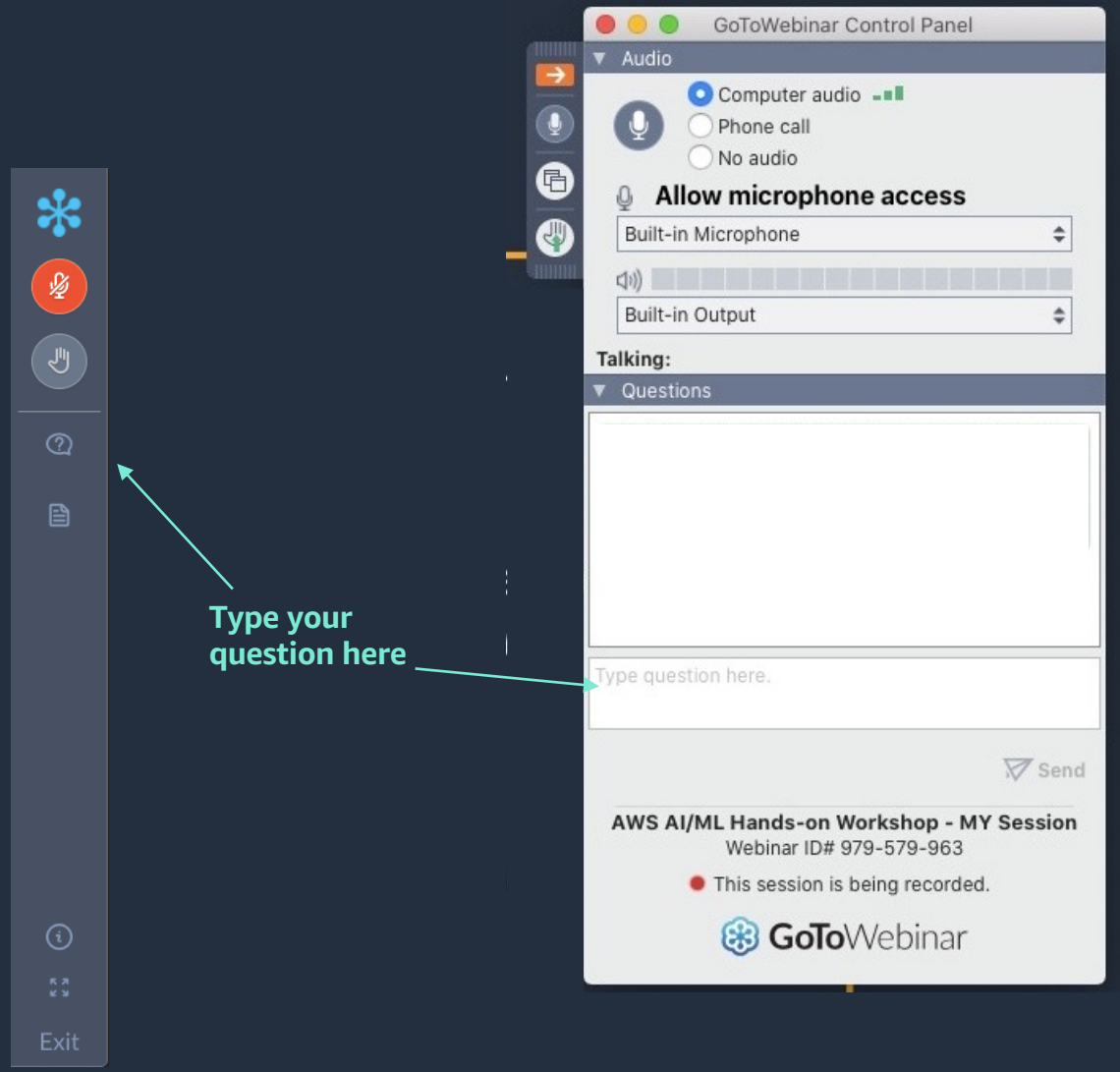
Workshop materials and instructions

- https://catalog.us-east-1.prod.workshops.aws/workshops/63069e26-921c-4ce1-9cc7-dd882ff62575/en-US
- https://catalog.workshops.aws/hcls-aiml/en-US

Questions & answers

If you have any questions or encounter issues during the workshop, our support team is online.

You can submit your query in the GoToWebinar Questions function. To submit questions, select
“Send”



Amazon SageMaker Introduction

The reach of ML is growing

INCREASED SPENDING

By 2025, global spending on artificial intelligence will reach $204 billion

—IDC

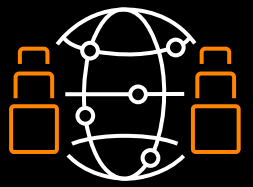
IDC, “Worldwide Spending on Artificial Intelligence,” https://bit.ly/3y7hDoP

FROM PILOTING TO OPERATIONALIZING

By the end of 2024, 75% of enterprises will shift from piloting to operationalizing AI

—Gartner

Gartner, “Gartner Identifies Top 10,” https://gtnr.it/3Bln3uU



AI TRANSFORMATION

57% said that AI would transform their organization in the next three years

—Deloitte

Deloitte, “Thriving in the Era of Pervasive AI” https:// bit.ly/3JSeCi6

aws health

AI/ML with AWS
Innovation, choice, and flexibility

100,000+
customers have used machine learning (ML) on AWS

250+
new capabilities for ML and artificial intelligence (AI) in just the last 12 months

92% of deep learning (DL) in the cloud runs on AWS

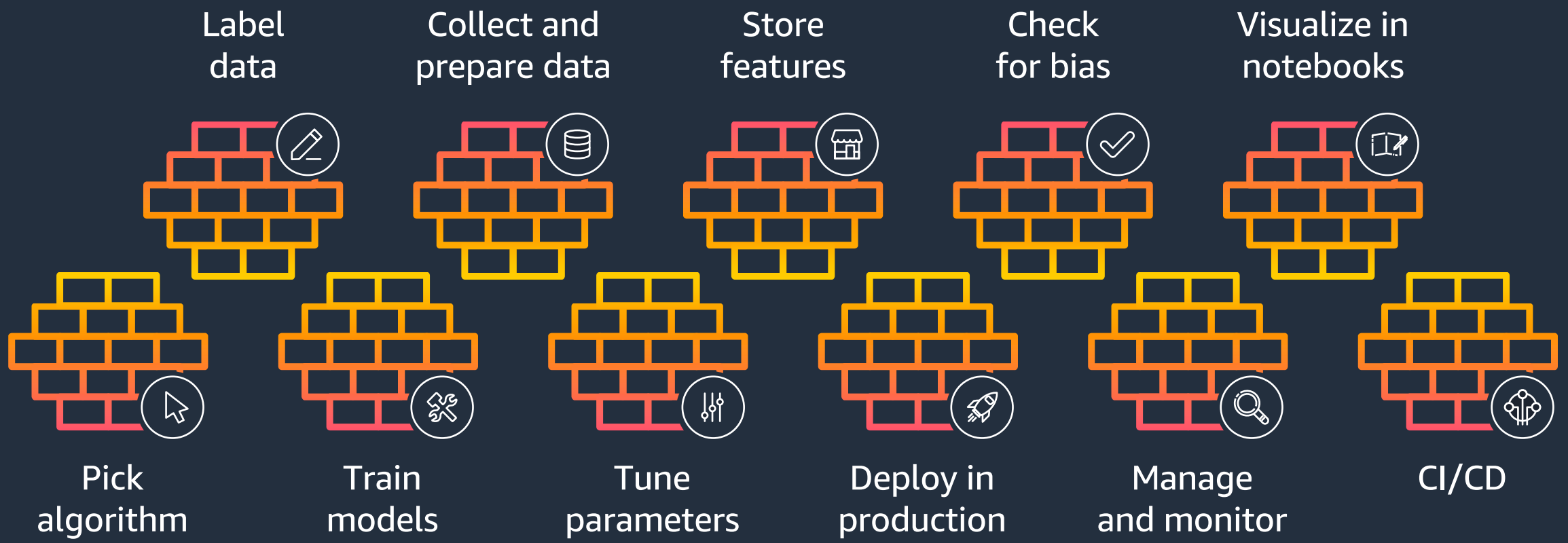
91% of cloud-based PyTorch runs on AWS

AWS ML SOLUTIONS

Reduce training time by 50%
Provide 90% scaling efficiency
Deliver 3x faster network throughput
Improve price and performance by 25%

aws health

Machine learning development is complex and costly



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

CONVERSATION
- Amazon Lex
- Amazon Transcribe Call Analytics
- Contact Lens
- Voice ID

CODE + DEVOPS
- Amazon CodeGuru
- Amazon CodeWhisperer
- Amazon DevOps Guru

INDUSTRIAL
- Amazon Monitron
- Amazon Lookout for Equipment
- Amazon Lookout for Vision

HEALTH
- Amazon HealthLake
- Amazon Comprehend Medical
- Amazon Transcribe Medical
- Amazon Omics

CORE

TEXT
- Amazon Translate
- Amazon Comprehend

SPEECH
- Amazon Polly
- Amazon Transcribe

VISION
- Amazon Textract
- Amazon Rekognition
- AWS Panorama

AMAZON SAGEMAKER

CANVAS
No-code ML for business analysts

STUDIO LAB
Learn ML

GROUND TRUTH
Label data

STUDIO IDE
- Prepare data
- Geospatial ML
- Build with notebooks
- Train models
- Deploy in production
- Store features
- Tune parameters
- Manage and monitor

CI/CD | GOVERNANCE | RESPONSIBLE ML

EDGE MANAGER
Manage edge devices

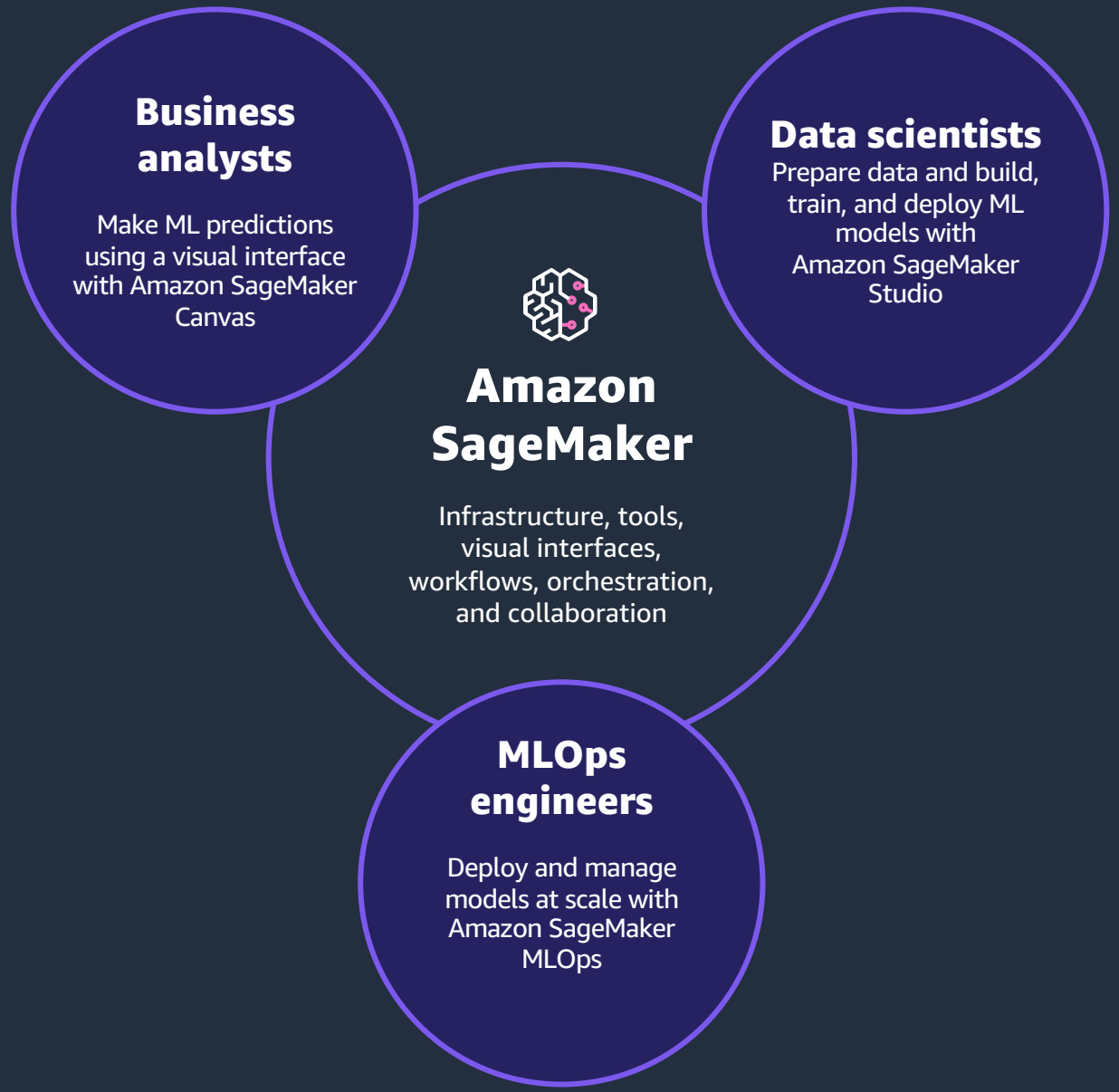
ML FRAMEWORKS & INFRASTRUCTURE

PyTorch, Apache MXNet, TensorFlow | Amazon EC2 | CPUs | GPUs | AWS Inferentia | AWS Trainium | Habana Gaudi | FPGA

aws health

What is SageMaker?

Amazon SageMaker helps organizations harness ML



aws machine learning

Overcoming the barriers to ML

Not enough ML builders

No-code ML tools

Make ML predictions regardless of ML experience

Access, process, and label massive volumes of data for ML

Purpose-built data preparation tools

Access, process, and label data for ML

Disparate data science tools

Integrated ML tools in a single interface

Build, train, and deploy models using IDEs

Tedious, manual ML operations

Built-in MLOps

Automate and standardize MLOps practices

Challenging to govern ML projects efficiently

Out-of-box ML governance tools

Simplify access control and enhance transparency across ML lifecycle

aws health

Amazon SageMaker feature tour

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

PREPARE

Geospatial
Visualize geospatial data

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

Shadow Testing
Validate model performance in production

Inference Recommender
Automatically select compute instance and configuration

Model Monitor
Maintain accuracy of deployed models

Kubernetes Operators & Components
Manage and monitor models on edge devices

Edge Manager
Manage and monitor models on edge devices

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

Canvas
Generate accurate machine learning predictions—no code required

Studio | RStudio
Integrated development environment (IDE) for ML

Governance
Model Cards | Dashboard | Permissions

aws health

Amazon SageMaker Data Wrangler

EXPLORE, PREPARE, AND PROCESS DATA WITH LITTLE TO NO CODE

Import data from multiple sources

Get insights on data and data quality

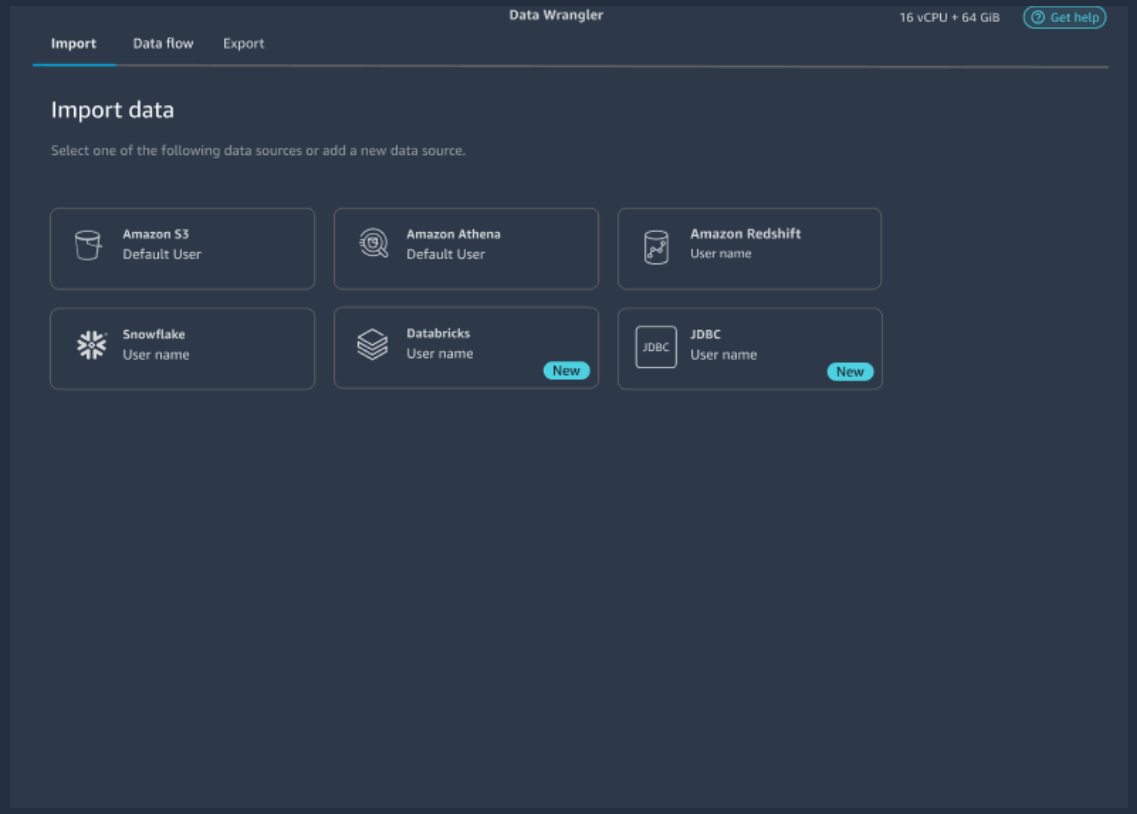
Visually explore, analyze, and prepare data

Quickly perform feature engineering

Automate ML data preparation workflows

aws health

Quickly connect and query data (1/3)



Connect to multiple data sources:

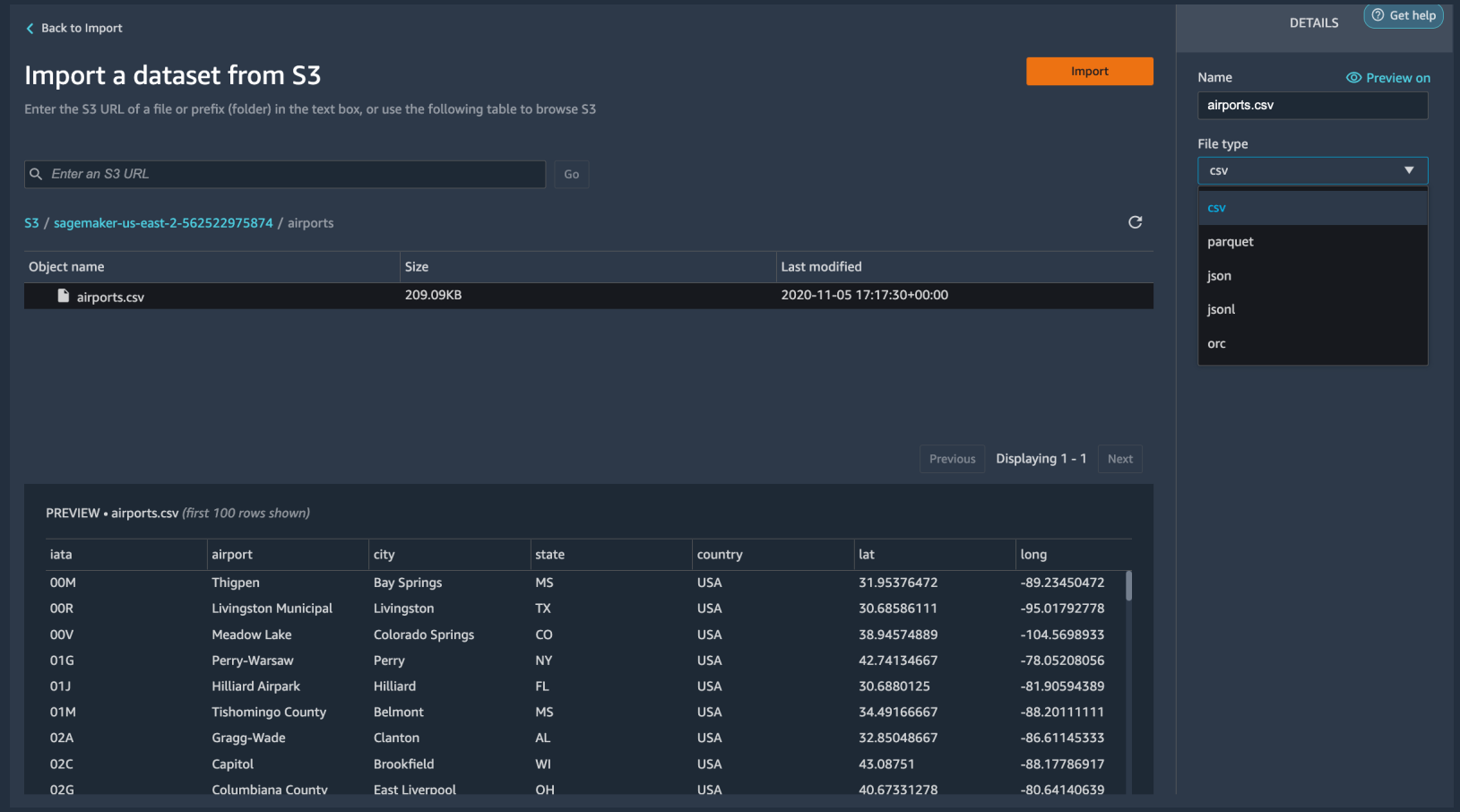
Amazon S3
Amazon Athena
Amazon Redshift
Snowflake
Databricks DeltaLake (NEW)
SageMaker Feature Store & more coming

Support for common file formats:

CSV files
Parquet files
JSON & JSONL (NEW)
ORC file (NEW)
Database tables

aws health

Quickly connect and query data (2/3)



Visually browse data sources like objects on S3, or database, schema, tables and objects in Snowflake

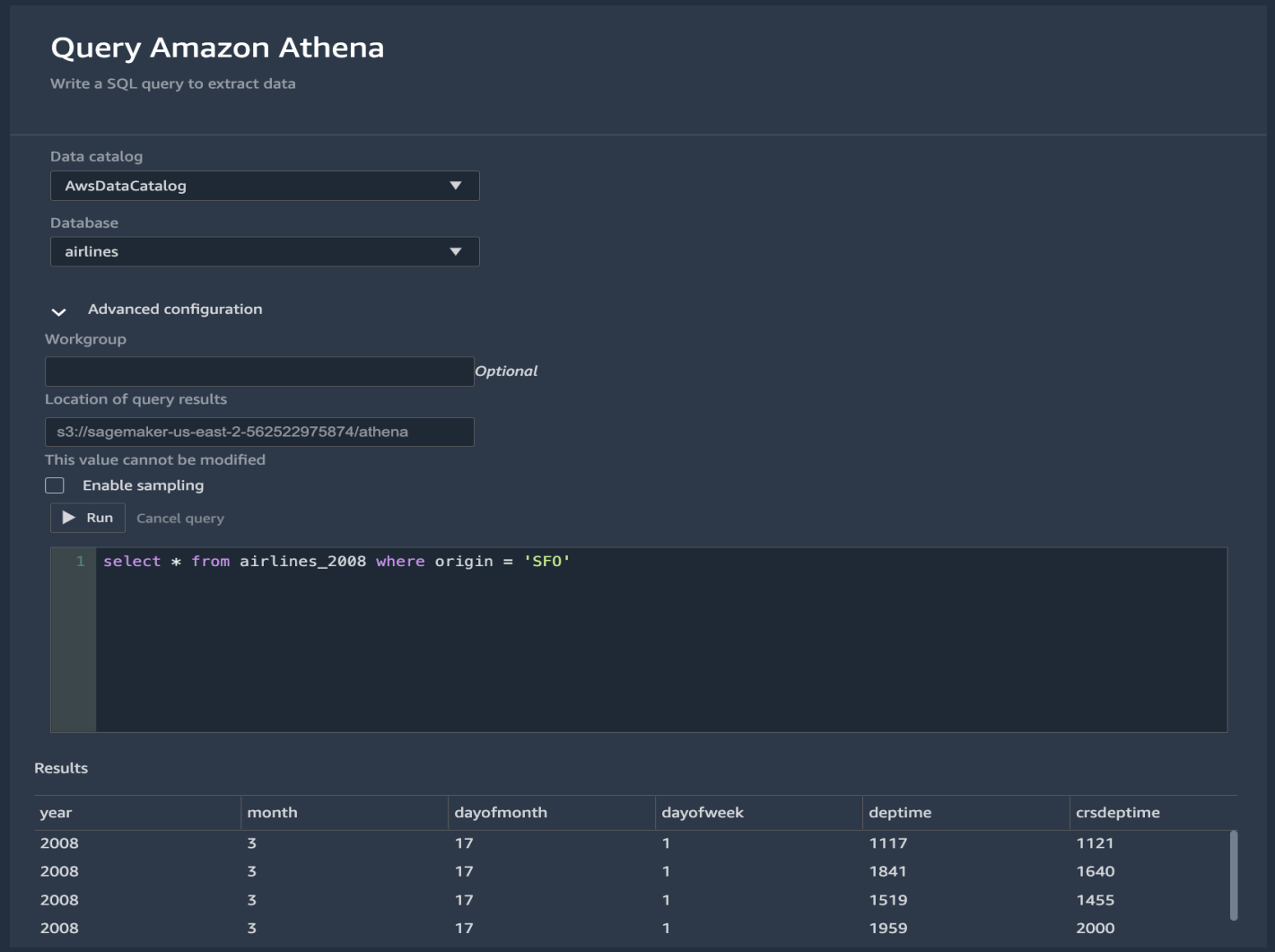
Preview & sample top rows

Join data from multiple sources

Support for VPC, KMS, CMK, and AWS Secrets Manager

aws health

Quickly connect and query data (3/3)



Query data using Amazon Athena

Browse data catalog, databases, & tables

Write queries & preview before importing data

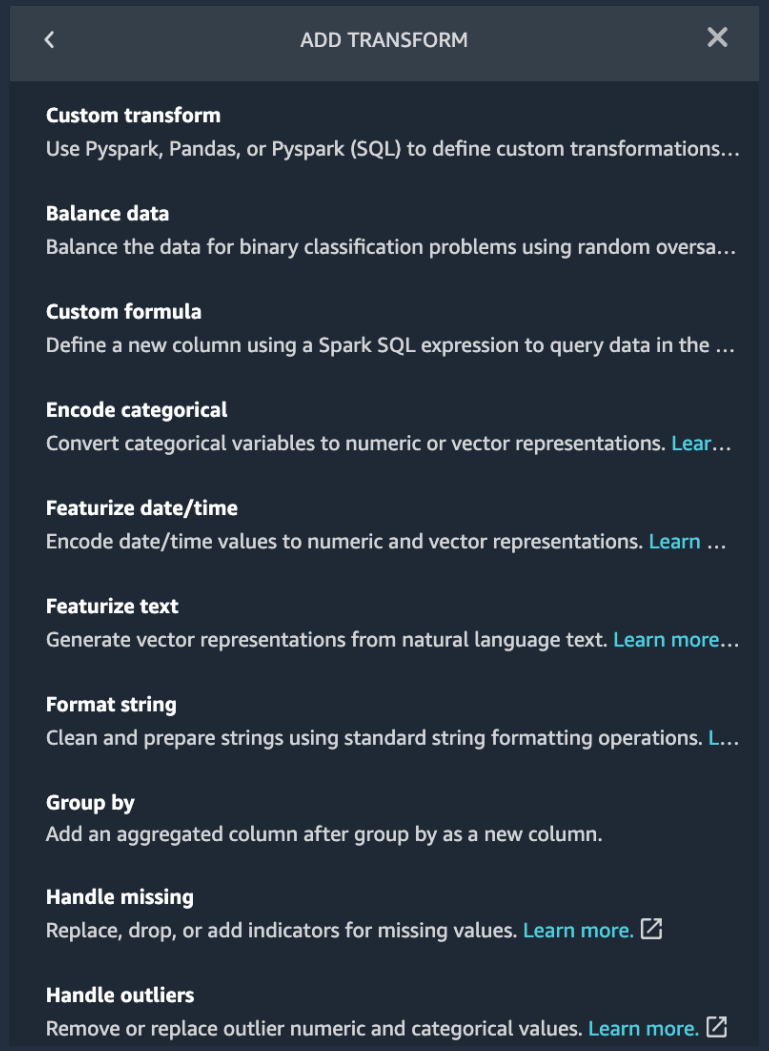
Includes support for Athena Workgroups & custom output bucket

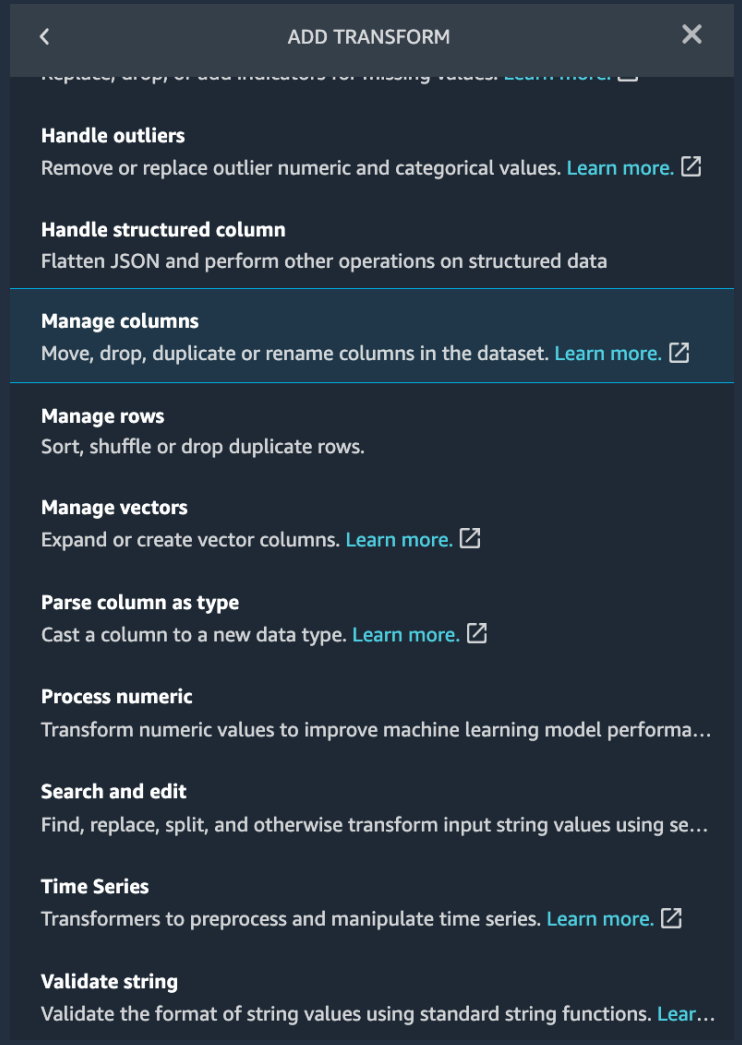
Easily transform data for ML with 300+ built-in transforms

300+ built-in data transformations (no code) for common data prep needs and ML specific needs

Built by data scientists for data scientists

ML specific transforms such as:
One hot encoding
Balance data
Time series transforms





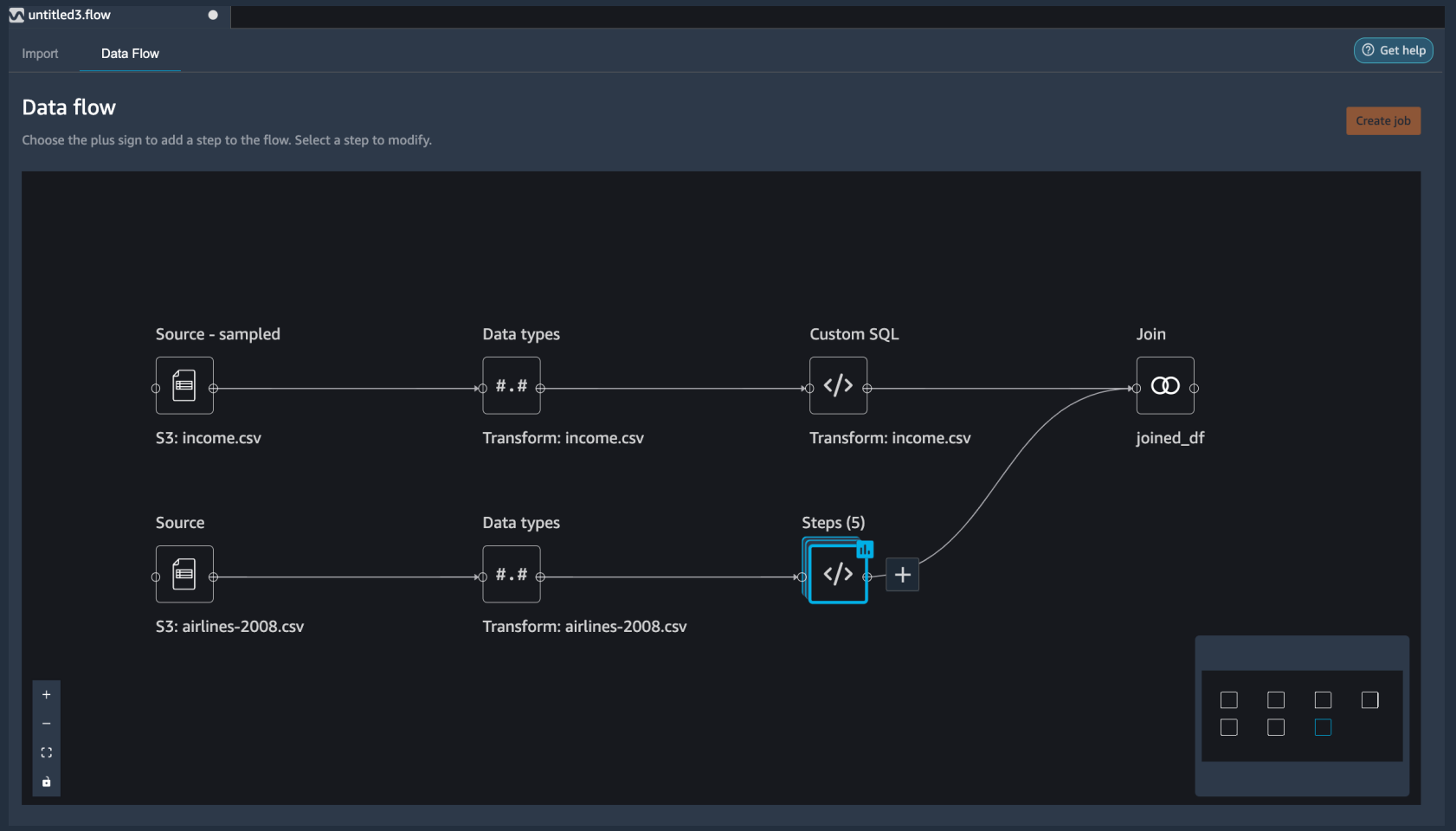
aws health

Easily visualize the steps of your data processing pipeline

Data Wrangler records all the steps of data prep workflow in a data flow graph

Visualize the order of transformations, join and concatenate operators

Easily navigate data transformation flow, and modify and delete steps iteratively



Amazon SageMaker Feature Store

SECURELY STORE, DISCOVER, AND SHARE FEATURES FOR ML

Online and off-line

Millisecond latency

Consistent features

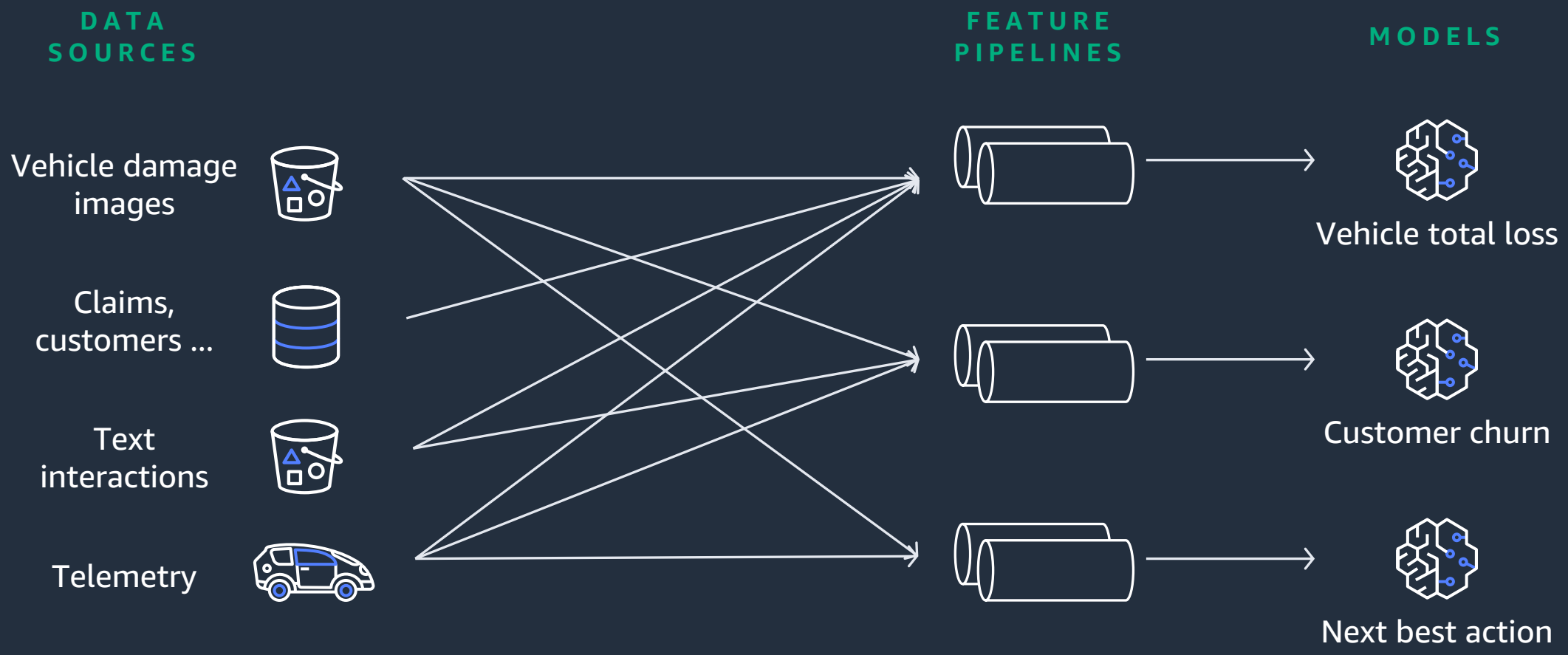
Visual search

Sharing and collaboration

aws health

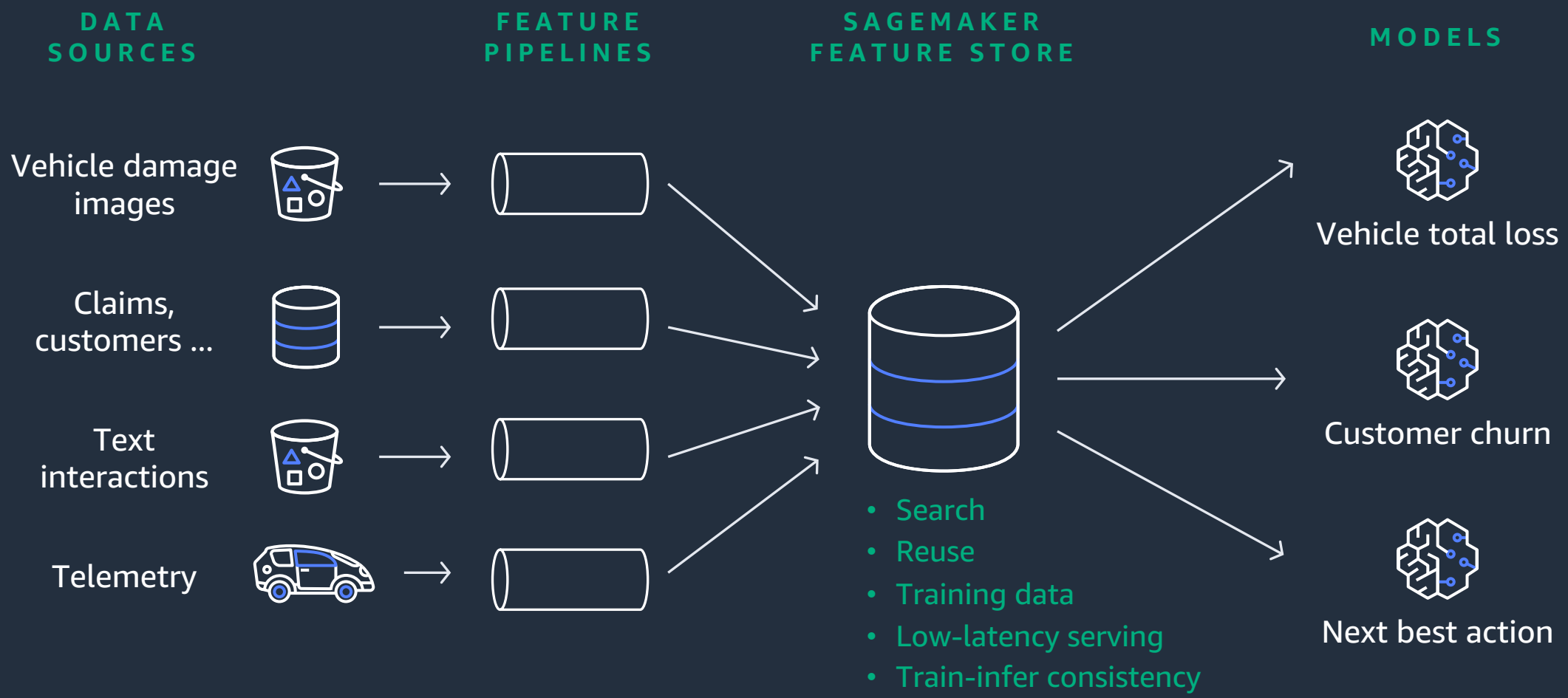
Without a feature store ...

Standalone feature engineering for each new model



With SageMaker Feature Store ...

Build features once, and reuse them across teams and models



Support for separate feature stores

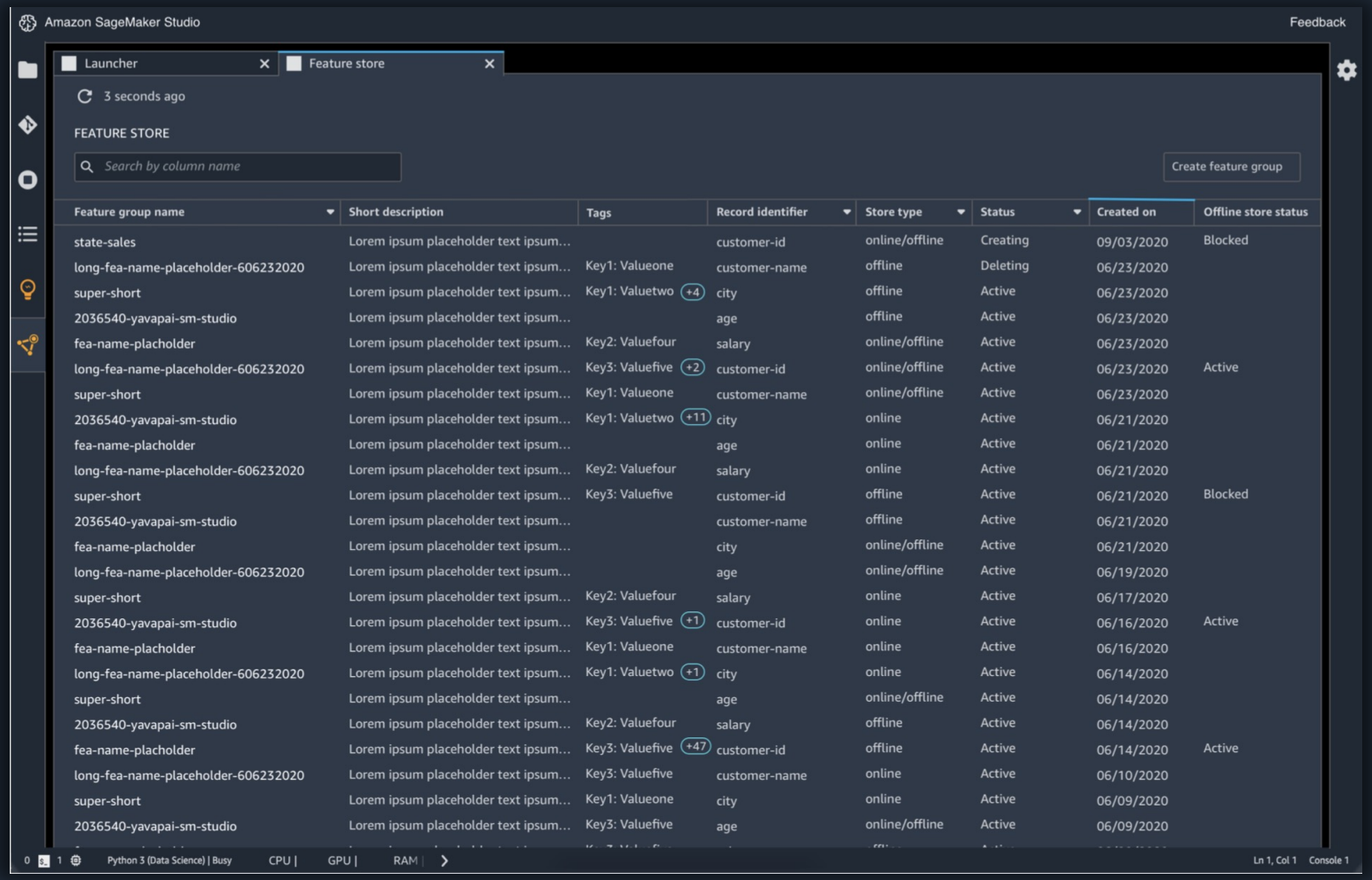
Online feature store

- Primarily used for real time predictions
- Use cases such as real-time fraud detection
- Latest copy of feature data
- High throughput writes
- Low millisecond latency reads

Offline feature store

- Primarily used for batch predictions and model training
- Historical record of feature data
- High throughput writes
- <15 minutes read after write consistency

Search and discover features using Feature Store



- Search features individually or by groups visually with SageMaker Studio
- Discover features by name, description, tags, and other metadata
- Understand how features are grouped relevant to ML applications

aws health

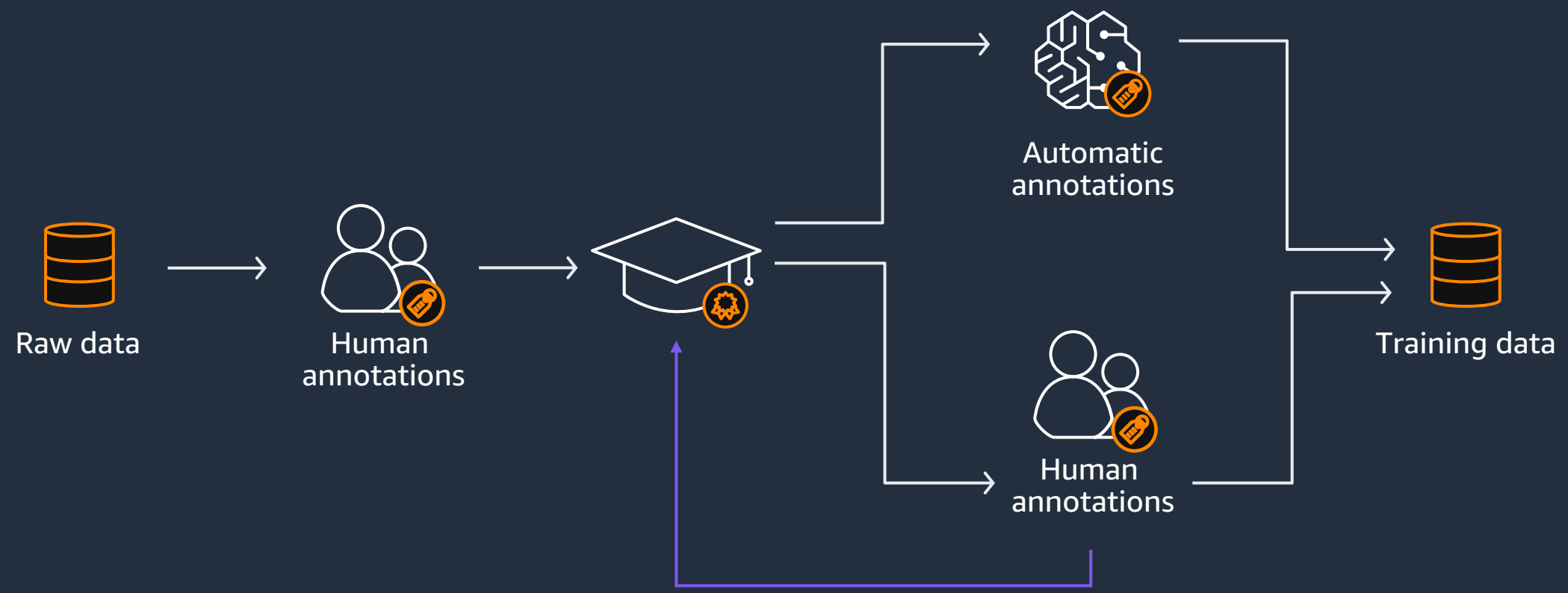
Amazon SageMaker Ground Truth

Build highly accurate training datasets using machine learning

- Reduce data labeling costs by up to 70%
- Access labelers through Amazon Mechanical Turk, Amazon approved vendors, or use private human labelers
- Achieve accurate results quickly

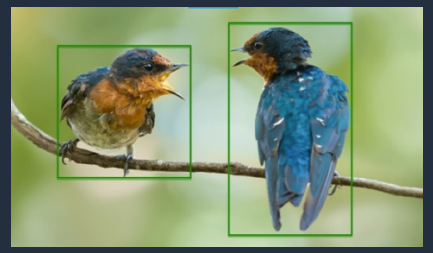
aws health

How Amazon SageMaker Ground Truth Works



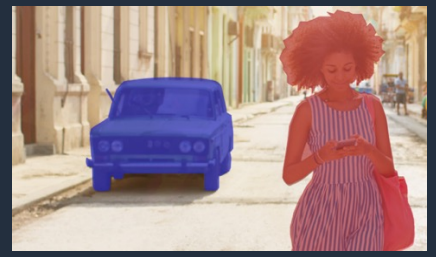
Turnkey data labeling service

70+ workflows

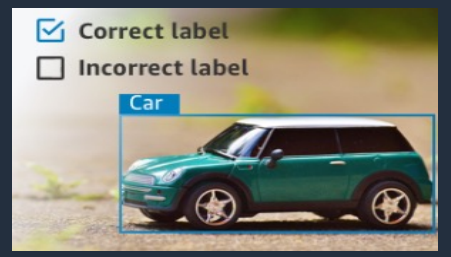


Bounding boxes

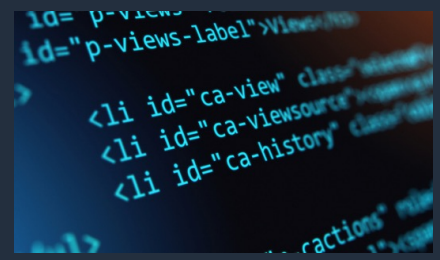
Image classification



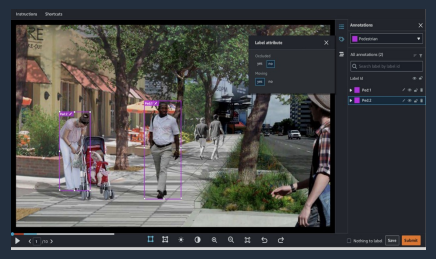
Segmentation



Label verification

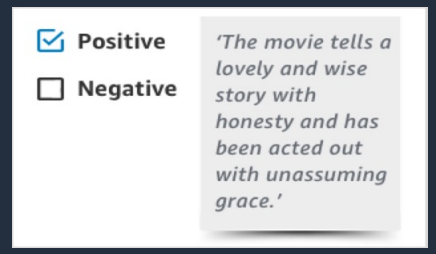


Custom template



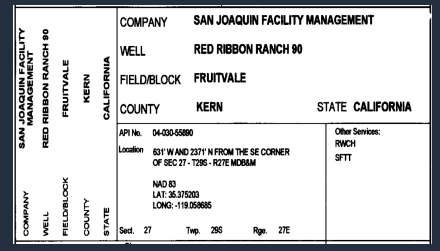
Video

LIDAR 3D Point Cloud



Text classification

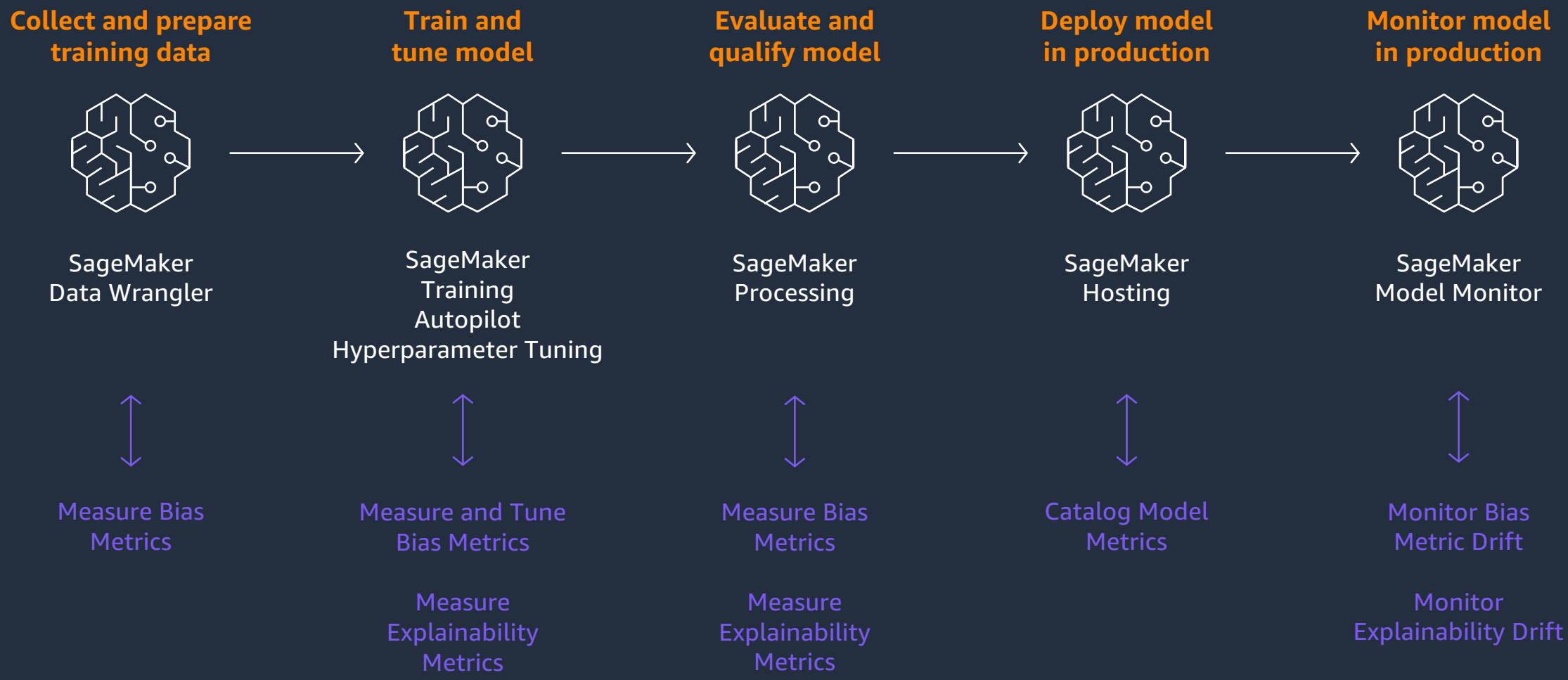
Named entity recognition



OCR, form, table

aws health

SageMaker Clarify works across the ML lifecycle



PREPARE

Geospatial
Visualize geospatial data

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

Shadow Testing
Validate model performance in production

Inference Recommender
Automatically select compute instance and configuration

Model Monitor
Maintain accuracy of deployed models

Kubernetes Operators & Components
Manage and monitor models on edge devices

Edge Manager
Manage and monitor models on edge devices

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

Canvas
Generate accurate machine learning predictions—no code required

Studio | RStudio
Integrated development environment (IDE) for ML

Governance
Model Cards | Dashboard | Permissions

aws health

Model options

Training code

AWS Marketplace for Machine Learning

Amazon SageMaker AutoPilot

- XGBoost - Gradient Boosted Trees
- Matrix Factorization
- Regression
- Principal Component Analysis
- K-Means Clustering
- And More!

Built-in Algorithms
No ML coding required

Bring Your Own Script
Amazon SageMaker builds the container
Open source containers

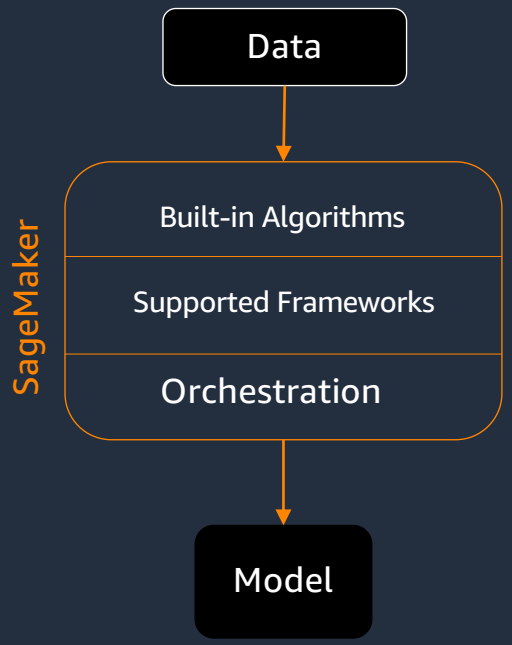
Bring Your Own Container
Full control, you build the container
R, C++, etc

Fully Managed, Distributed, Auto-Scaled, Secured

aws health

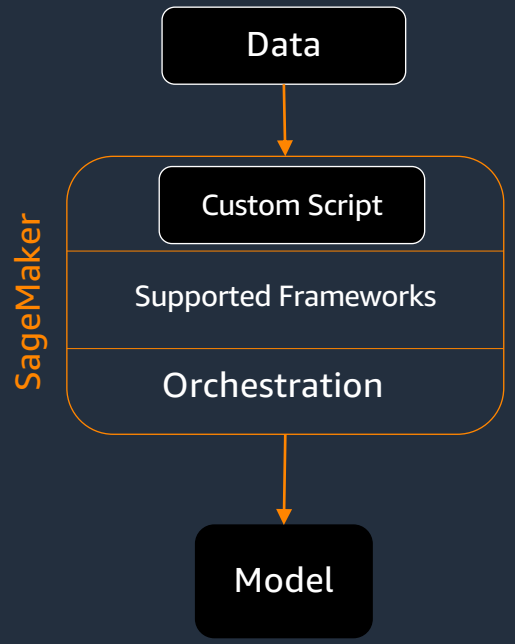
Amazon SageMaker | Training

Sagemaker Built-in algorithms



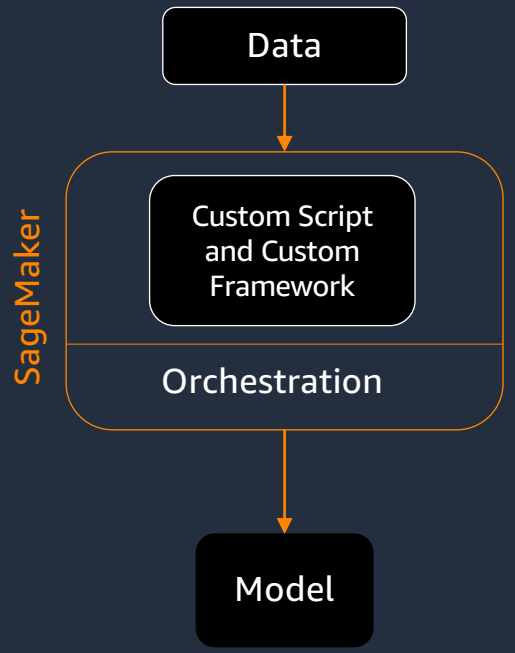
Built in high performance algorithms

Custom script on supported framework



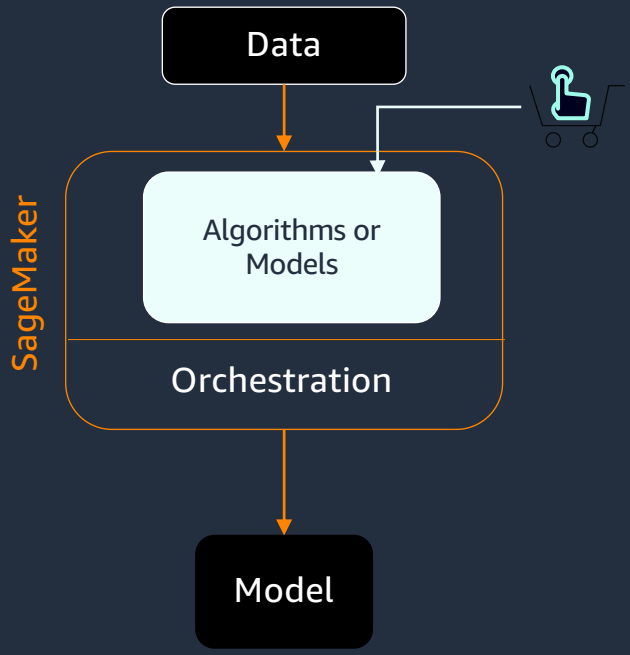
Supported Frameworks: Apache MXNet, TensorFlow , Scikit-learn, PyTorch, Chainer

BYO algorithm and framework



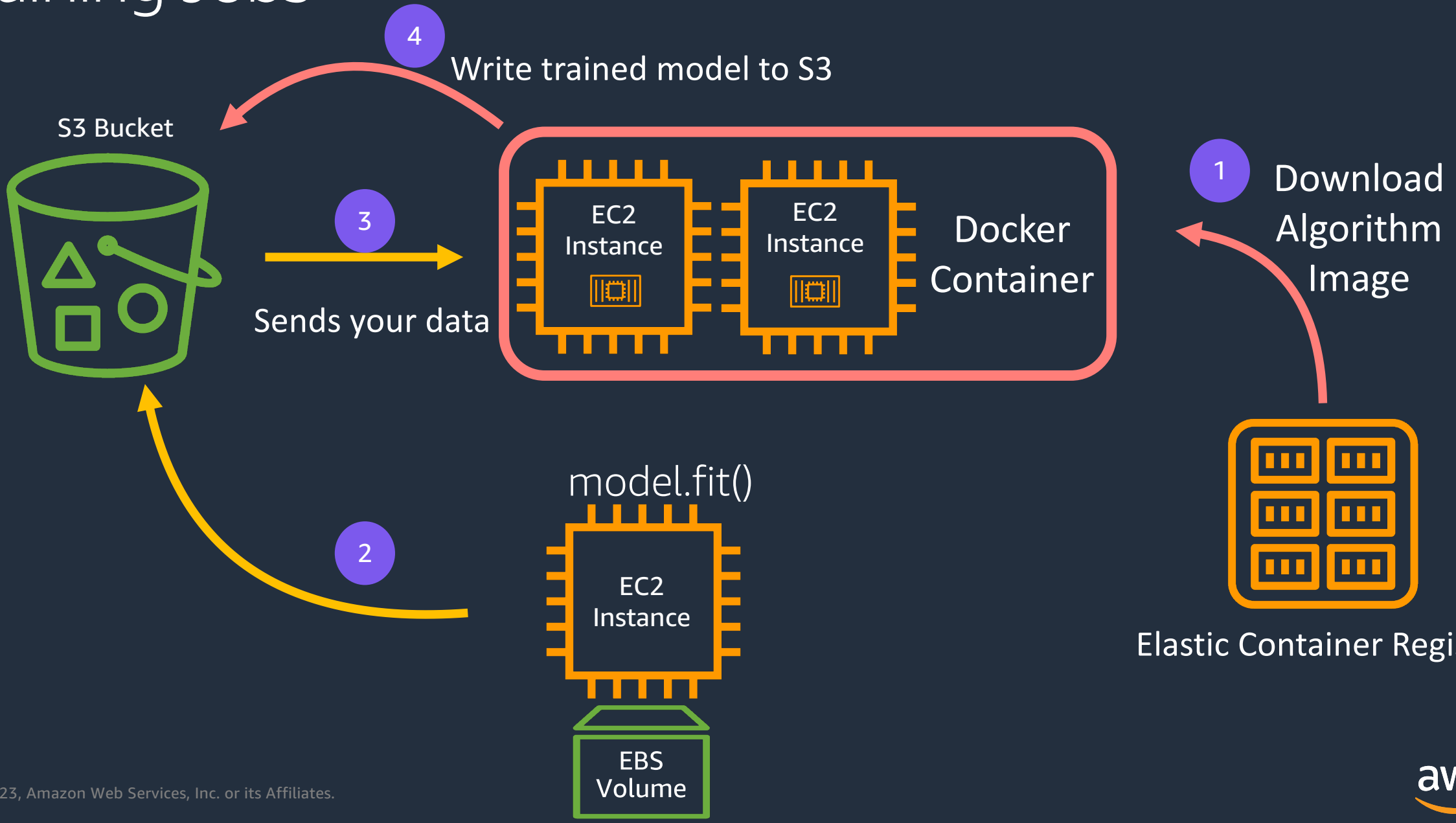
Docker containers with your own algorithms and frameworks

AWS Marketplace algorithms



3rd party algorithms and models

Training Jobs

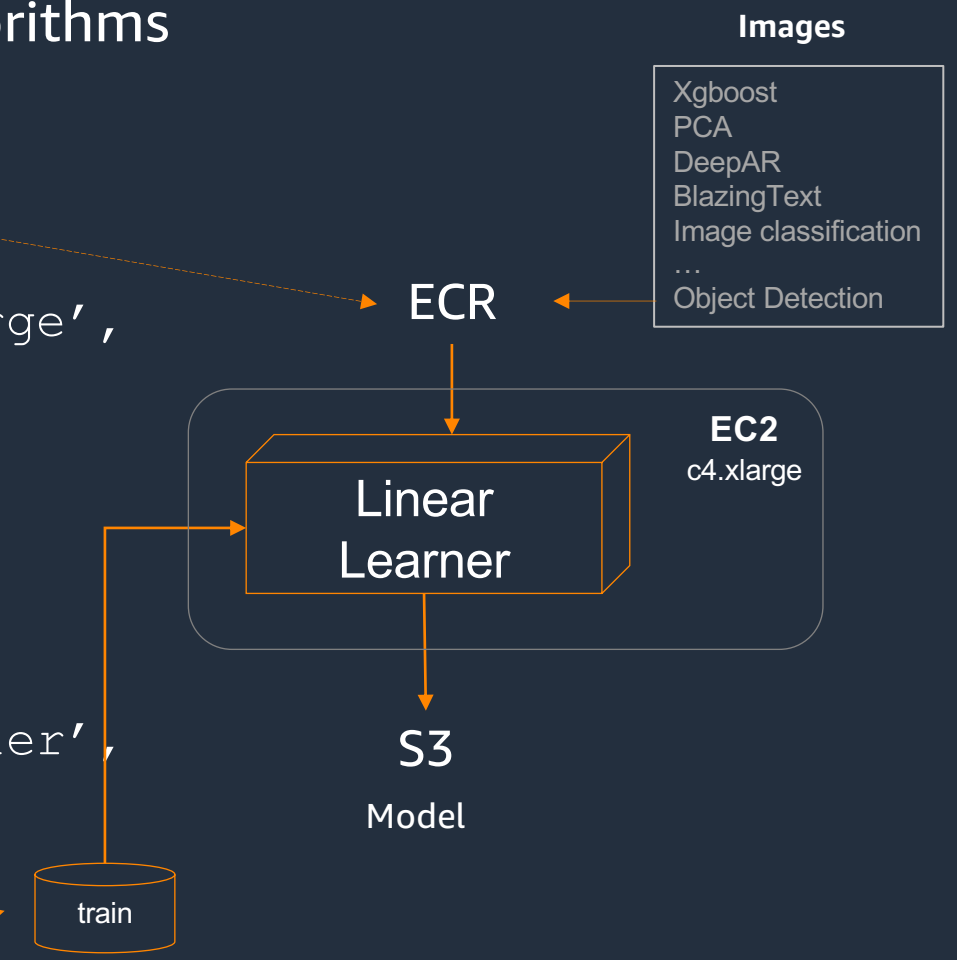


aws

Amazon SageMaker | Training

Use built-in algorithms

```
linear = Estimator('linear-learner',
                   train_instance_count=1,
                   train_instance_type='ml.c4.xlarge',
                   output_path=output_location,
                   sagemaker_session=sess)

linear.set_hyperparameters(
                   feature_dim=784,
                   predictor_type='binary_classifier',
                   mini_batch_size=200)

linear.fit({'train': s3_train_data})
```



Training – Sample Code

```
s3_input_train = sagemaker.inputs.TrainingInput(s3_data='s3://{}/{}/train'.format(bucket, prefix), content_type='csv')
s3_input_validation = sagemaker.inputs.TrainingInput(s3_data='s3://{}/{}/validation/'.format(bucket, prefix), content_type='csv')
```

```
sess = sagemaker.Session()

xgb = sagemaker.estimator.Estimator(container,
                                    role,
                                    instance_count=1,
                                    instance_type='ml.m4.xlarge',
                                    output_path='s3://{}/{}/output'.format(bucket, prefix),
                                    sagemaker_session=sess)
xgb.set_hyperparameters(max_depth=5,
                        eta=0.2,
                        gamma=4,
                        min_child_weight=6,
                        subsample=0.8,
                        silent=0,
                        objective='binary:logistic',
                        num_round=100)

xgb.fit({'train': s3_input_train, 'validation': s3_input_validation})
```

aws health

```
In [ ]: bt_model = sagemaker.estimator.Estimator(container,
                                                  role,
                                                  train_instance_count=1,
                                                  train_instance_type='ml.c4.4xlarge',
                                                  train_volume_size = 30,
                                                  train_max_run = 360000,
                                                  input_mode= 'File',
                                                  output_path=s3_output_location,
                                                  sagemaker_session=sess)
```

Number of EC2 instances

Disk space

Type of EC2 instances

aws

Algorithm Container

```
In [ ]: bt_model = sagemaker.estimator.Estimator container,
                  role,
                  train_instance_count=1,
                  train_instance_type='ml.c4.4xlarge',
                  train_volume_size = 30,
                  train_max_run = 360000,
                  input_mode= 'File',
                  output_path=s3_output_location,
                  sagemaker_session=sess)
```

SageMaker Estimator

Execution Role

```
In [ ]: bt_model.fit(inputs=data_channels, logs=True)
```

Cluster comes online

Logs to CloudWatch

Monitor via console or notification stream

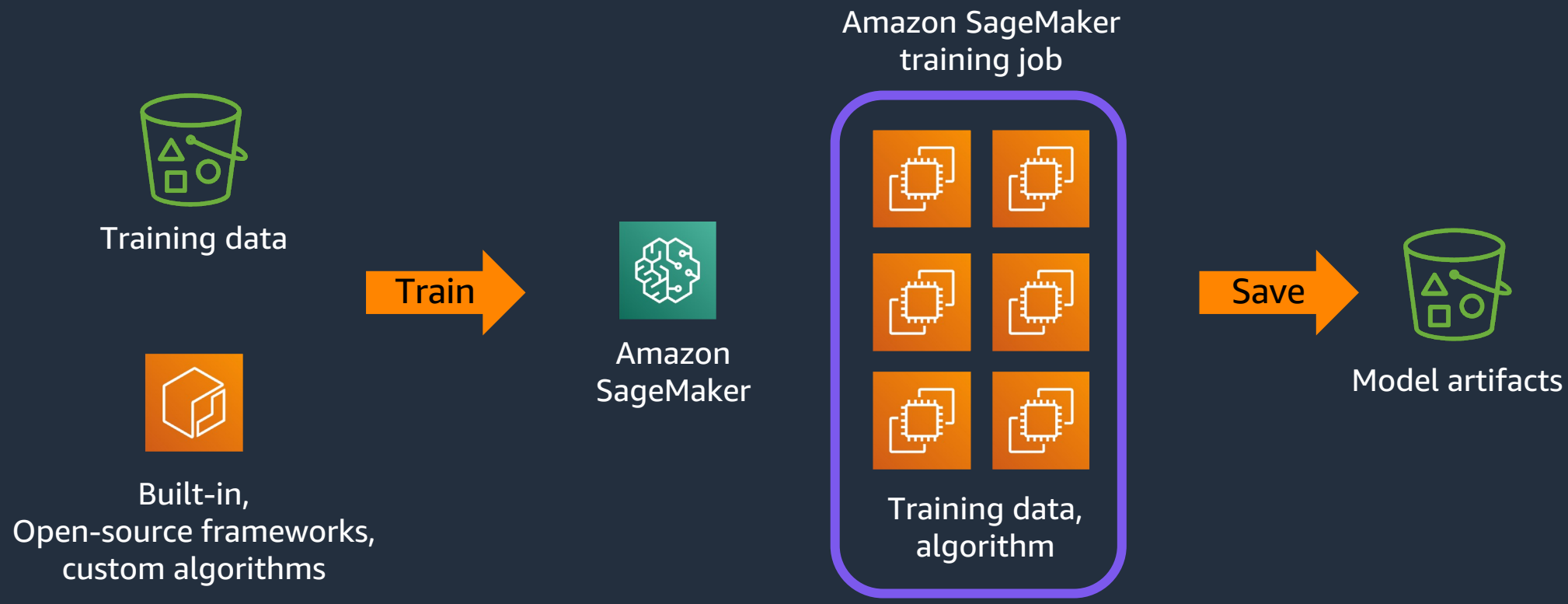
aws

Loading training data from Amazon S3

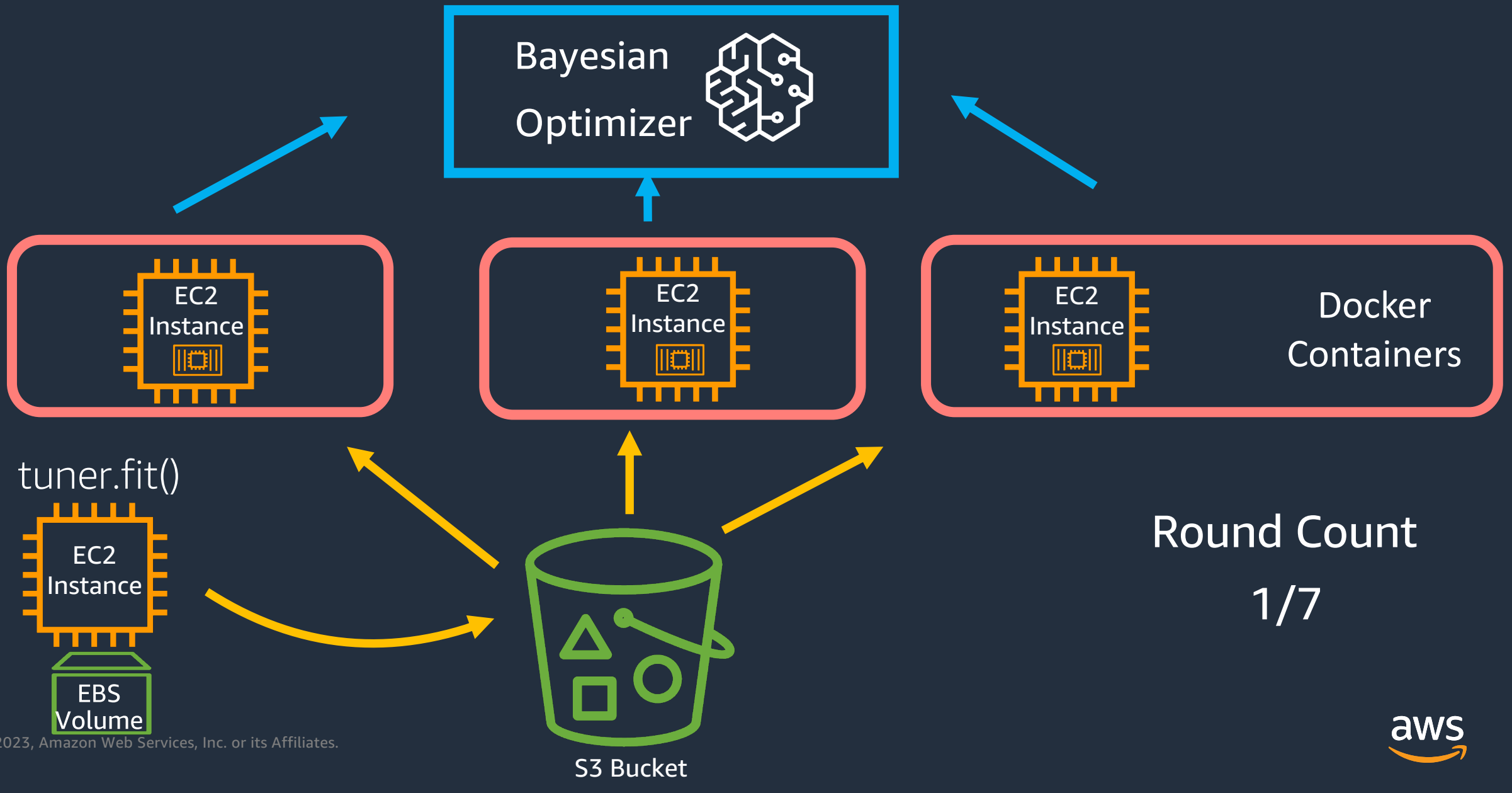
- Two modes: File Mode and Pipe Mode
  - *input_mode* parameter in *sagemaker.estimator.estimator*
- File Mode copies the dataset to training instances
  - You need to provision enough storage
  - *S3DataSource* object
  - *S3DataDistributionType*: *FullyReplicated* | *ShardedByS3Key*
  - Different data formats are supported: CSV, protobuf, JSON, libsvm (check algo docs!)
- Pipe Mode streams the dataset to training instances
  - This allows you to process infinitely large datasets
  - Training starts faster
  - This mode is supported by some built-in algos as well as TensorFlow
  - Your dataset must be in recordIO-encoded protobuf format

Fully managed training

Billable by the second, 100% utilization



Hyperparameter Tuning Jobs



aws

1. Pick hyperparameters and ranges

```
hyperparameter_ranges = {'eta': ContinuousParameter(0, 1),
                         'min_child_weight': ContinuousParameter(1, 10),
                         'alpha': ContinuousParameter(0, 2),
                         'max_depth': IntegerParameter(1, 10)}
```

2. Pick objective metric

```
objective_metric_name = 'validation:auc'
```

3. Pick job parameters

```
tuner = HyperparameterTuner(xgb,
                            objective_metric_name,
                            hyperparameter_ranges,
                            max_jobs=20,
                            max_parallel_jobs=3)
```


aws

Deployment flexibility

SageMaker Real Time Inference

- Creates a long-running microservice as an endpoint
- Effective for low latency for small data batches
- Can be deployed as a Multi-Model Endpoint

SageMaker Batch Transform

- Bulk data transformation
- Periodic arrival of big chunks of data
- Configurable inference compute infrastructure

SageMaker Serverless Inference

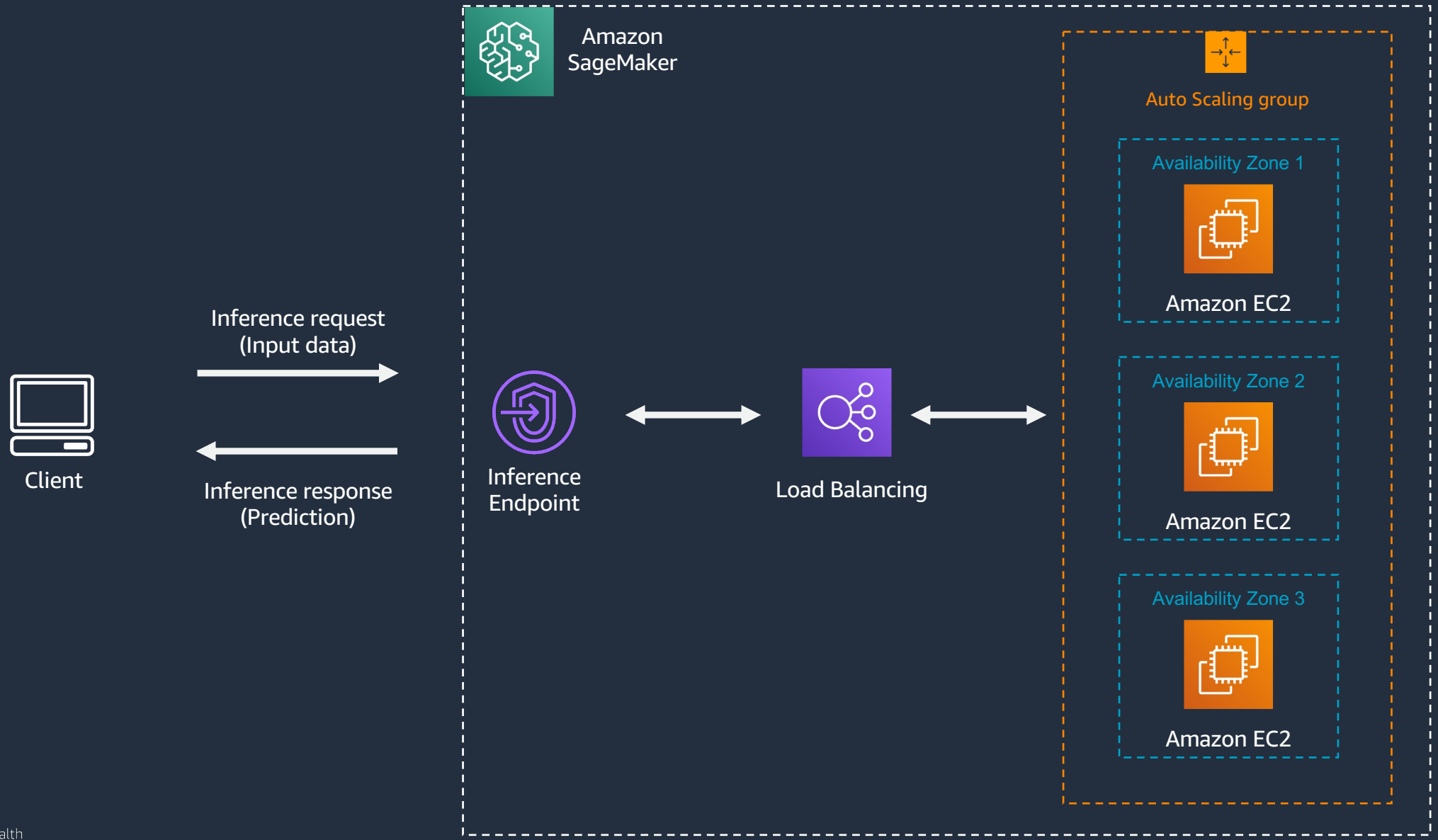
- Deploy without managing infrastructure
- Ideal for intermittent or unpredictable traffic

SageMaker Asynchronous Inference

- Queues incoming requests
- Ideal for for large payloads and long processing times
- Can scale down to zero instances, saving costs

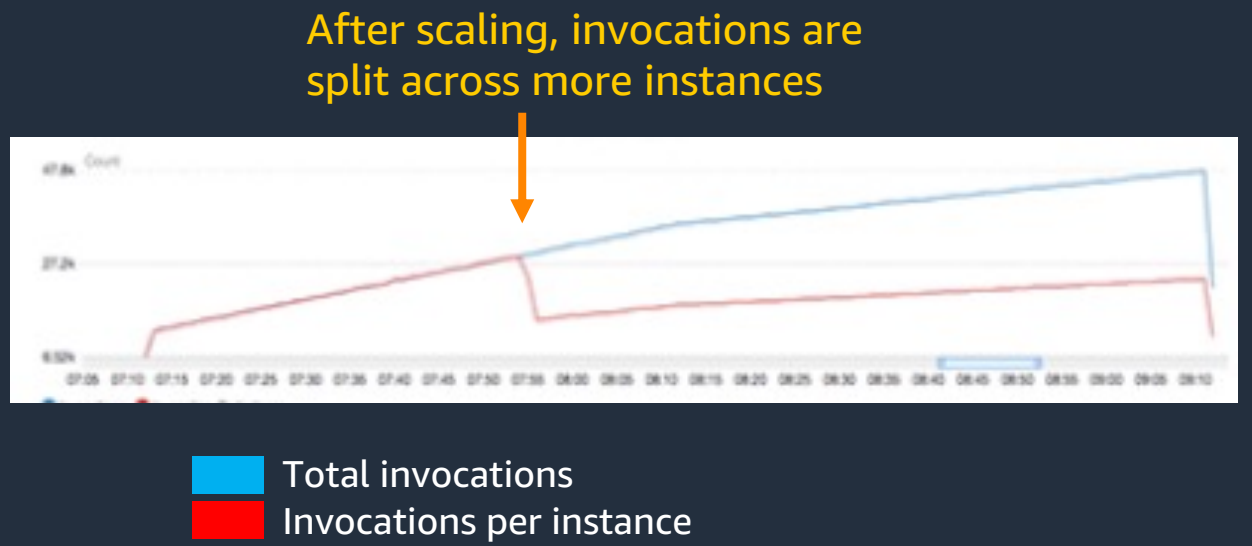
aws health

Amazon SageMaker Real-time endpoint



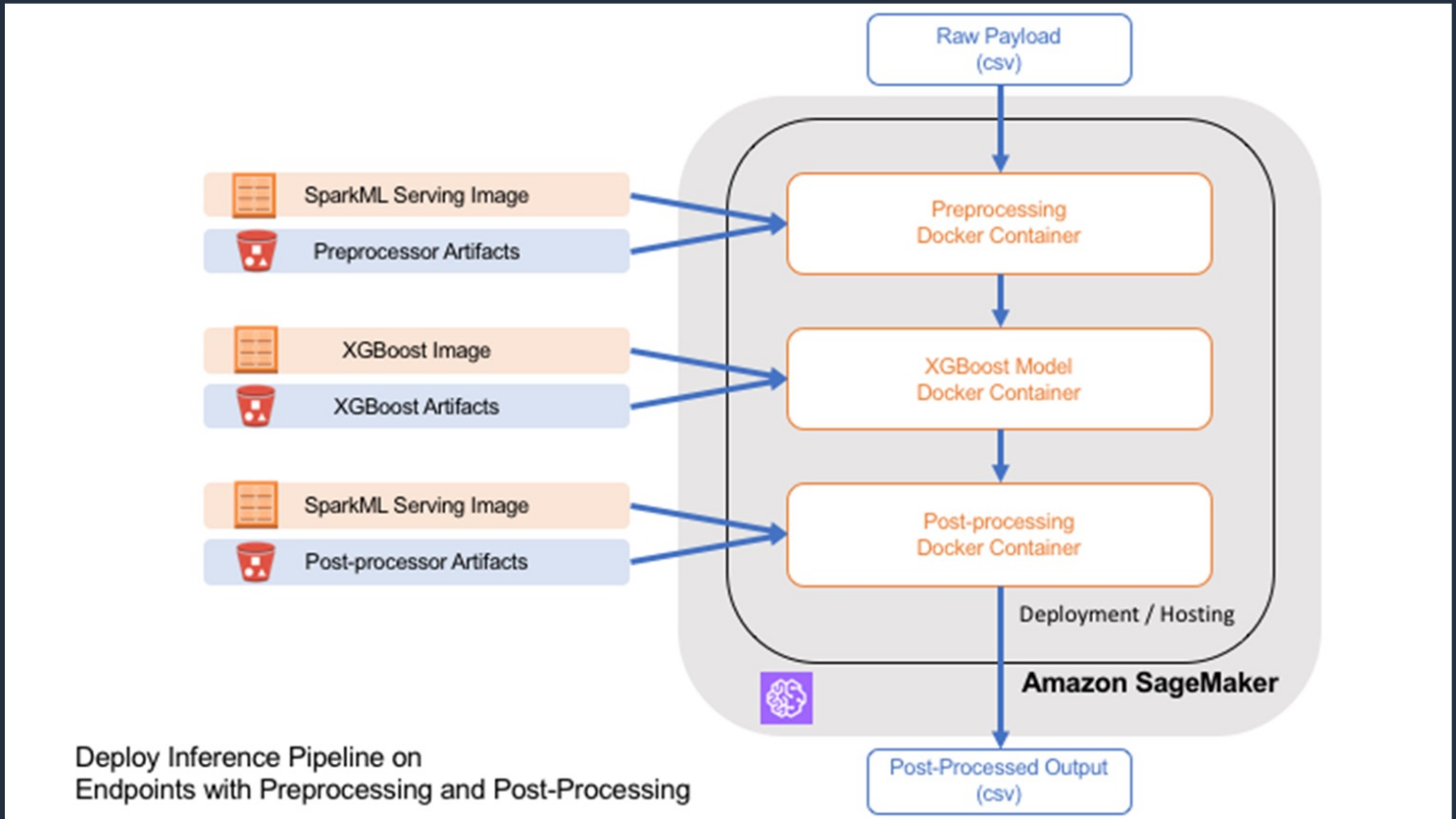
Autoscaling for real-time inference endpoints

- Provision steady state capacity
- Set minimum, maximum instance counts, scaling criteria
- SageMaker auto-scales up to meet demand, and scales back down
- Significant cost savings



Inference Pipelines for sequential execution of models

Execute data processing on inference requests. Maintain single copy of data processing code for training and inference.

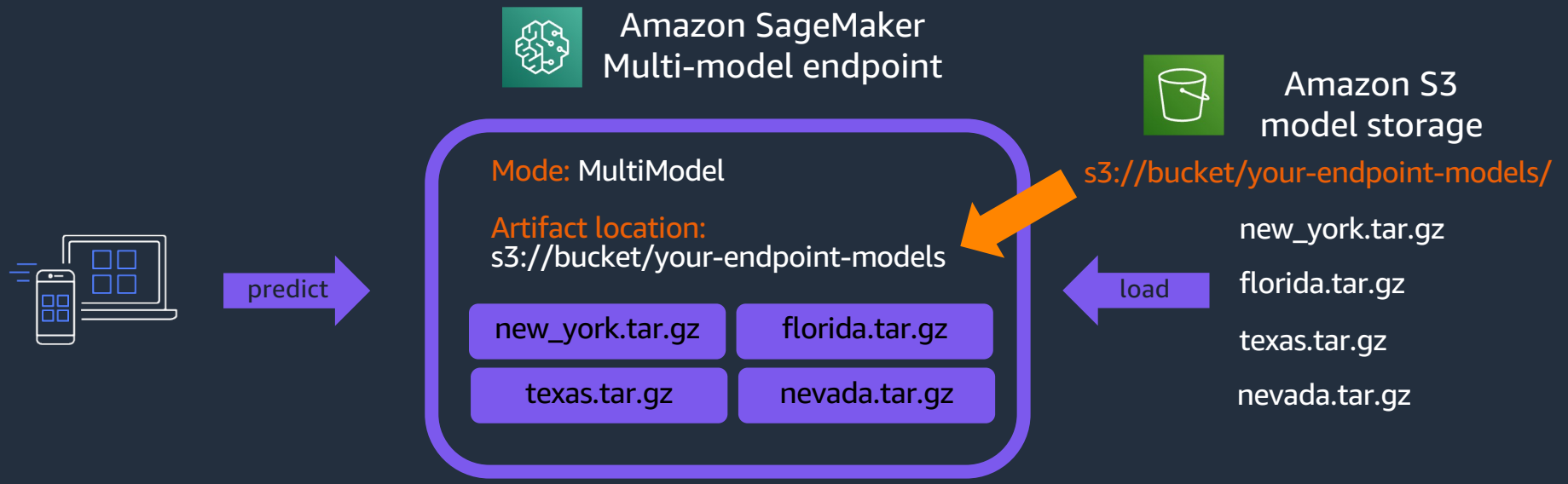


Built-in containers – Scikit-Learn and Apache Spark MLLib

Add Up to 5 containers; execute sequentially

Containers co-located on instances for low latency

Multi-Model Endpoints

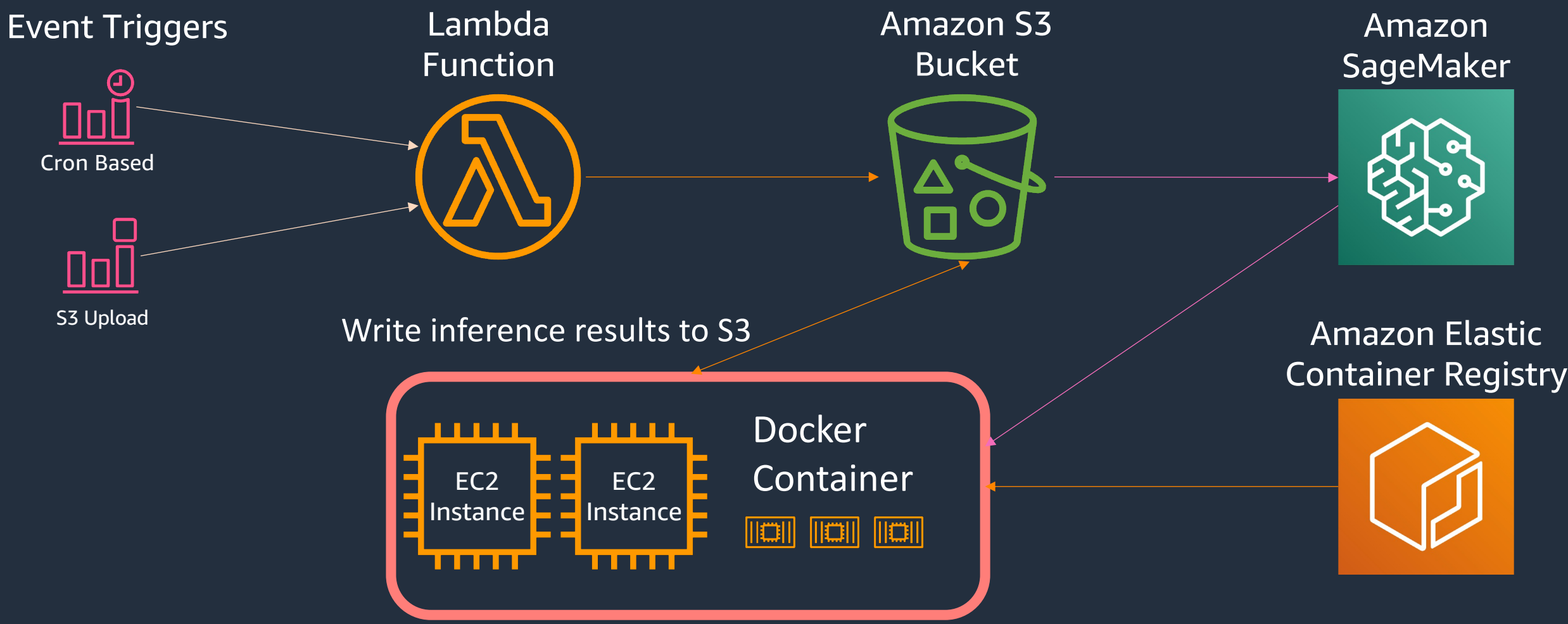


Key Capabilities

- Deploy tens to tens of thousands of models; target model for each inference request
- Two modes for model caching: Caching Enabled, Caching Disabled
- Support for built in algorithms, SageMaker frameworks, and custom models
- Support for inference pipelines and condition keys to restrict access to models

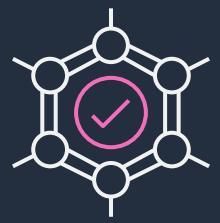
aws health

Batch Transform Common Design Pattern



aws

Serverless Deployment - Fully managed offering



Managed infrastructure

Security

Monitoring

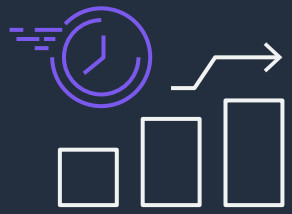
Logging

Built in availability and fault tolerance

Serverless

No need to select instance types or provision capacity

Choose memory options based on inference processing needs



Automatically scale out, in, and down to 0

No need to set scaling policies

Pro Tips

- Turn your endpoints off when not in use with Lambda
- Use inference pipelines for pre and post processing
- Bring more models in your own Docker container
- Consider the size of data hitting a single endpoint
- You can train your model elsewhere and host in SageMaker

aws

Lab: Train, Tune and Deploy model using SageMaker Built-in Algorithm

Prerequisites + “Lab 2. Train, Tune and Deploy XGBoost”

Q&A and chat are open!

Until 14:30

Bring Your Own Custom Models

Model Options

In this session: Bring Your Own Script/Container

Training code

AWS Marketplace for Machine Learning

Amazon SageMaker AutoPilot

- XGBoost - Gradient Boosted Trees
- Matrix Factorization
- Regression
- Principal Component Analysis
- K-Means Clustering
- And More!

Built-in Algorithms (17)
No ML coding required

Bring Your Own Script
Amazon SageMaker builds the container
Open source containers

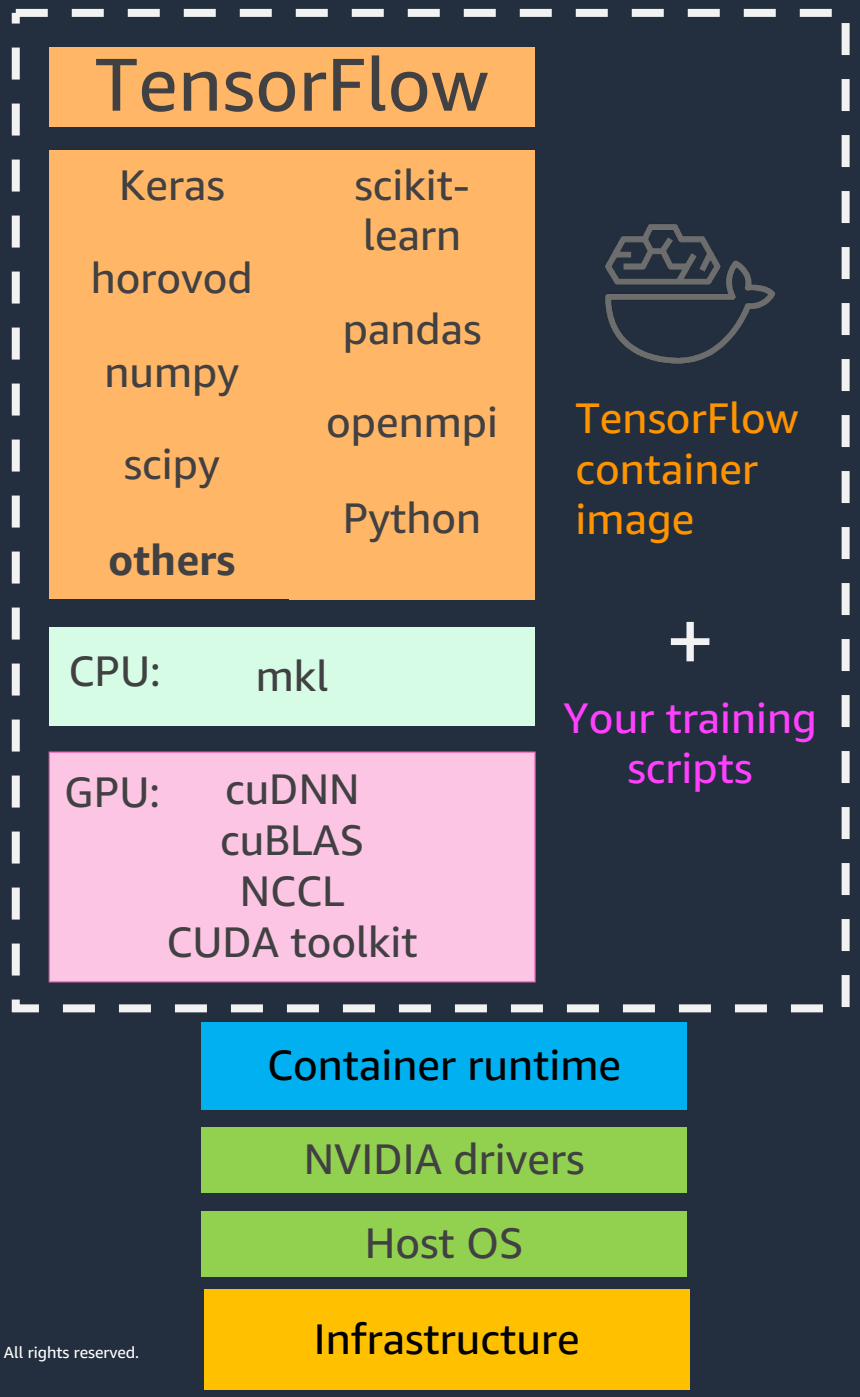
Bring Your Own Container
Full control, you build the container
R, C++, etc

Fully Managed, Distributed, Auto-Scaled, Secured

aws health

Containers for Machine Learning

Containers for machine learning



ML environments that are:

- Lightweight
- Portable
- Scalable
- Consistent

Packages:

- Training code
- Dependencies
- Configurations

TensorFlow

Keras
scikit-learn
horovod
pandas
numpy
openmpi
scipy
Python
others

CPU: mkl

GPU: cuDNN
cuBLAS
NCCL
CUDA toolkit

TensorFlow container image

+

Your training scripts

push

pull

Amazon ECR

Container registry

TensorFlow

Keras
scikit-learn
horovod
pandas
numpy
openmpi
scipy
Python
others

CPU: mkl

GPU: cuDNN
cuBLAS
NCCL
CUDA toolkit

TensorFlow container image

+

Your training scripts

Container runtime

NVIDIA drivers
Host OS

Development system

Container runtime

NVIDIA drivers
Host OS

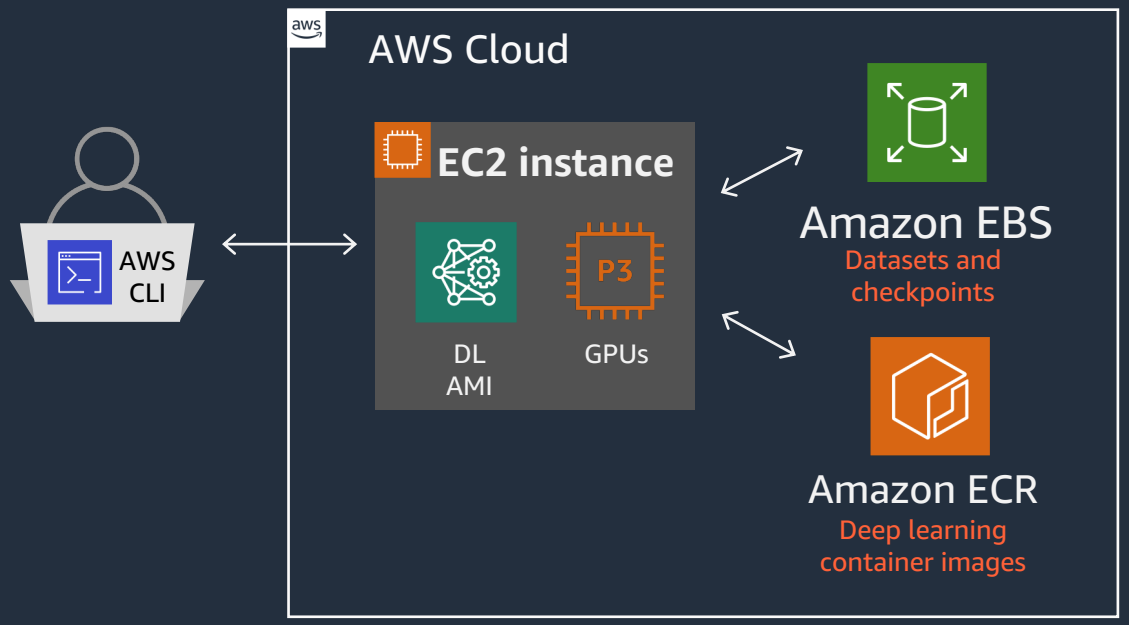
Training cluster

AWS Deep Learning Containers

- Prepackaged Docker container images fully configured and validated
- Optimized for performance with latest NVIDIA driver, CUDA libraries, and Intel libraries
- Consistent and reproducible deployment and lightweight
- Optimized for distributed machine learning
- Runs on Amazon ECS, Amazon EKS and Amazon SageMaker

aws health

Machine Learning Setups on AWS Today



- Code and dependencies
- Collaborative development
- Performance optimizations
- Scaling
- Infrastructure management

ML Infrastructure and Cluster Management

ML services
Fully managed service that covers the entire machine learning workflow

Amazon SageMaker


Jupyter notebook instances


High-performance algorithms


Large-scale training


Optimization


One-click deployment


Fully managed with auto scaling

Management
Deployment, scheduling, scaling, and management of containerized applications


Amazon Elastic Container Service (Amazon ECS)


Amazon Elastic Kubernetes Service (Amazon EKS)

Compute
Where the containers run


Amazon EC2

Image registry
Container image repository


Amazon Elastic Container Registry (Amazon ECR)

aws health

3 Ways to Train Using Amazon SageMaker

Use built-in algorithms

(Bring Your Own Data)

- K-Means Clustering
- Principal Component Analysis
- Neural Topic Modelling
- Factorization Machines
- Linear Learner (Regression)
- BlazingText
- Reinforcement learning
- XGBoost
- Topic Modeling (LDA)
- Image Classification
- Seq2Seq
- Linear Learner (Classification)
- DeepAR Forecasting

Use deep learning frameworks

(Bring Your Own Data)

(Bring your own training script)

TensorFlow

Keras

mxnet

GLUON

HOROVOD

PYTORCH

Chainer

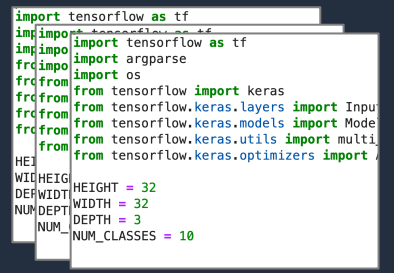
ONNX

Use custom containers

(Bring your own data)

(Bring your own container)

Custom container

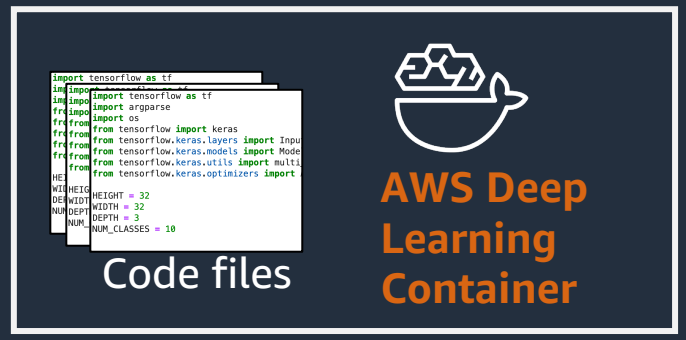


Code files

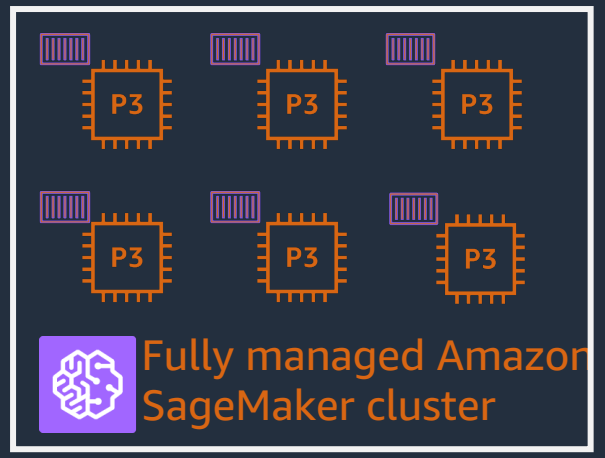
aws health

Amazon SageMaker
Bring your own Script

High Level Workflow

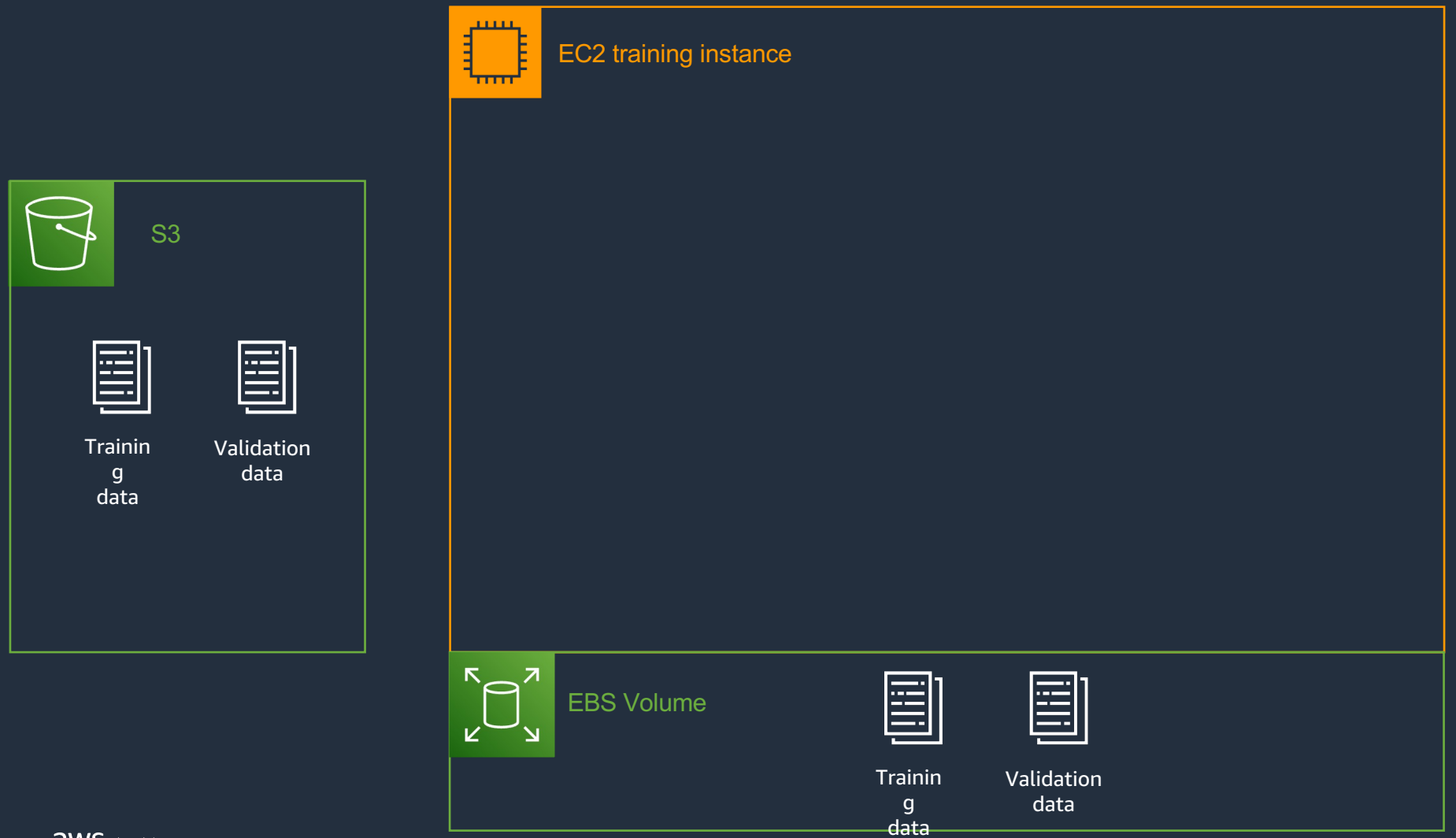


```
from sagemaker.tensorflow import TensorFlow

cifar10_estimator = TensorFlow(entry_point='source/cifar10.py',
                               role=role,
                               framework_version='1.14',
                               train_instance_count=1,
                               train_instance_type='ml.p3.xlarge')
```

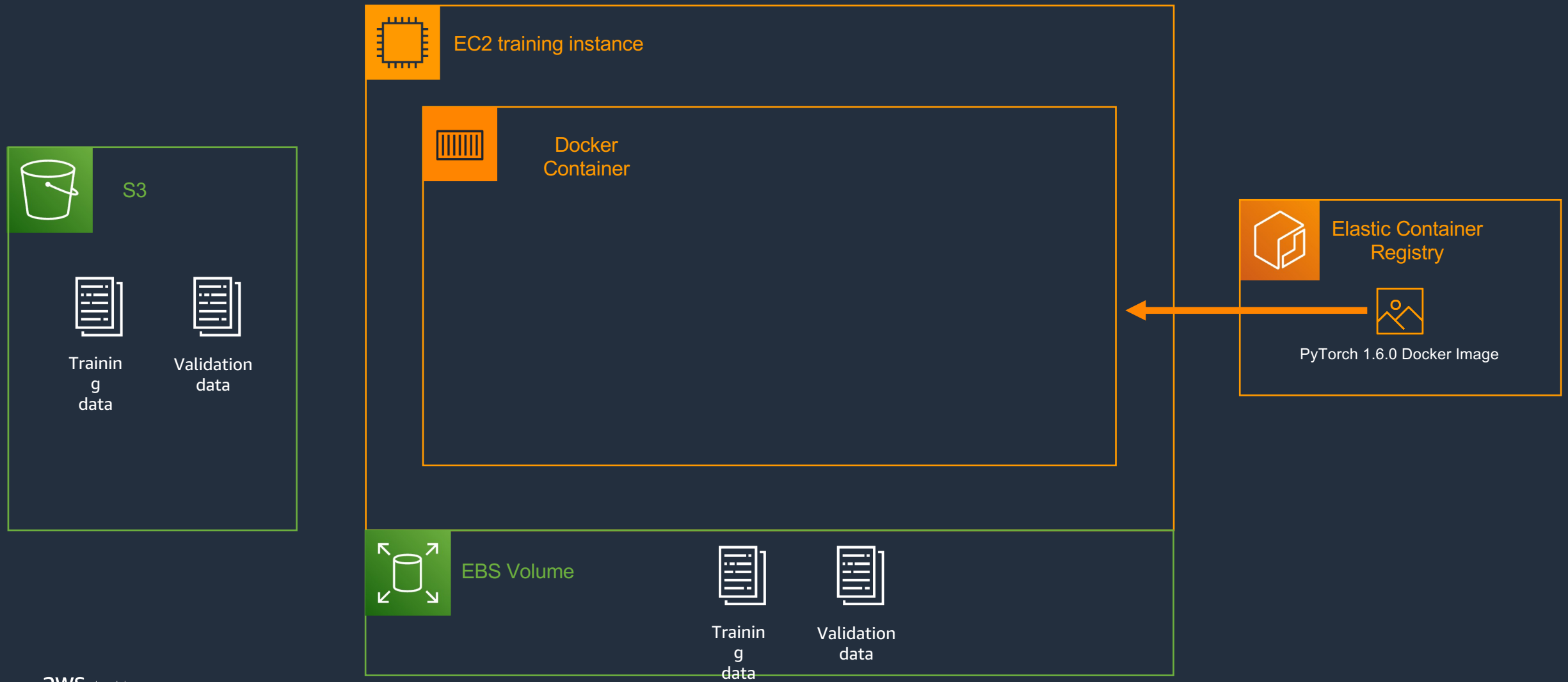


aws health

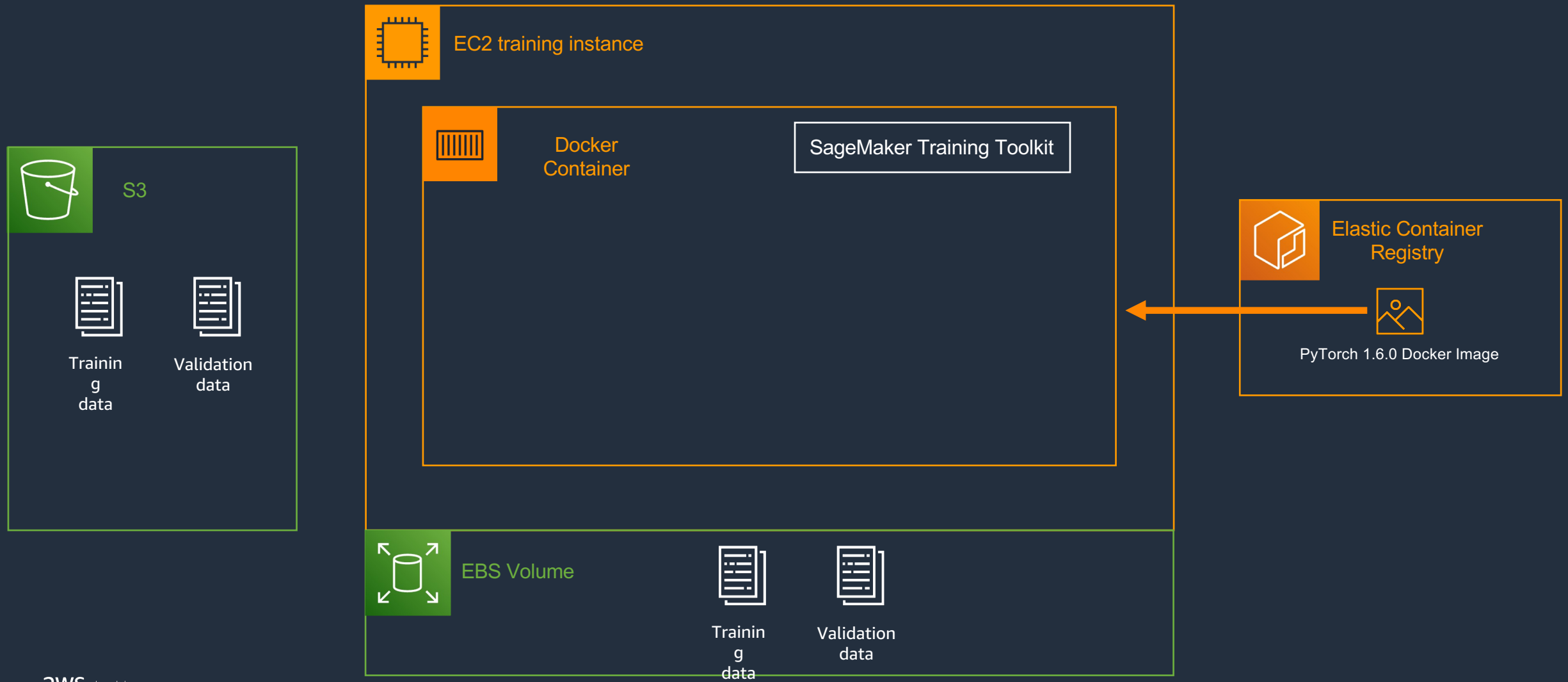
Copy the data from S3 to EBS volume of the EC2 instance



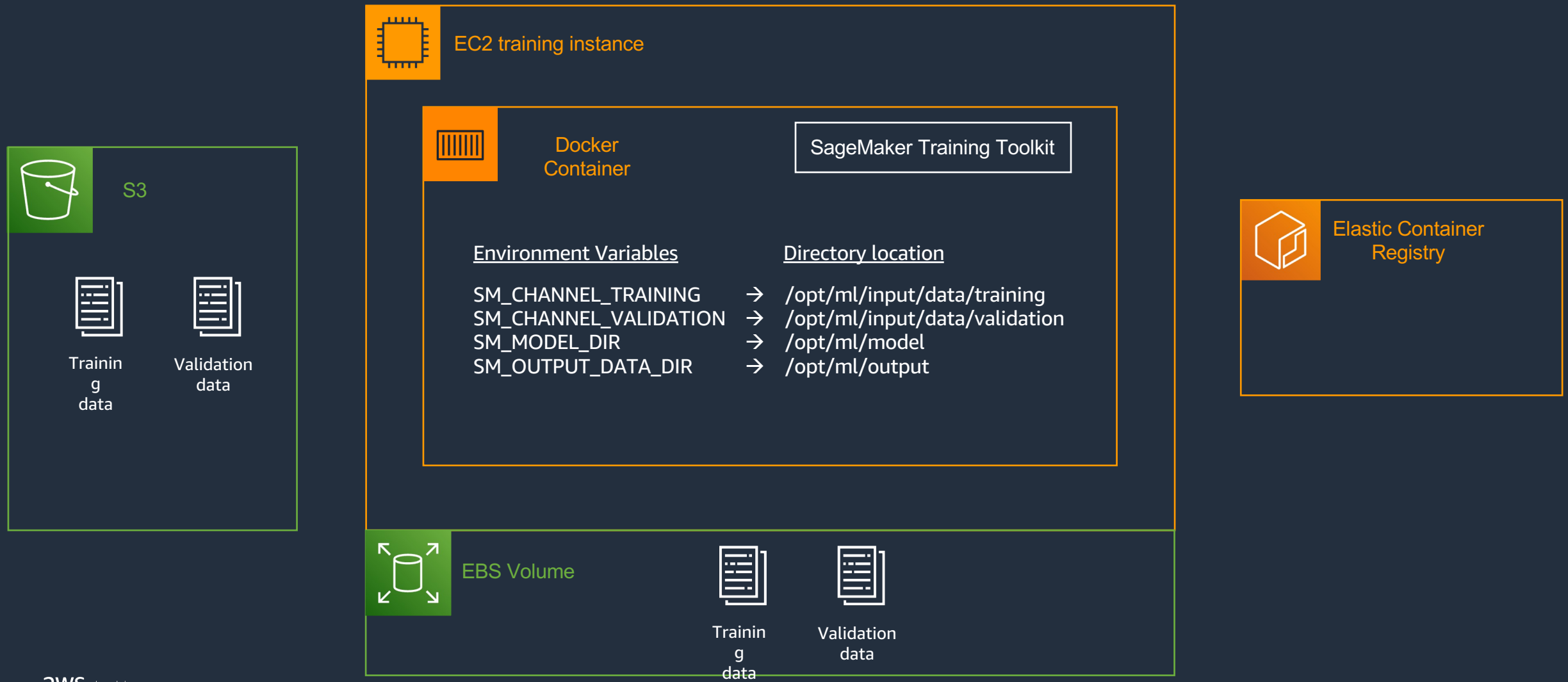
Docker Container



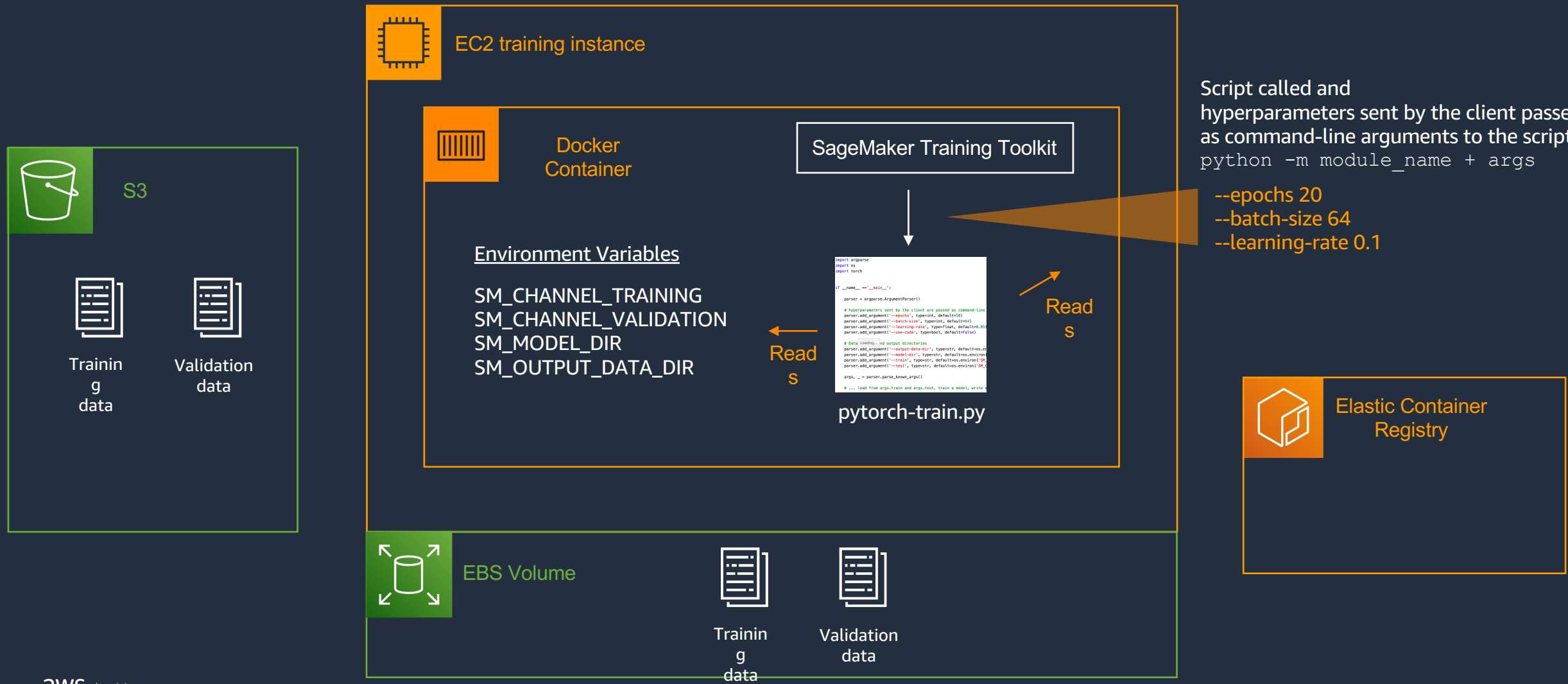
SageMaker Training Toolkit



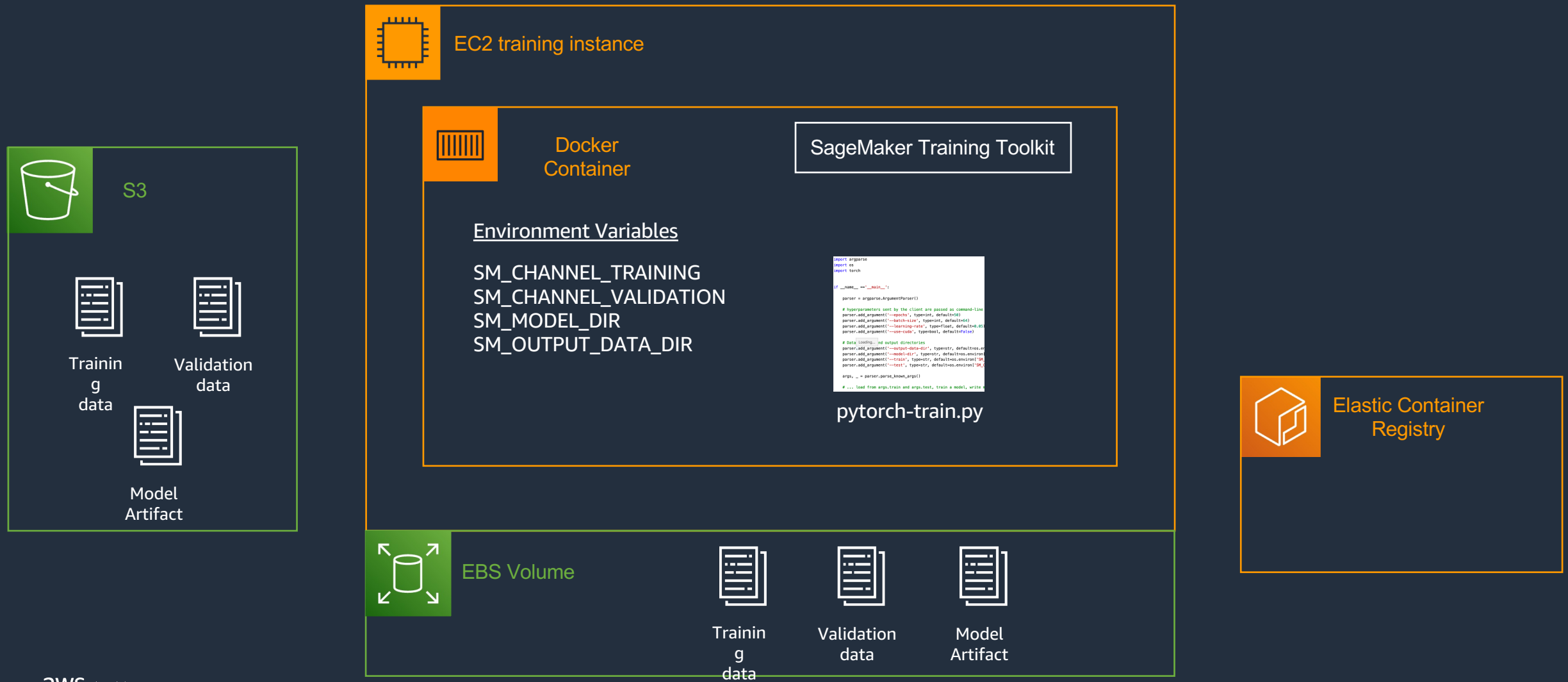
Environment variables and directory location



Your script is called with arguments present in the Estimator



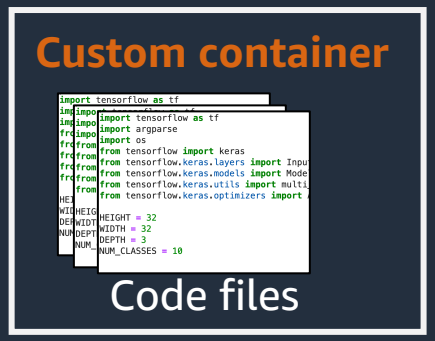
Copy of the Model Artifact to S3 after training



Amazon SageMaker
Bring your own Container

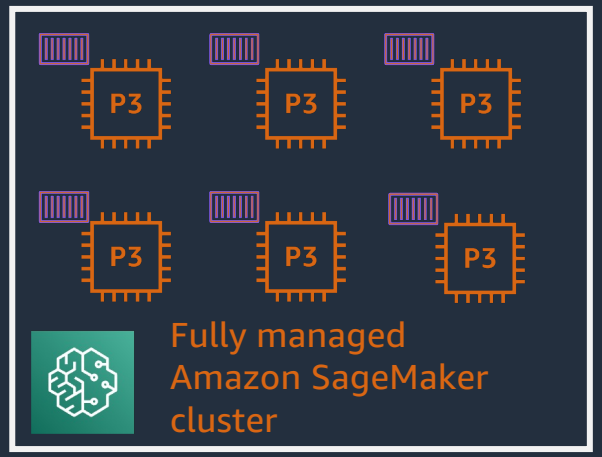
High Level Workflow

Docker build



Container registry

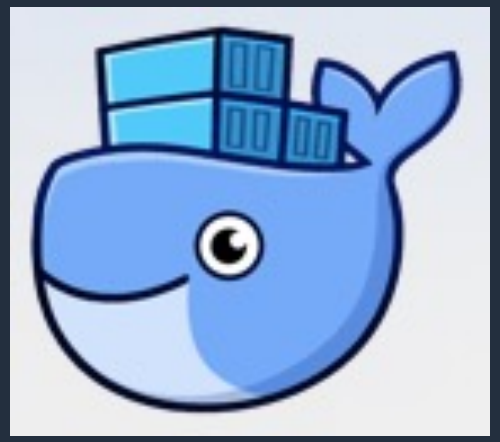
```
ic = sagemaker.estimator.Estimator(training_image,
                                   role,
                                   train_instance_count=1,
                                   train_instance_type='ml.p3.xlarge',
                                   train_volume_size = 50,
                                   train_max_run = 360000,
                                   input_mode= 'File',
                                   output_path=s3_output_location,
                                   sagemaker_session=sess)
```



Amazon Simple Storage Service

aws health

Bring Your Own Docker File



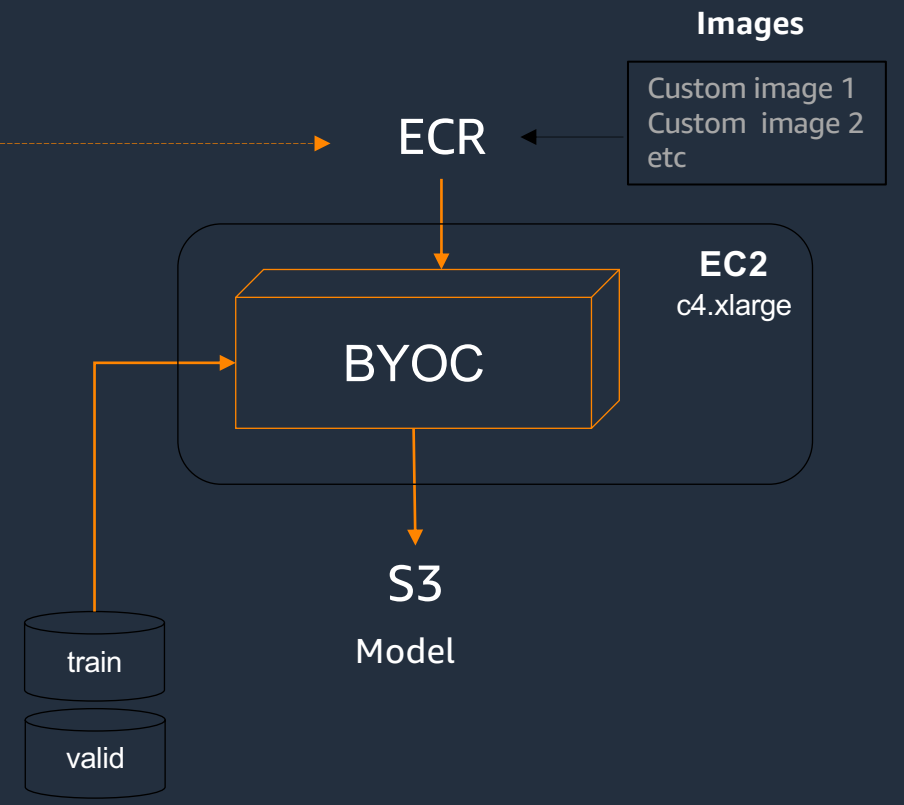
Customer Managed

1. Write your model however you please
2. Point to your model within your Docker file
3. Register your container on ECR
4. Point to your container's address in ECR
5. Don't forget to implement a serve() function!

Amazon SageMaker | Training

Bring your own container

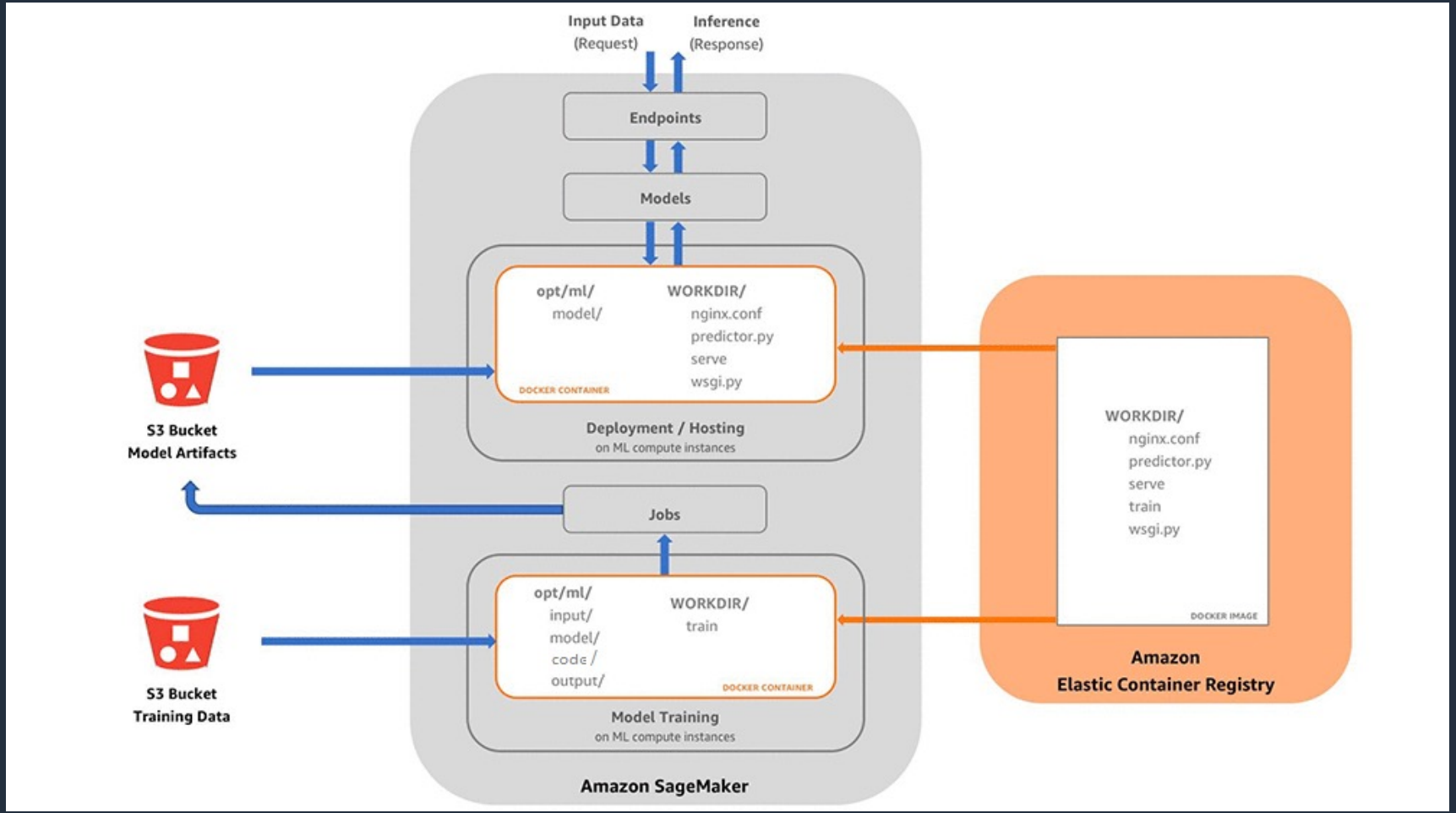
```
estimator = Estimator(image='custom_container',
                      train_instance_type='ml.c4.xlarge',
                      train_instance_count=1,
                      hyperparameters=hyperparameters,
                      py_version='py3')

estimator.fit({'training': inputs})
```



Bring Your Own Container

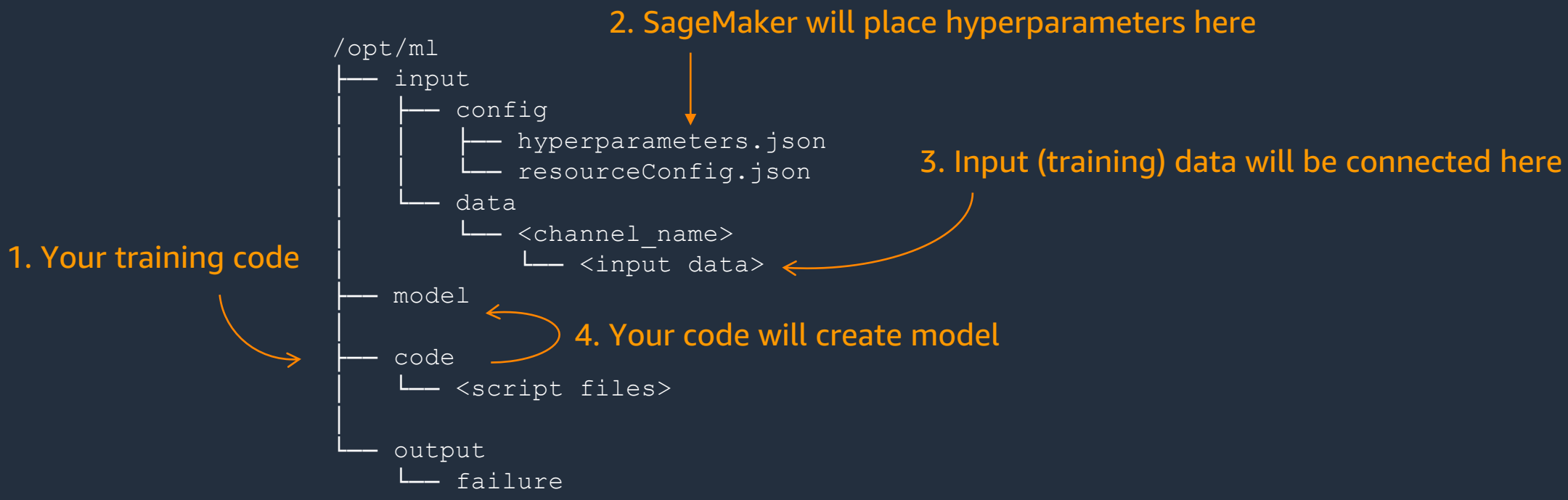
Custom Docker image:

- Training image
- Inference (serving) image



Amazon SageMaker | Training

Bring your own container



5. Final model from /opt/ml/model will be saved to S3 at the end of training session, before the container is destroyed.

aws

ML Marketplace

Algorithms

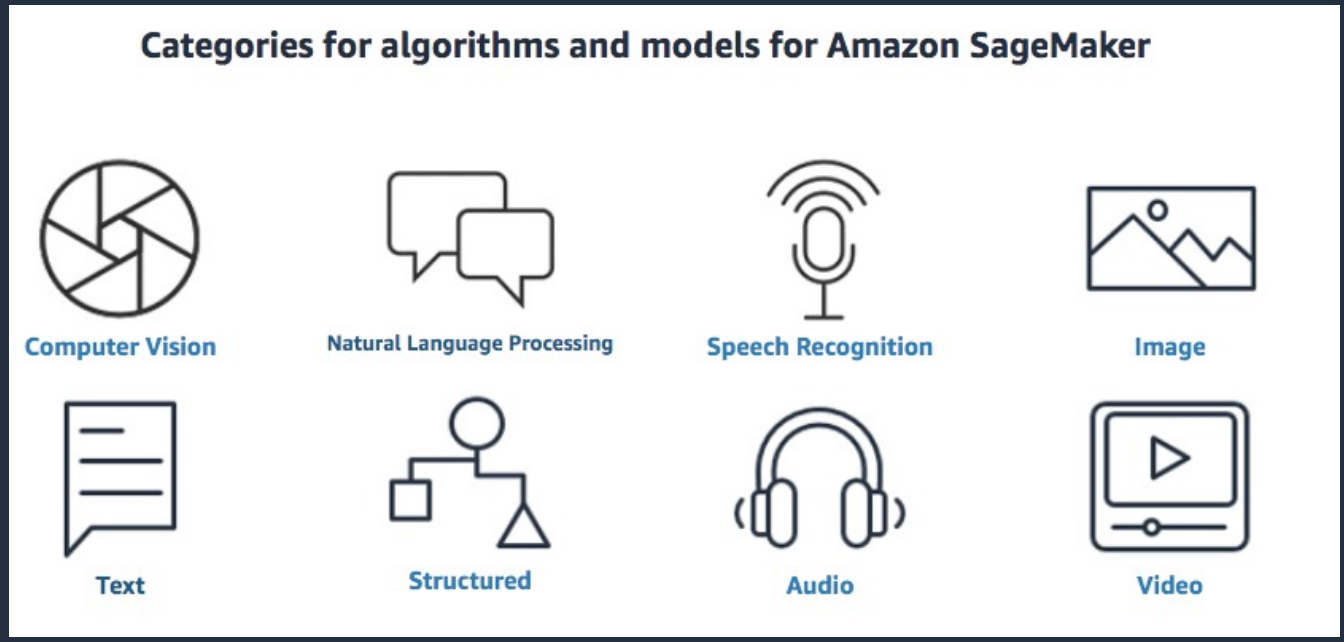
- You can train on your data

Models

- Use a pretrained model artifact

- Subscription model
- Free tier!

230+ solutions!



Model parallelism

Automated and efficient model partitioning

Minimal code change

Optimized for training large models

Data parallelism

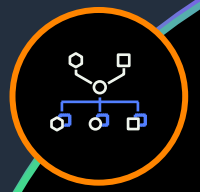
Reduced training time

Optimized for AWS network

Support for popular ML framework APIs

Distributed training on Amazon SageMaker

https://aws.amazon.com/sagemaker/distributed-training/

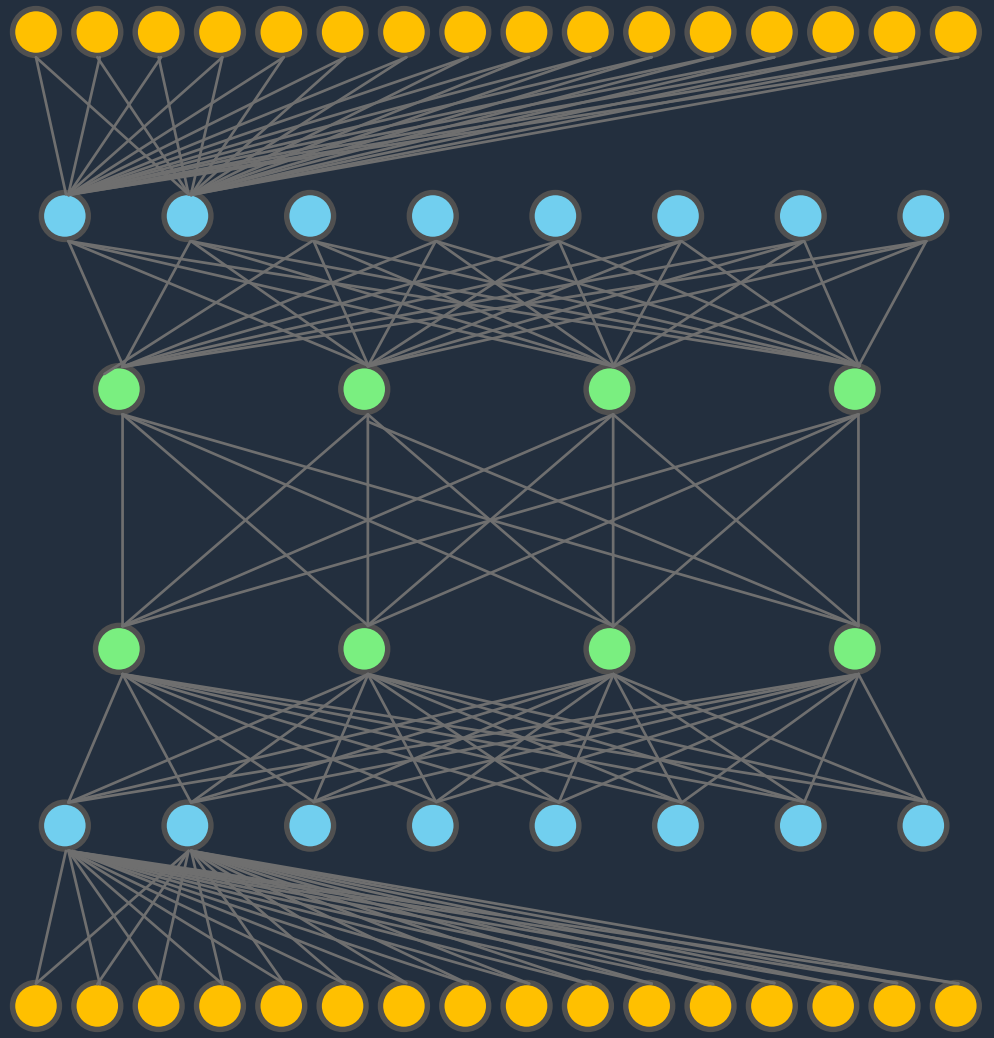





aws health

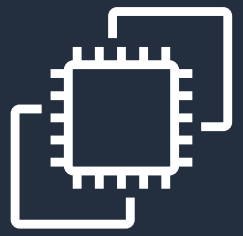
DataParallel in SageMaker

- Library for distributed training of deep learning models in TensorFlow and PyTorch
- Accelerates training for network-bound workloads
- Built and optimized for AWS network topology and hardware
- 20%–40% faster and cheaper; best performance on AWS



aws health

Model parallelism on SageMaker



Efficient pipelined training

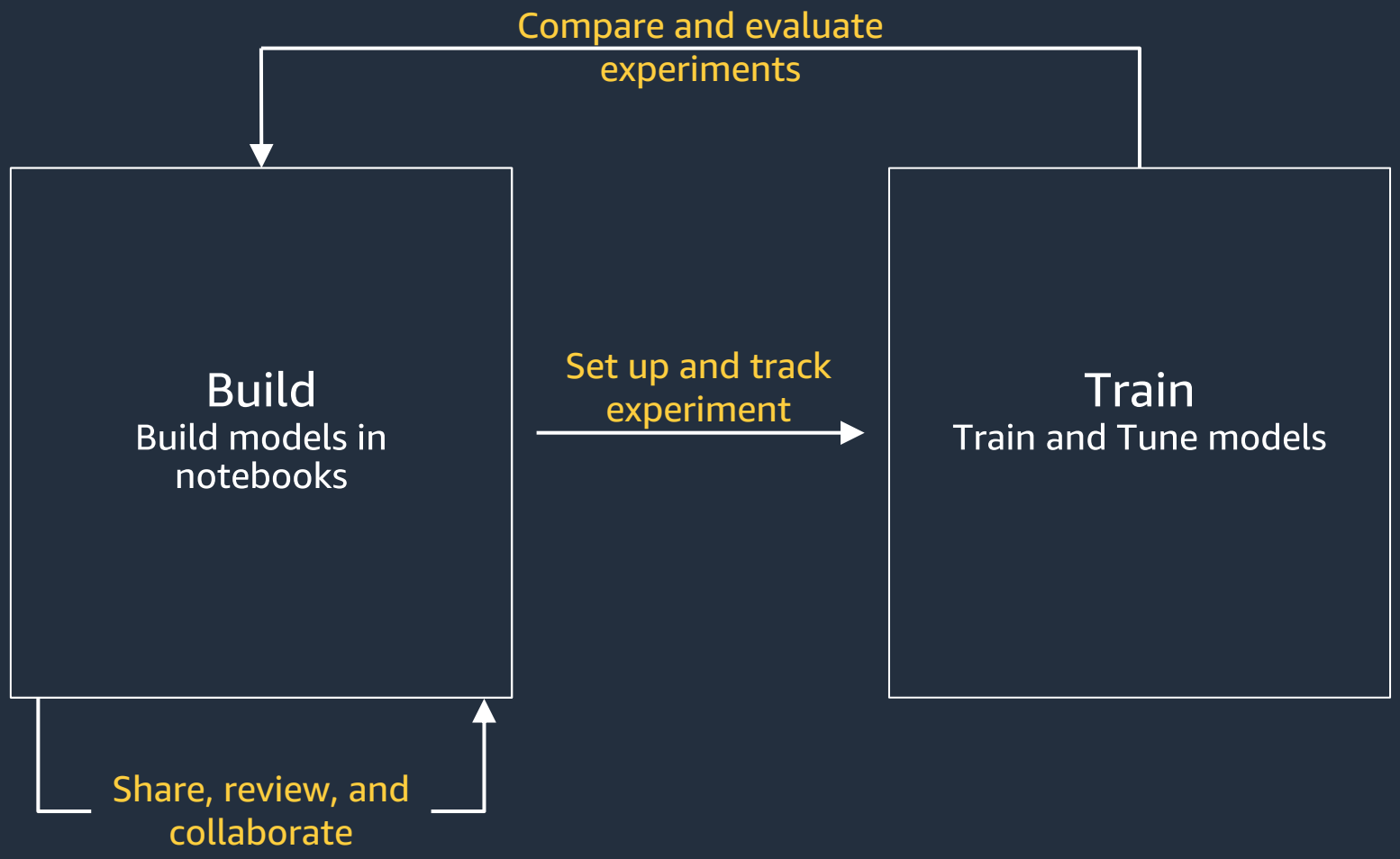
Automated model partitioning

Managed SageMaker training

Tight framework integration

aws health

Manage Experiments



aws health

Amazon SageMaker Experiments

Organize, track, and compare training experiments

Tracking at scale

Track parameters and metrics across experiments and users

Custom organization

Organize experiments by teams, goals, and hypotheses

Visualization

Easily visualize experiments and compare

Metrics and logging

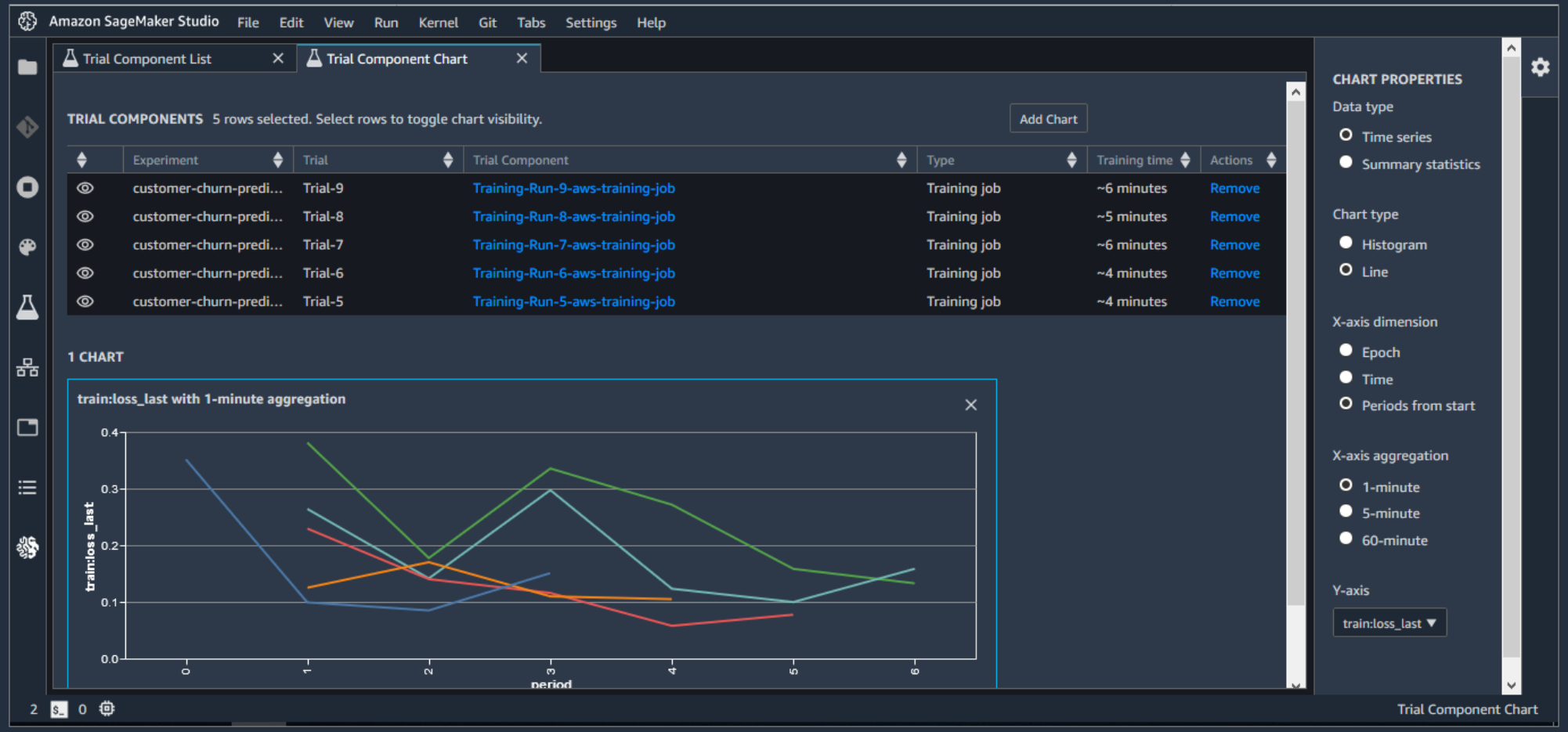
Log custom metrics using the Python SDK and APIs

Fast iteration

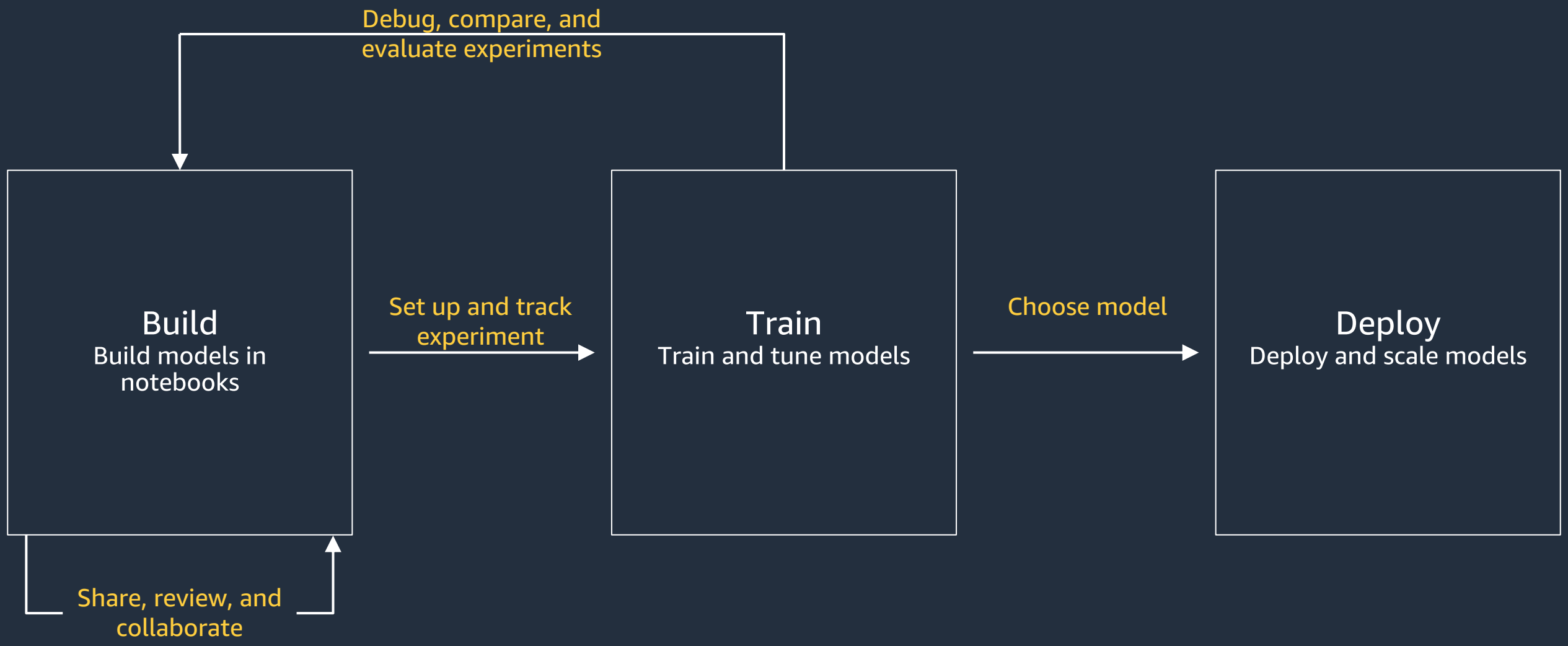
Quickly go back and forth, and maintain high-quality

aws health

Use Amazon SageMaker Experiments to track and manage thousands of experiments



Deploy models



aws health

Pro Tips

	Script Mode	BYO Docker	ML Marketplace
Pros	Quickly train your own model	Most flexible	Huge variety: easy to add value quickly
Cons	Limited to managed options	More time consuming	Less insight into solution

SageMaker reads from /opt/ml

Find an example to modify

Takeaways

Code and dependencies

Collaborative development

Performance optimizations

Scaling

Infrastructure management

- Containers let you build lightweight, portable and consistent ML environments
- AWS DL containers include frameworks optimized by experts to deliver the best performance on CPUs and GPUs
- Leverage Amazon ECR along with git for collaborative development
- Leverage Amazon SageMaker to manage large-scale ML workloads

aws health

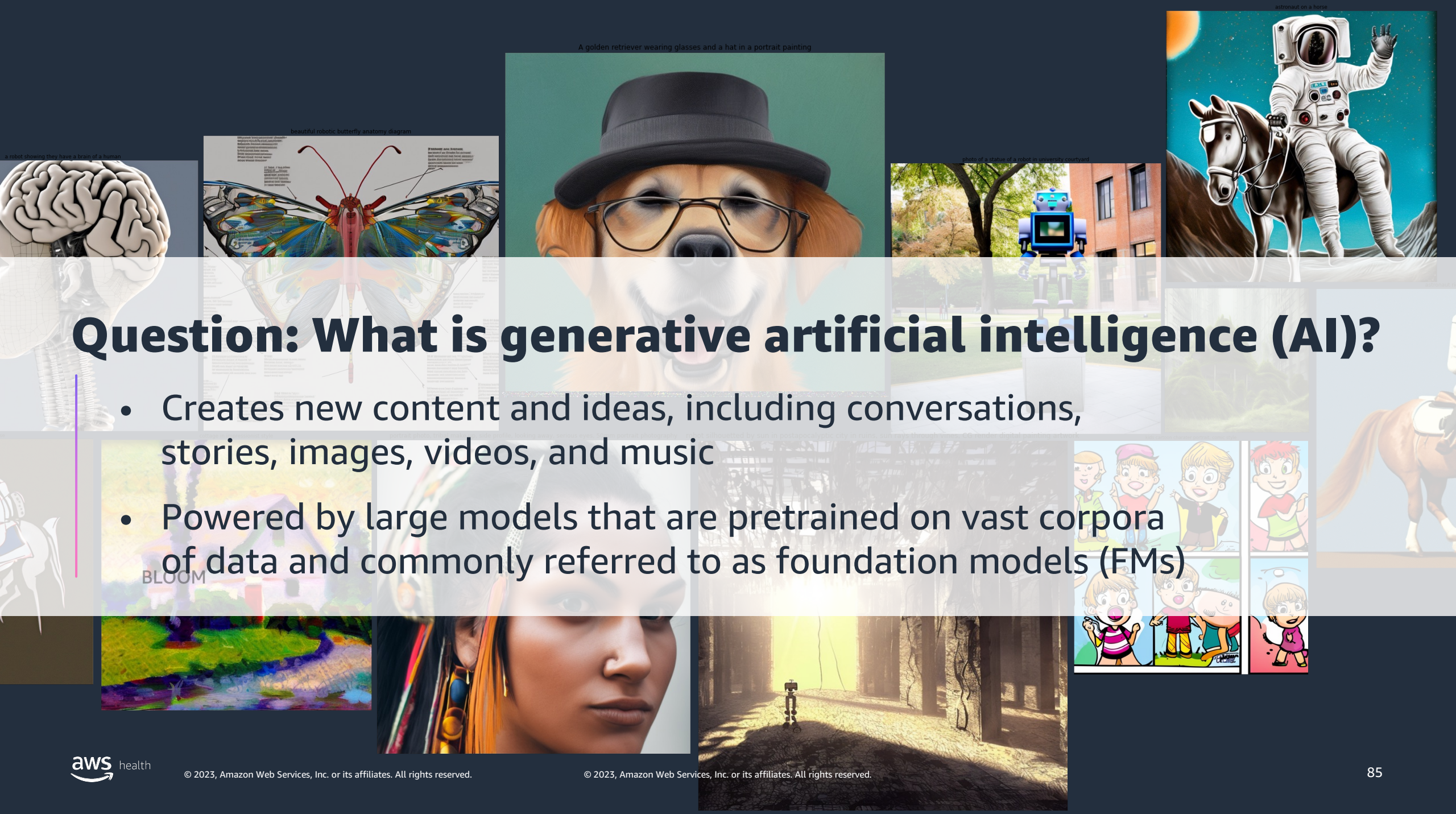
Lab: Bring your custom model

Lab 3. Bring your own model

Q&A and chat are open!

Until 15:45

Advanced ML & Generative AI



Question: What is generative artificial intelligence (AI)?

- Creates new content and ideas, including conversations, stories, images, videos, and music
- Powered by large models that are pretrained on vast corpora of data and commonly referred to as foundation models (FMs)

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Where does generative AI fit?

Artificial intelligence (AI)

Any technique that allows computers to mimic human intelligence using logic, if-then statements, and machine learning

Machine learning (ML)

A subset of AI that uses machines to search for patterns in data to build logic models automatically

Deep learning (DL)

A subset of ML composed of deeply multi-layered neural networks that perform tasks like speech and image recognition

Generative AI

Powered by large models that are pretrained on vast corpora of data and commonly referred to as foundation models (FMs)

Common generative AI use cases

Text generation

Q&A

Text summarization

Text extraction

Paraphrase rephrase

Search

Code generation

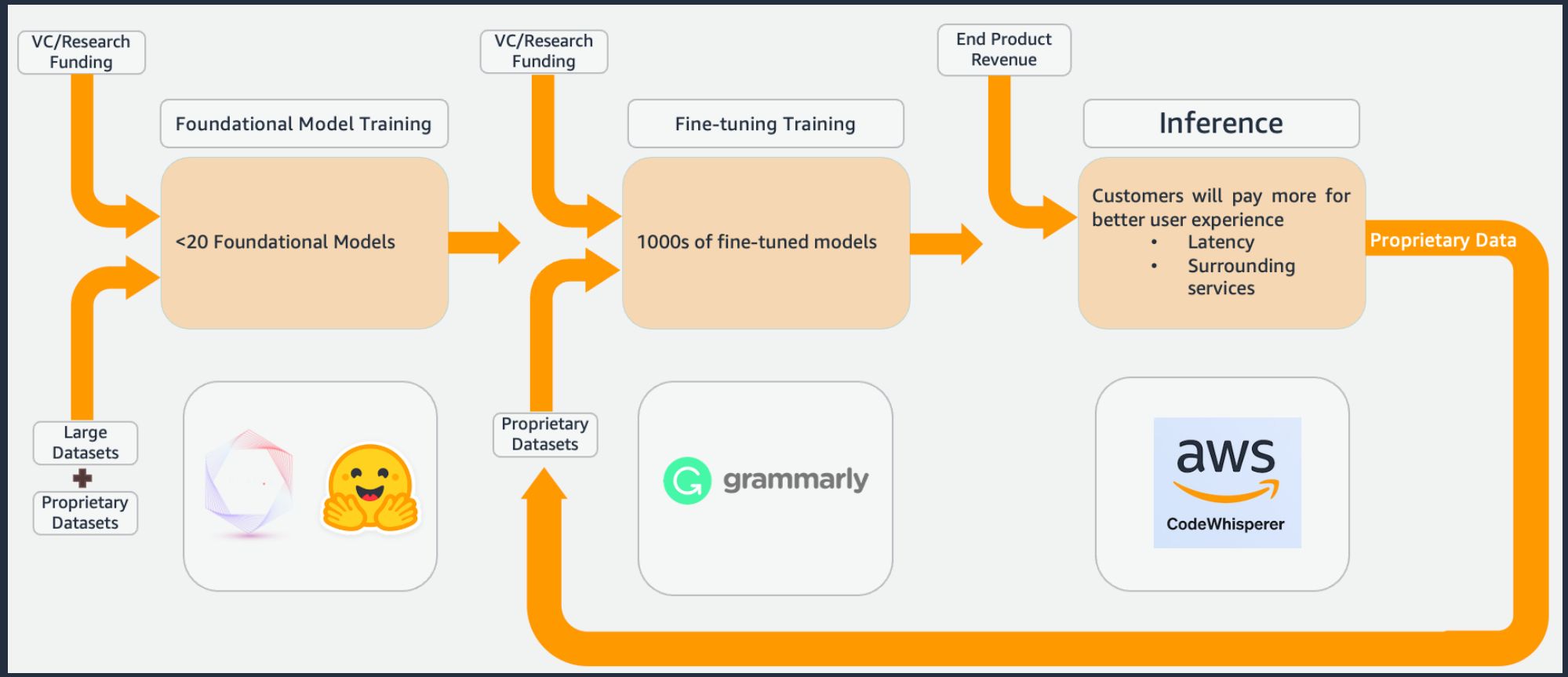
Image generation

Image classification

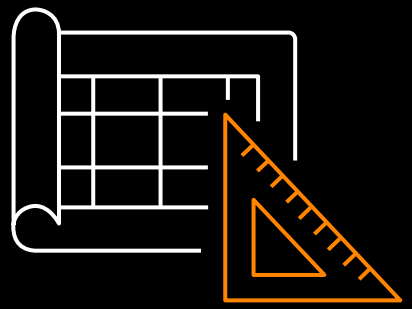
Audio generation

Video generation

GenAI Workload Flow



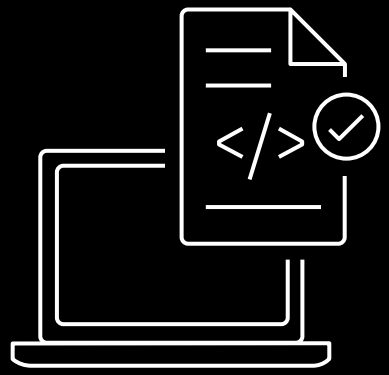
GenAI Workloads



Build your own foundation model from scratch

Expensive, time consuming and requires deep expertise

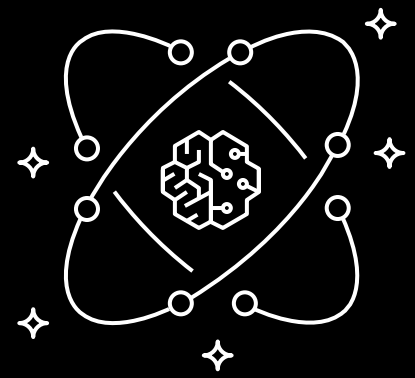
Model Provider



Use publicly available foundation models and Fine-Tune them

Substantial undifferentiated work needed to operationalize

Model Tuner

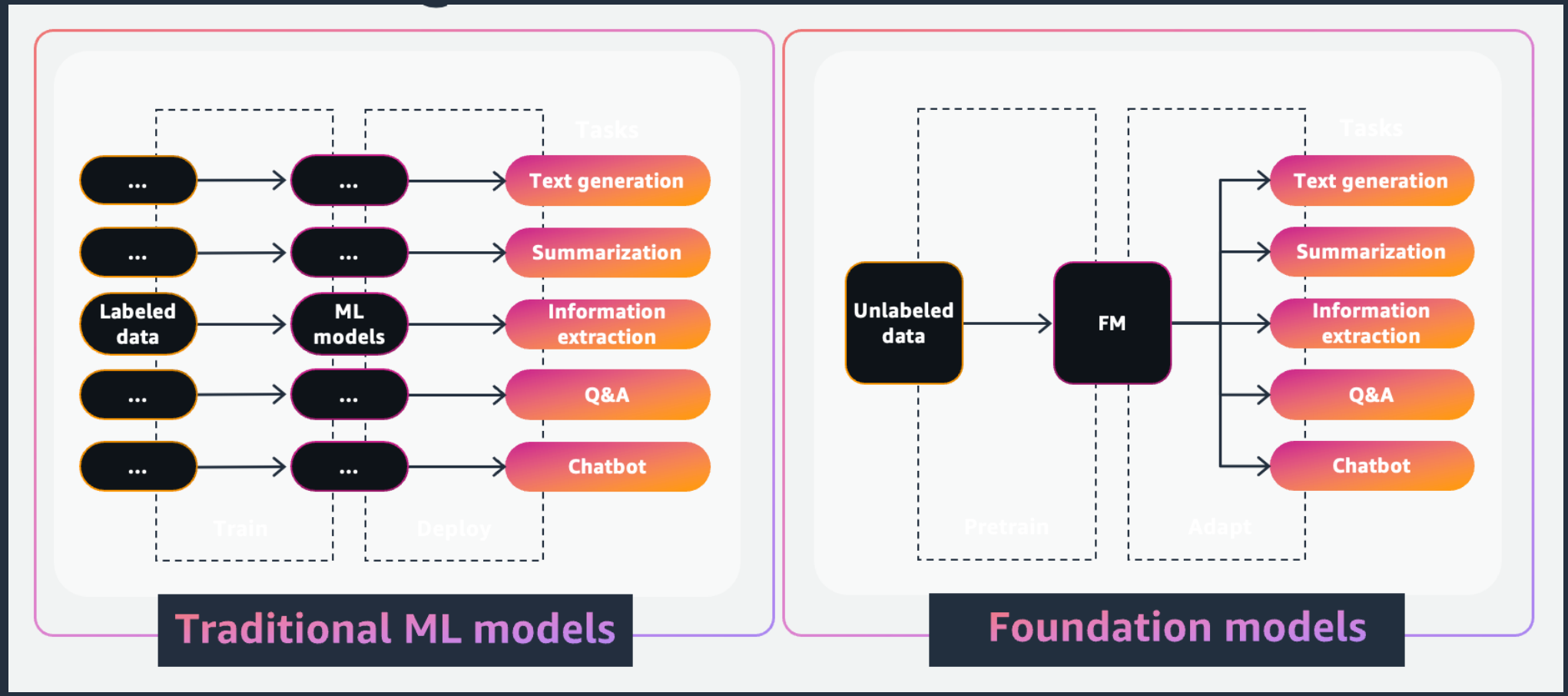


Use generative AI services or APIs offered by foundation model vendors

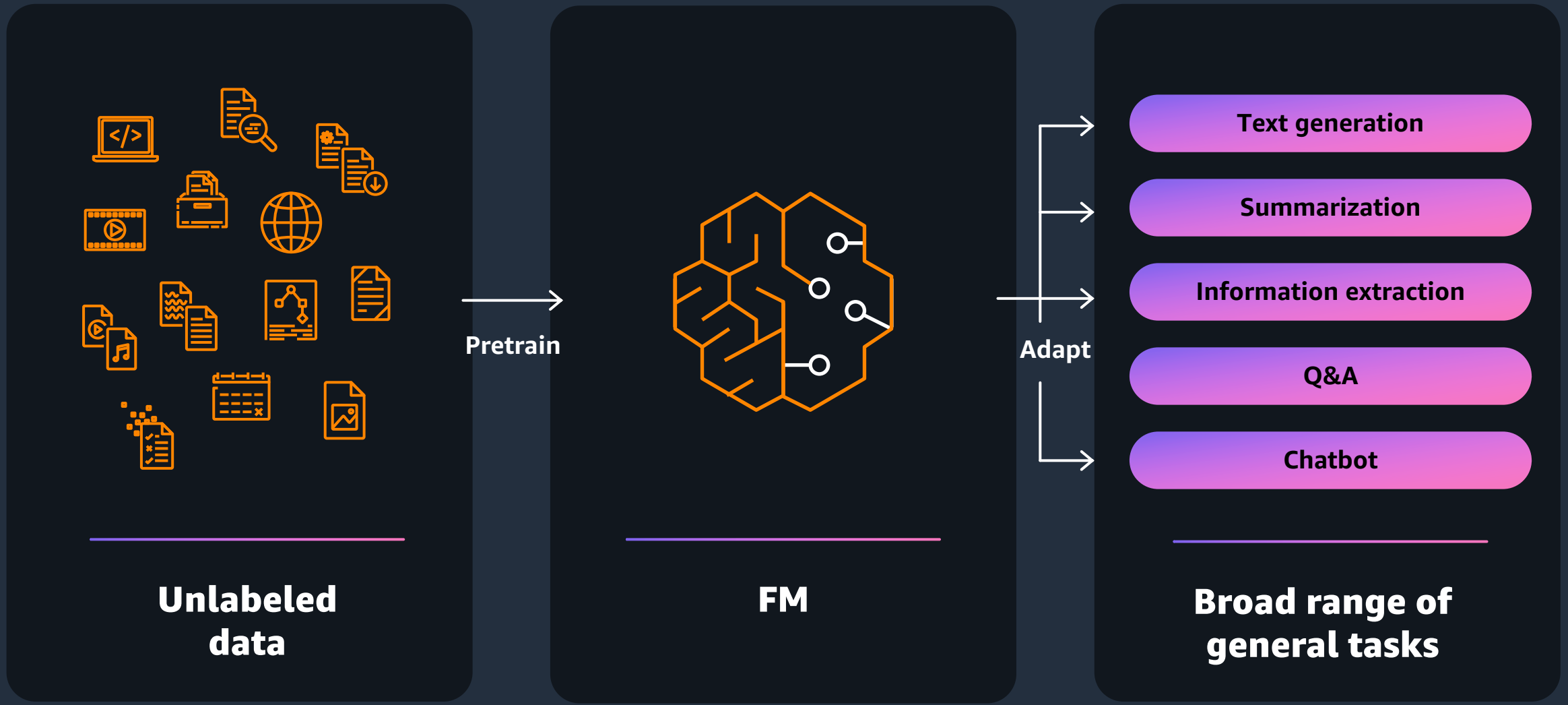
No control over data, costs, and no customization support

Model Consumer

Generative AI @ AWS: Foundational models



How foundation models work



Types of foundation models

Input	FM	Output
"Summarize the articles on impact of walking on heart health"	**Text-to-text** Generate text from simple natural-language prompts for various applications	"Ten thousand steps per day is optimum for maintaining a healthy heart"
"hand soap"	**Text-to-embeddings** Generate numerical representation of text for applications like search and finding similarities between documents	Numerical representation of "Hand soap refills Hand soap dispenser Hand soap antibacterial"
"a photo of an astronaut riding a horse on mars"	**Multimodal** Generate and edit images from natural-language prompts	

NEW

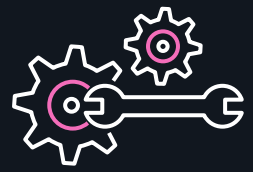
Amazon Bedrock

The easiest way to build and scale generative AI applications with FMs

Amazon Bedrock key benefits

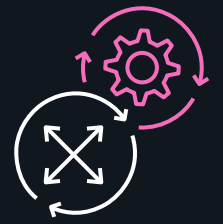
Accelerate development of generative AI applications using FMs through an API, without managing infrastructure

Choose FMs from AI21 Labs, Anthropic, Stability AI, and Amazon to find the right FM for your use case



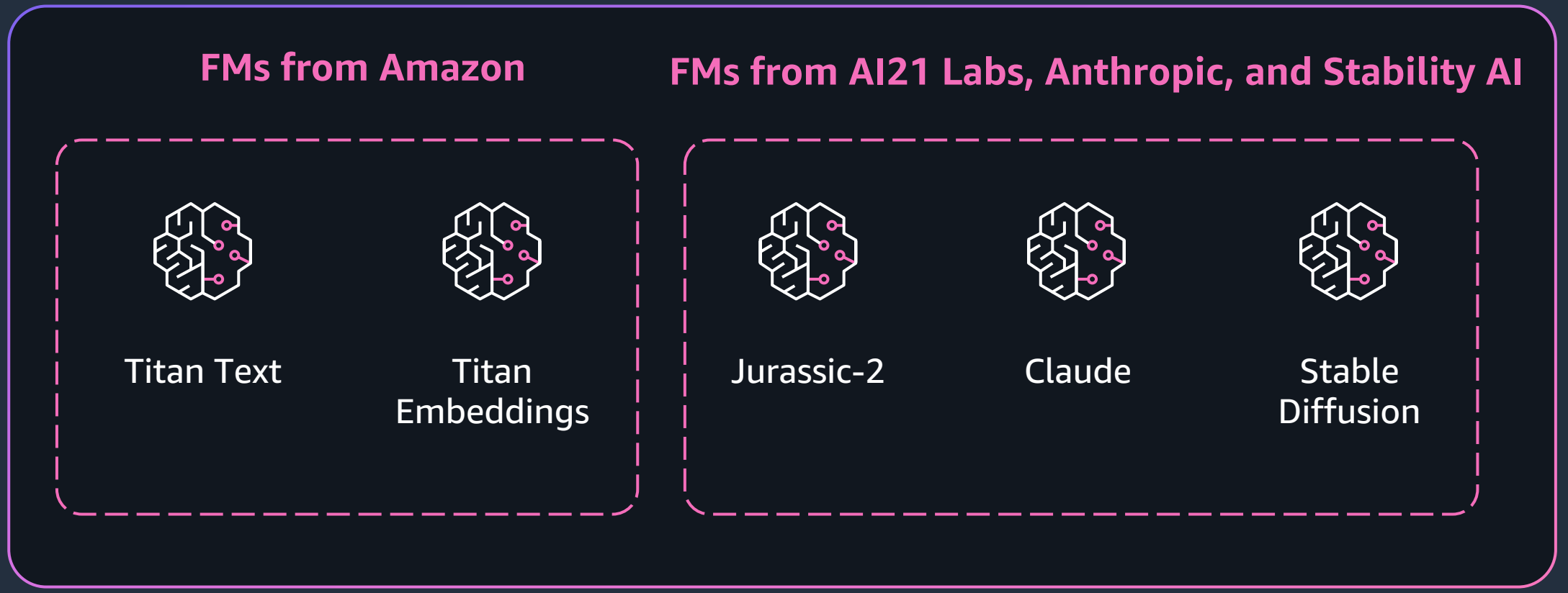
Privately customize FMs using your organization's data

Enhance your data protection using comprehensive AWS security capabilities



Use AWS tools and capabilities that you are familiar with to deploy scalable, reliable, and secure generative AI applications

Bedrock supports a wide range of foundation models



Amazon Titan

INNOVATE RESPONSIBLY WITH HIGH-PERFORMING FMs FROM AMAZON

Titan Text
focused on NLP tasks

Titan Embeddings
for enterprise tasks such as search and personalization

Benefits

- Built with 20+ years of Amazon ML experience
- Automate language tasks such as summarization and text generation with Amazon Titan Text FM
- Enhance search accuracy and improve personalized recommendations with Amazon Titan Embeddings FM
- Support responsible use of AI by reducing inappropriate or harmful content

Foundation models from top AI startups

Jurassic-2

Multilingual LLMs for text generation in Spanish, French, German, Portuguese, Italian, and Dutch

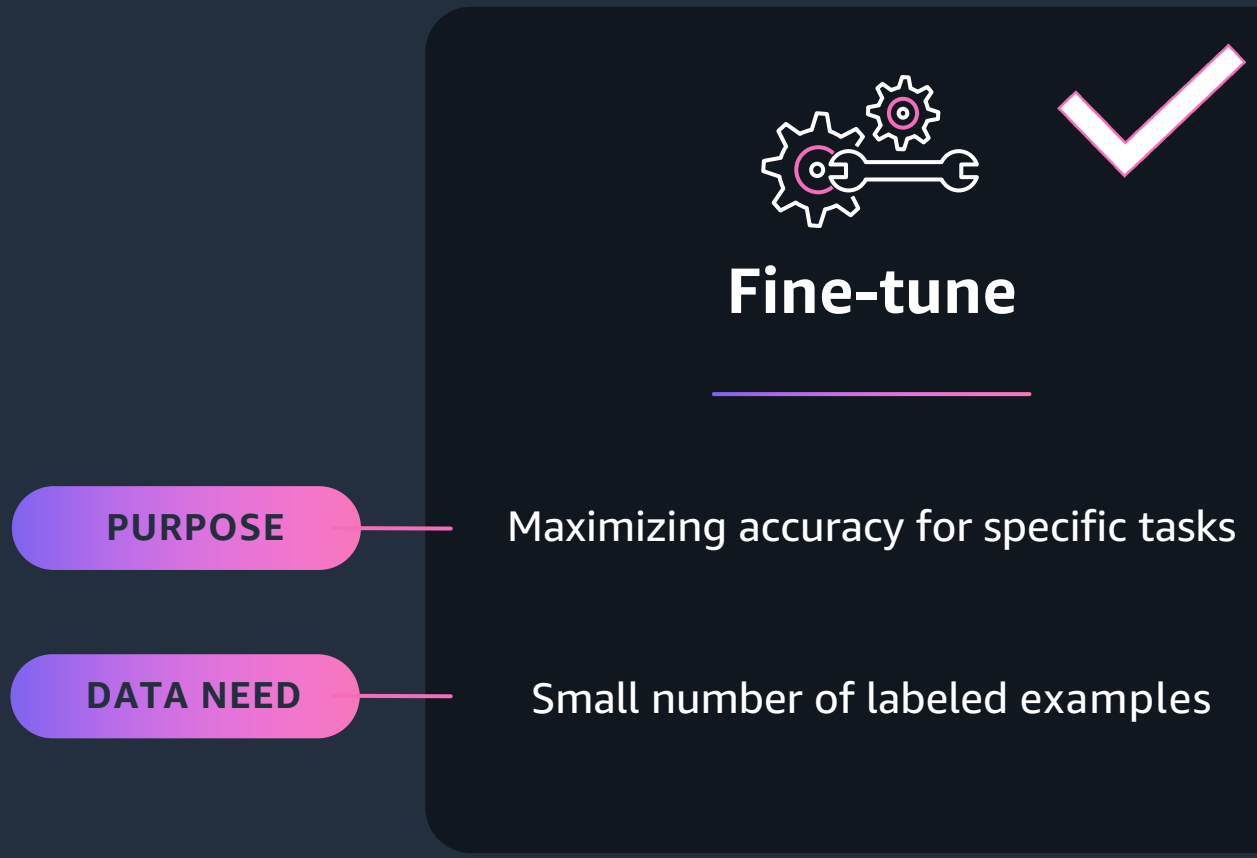
Claude

LLM for conversations, question answering, and workflow automation based on research into training honest and responsible AI systems

Stable Diffusion

Generation of unique, realistic, high-quality images, art, logos, and designs

Privately customize foundation models using your organization’s data



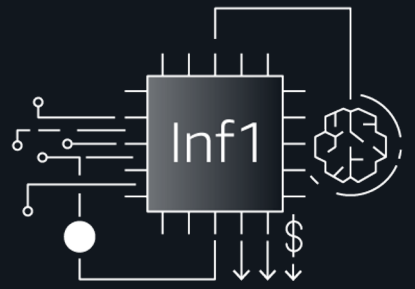
NEW

Amazon EC2 Trn1n and Amazon EC2 Inf2

Purpose-built ML accelerators for the best price performance for training and inference in the cloud

Purpose-built accelerators for generative AI

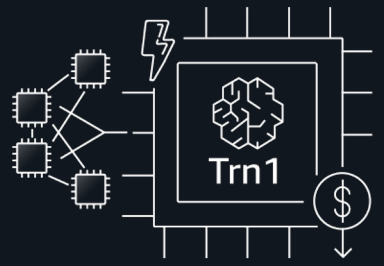
AWS Inferentia



Lowest cost per inference in the cloud for running deep learning (DL) models

Up to 70% lower cost per inference
than comparable Amazon EC2 instances

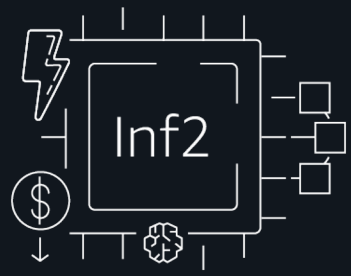
AWS Trainium



The most cost-efficient, high-performance training of LLMs and diffusion models

Up to 50% savings on training costs
over comparable Amazon EC2 instances

AWS Inferentia2



High performance at the lowest cost per inference for LLMs and diffusion models

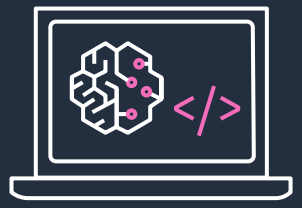
Up to 40% better price performance
than comparable Amazon EC2 instances

NEW

Amazon CodeWhisperer

Build applications faster and more securely with an AI coding companion

Amazon CodeWhisperer: Now generally available and free to use for individual developers!



Generate code suggestions in real time

Scan code for hard-to-find vulnerabilities

Flag code that resembles open-source training data or filter by default

During preview Amazon ran a productivity challenge, and participants who used Amazon CodeWhisperer were 27% more likely to complete tasks successfully and did so an average of 57% faster than those who did not use CodeWhisperer.

Healthcare and Life Sciences

Accelerate drug discovery and research using foundation models to design vaccines, antibodies, and enzymes

Improve customer support with faster agent assisted claim and document processing

Deliver better care with personalized medicine

Keep patient data safe using synthetic data generation for research

Lab: Advanced ML & Generative AI

Until 17:20

aws health

EMEA HEALTHCARE & LIFE SCIENCES WORKSHOPS

https://aws-experience.com/

- June 27th: AWS Cloud Fundamentals
- June 28th: Genomic Data Analysis with Amazon Omics
- July 4th: Machine Learning
- July 5th: AWS for Pharma Manufacturing
- July 6th: Security, Encryption & Data Protection Immersion Day
- July 11th: Sustainability
- July 13th: High-performance computing
- July 18th: Compliance in the Cloud

aws health

Please complete the workshop survey

http://bit.ly/3CVzhO0

aws health

Thank you!

Yegor Tokmakov

Sr. Solutions Architect
AWS Health

Dmitri Laptev

Sr. Solutions Architect
AWS Startups

Aamna Najmi

Data Scientist
AWS Professional Services