Deploy MultiModel using Nvidia Triton Inference Server

Agenda

- Quick Overview of Inferences and Challenges
- Overview of NVidia Triton Inference Server
- AWS GPU compute offering
- **Demo - Deploy MultiModel in Sagemaker using Triton Inference Server**

Ayyanar J

DevOps Lead and
Data Science System Engineer

Follow me on https://www.linkedin.com/in/jayyanar/

https://cloudnloud.com **info@cloudnloud.com**

ABOUT ME

I started my career as humble Hardware and Networking engineer in 2005 in HCL Infosystem.

Over the last 17 years of my IT Career, I have worked as Wintel, Linux Middleware Engineer, Infrastructure Architect, Cloud Solution Architect, Bigdata Manager, DevOps Lead, and DataPlatform Lead Engineer. I worked in Europe, Canada, and the US for a brief period of time.

I have 50+ Technical Certification in AWS, Azure, IBM Cloud, CKA/CKAD, TOGAF Level 2 Certified

Currently, I am working as an individual consultant for Europe based Company

I always love to learn - Unlearn - Relearn, Motivated to Share knowledge with peers, community and learn from them.

CloudnLoud **Community**

MACHINE LEARNING LIFECYCLE

Training

Inference

Problem Statement -

How we address via ML/AI

Value out of ML

1

Gather Required data

Explore the data

Set the Baseline Accuracy ~

2

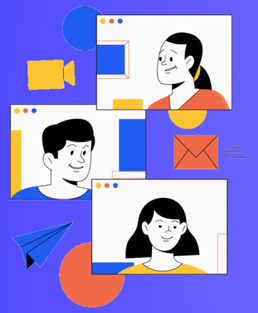
No

Identify the Model to be used

Train the Model using training data

Evaluate the Model with test data

3



Accuracy / Prediction Achieved as Expected

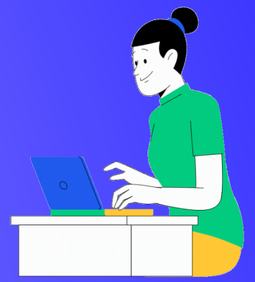
Yes

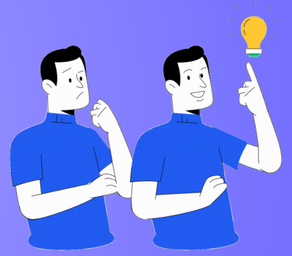
4

Deploy the Model as Batch Processing.

Deploy the Model as Realtime Inference

5





For Realtime Inference -

Monitor the Accuracy / Throughput - Collect

The Realtime data collected

6

CORE MLOPS CAPABILITIES

Model Serving (Inference)

Model Registry

Online Experimentation

Machine learning Metadata

and

Artifacts Repository

MACHINE LEARNING INFERENCE

Inference - Using Trained models to predict outcomes from new observations in efficient deployment

- Online inference in near real time for high-frequency singleton requests (or mini batches of requests), using interfaces like REST or gRPC.
- Streaming inference in near real time, such as through an event-processing pipeline.
- Offline batch inference for bulk data scoring, usually integrated with extract, transform, load (ETL) processes.
- Embedded inference as part of embedded systems or edge devices.

WHY WE NEED TO TALK ABOUT INFERENCE NOW

- In the next 5 years, large models like GPT-3/4, BERT, and T5, BioBert, BioGPT capable of producing impressive results on a variety of tasks in Natural Language Processing (NLP), including text classification, language translation, and question answering will bring value to business and a lot of innovation.

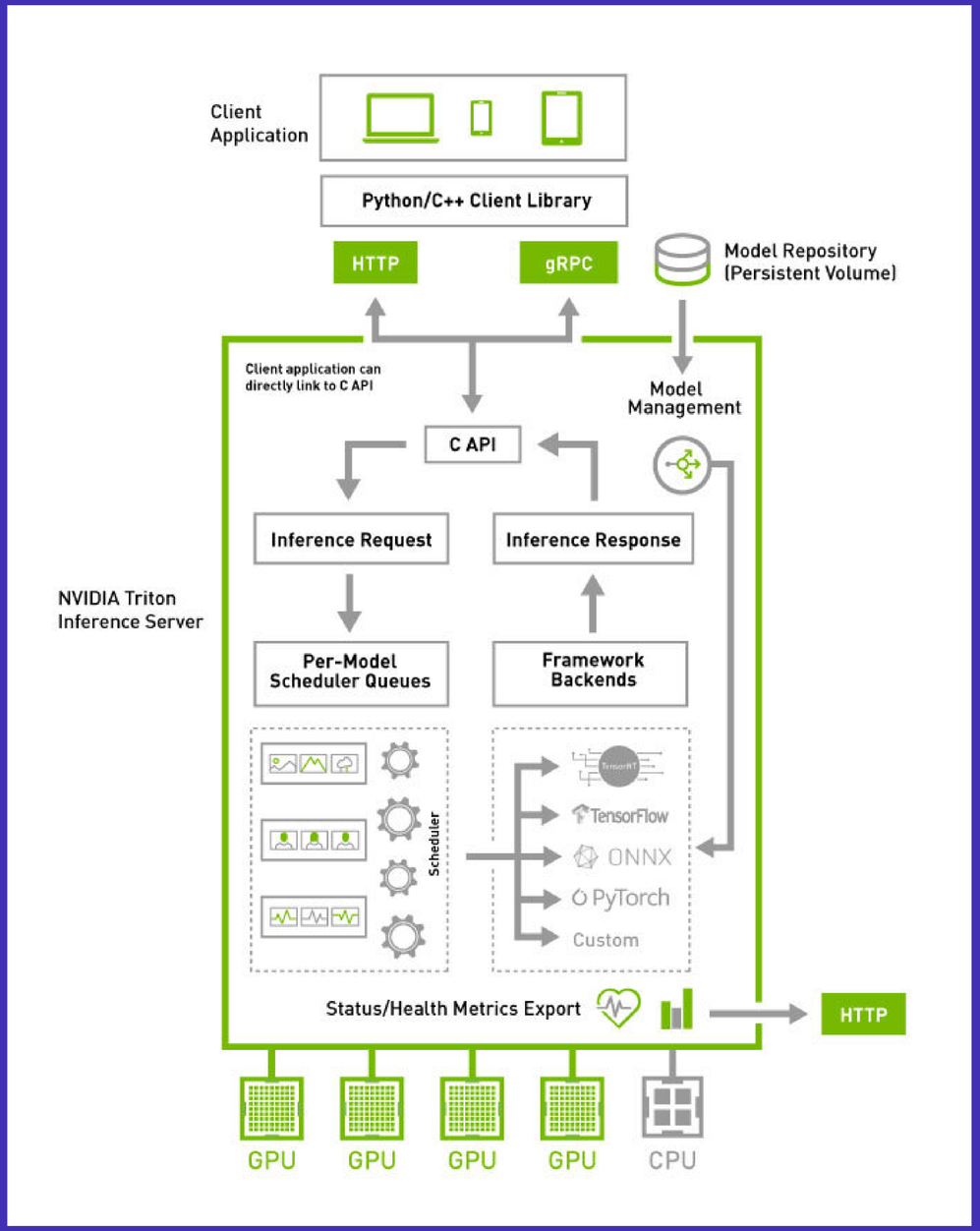
- Also Large computer image models are deep neural networks trained to perform tasks related to image processing, such as image classification, object detection, segmentation, and generation. Some examples include ResNet, Inception, and GANs. These models typically have a large number of parameters, trained on massive datasets of images, and can produce state-of-the-art results on a wide range of image-related tasks.

- Now we have a huge array of foundation models. Business or Organization need to host these large model inferences to scale cost effectively.

INFERENCES CHALLENGES FOR MLOPS ENGINEER

- My Journey - Local, Lambda, Fargate, EKS, Sagemaker Inference, Serverless Inference.
-
- Multiple Model Frameworks and Serving Options - Pytorch, Tensorflow, XGBoost, ONNX, TensorRT.
- Deployment Options -Single Model, Multi Model, Multi Containers, Serial Inference, OnPrem, Selecting Cloud vendor.
- Migrating Legacy Models
- Performance - Latency, Throughput, Monitoring the Metrics.
- Diverse Compute Option - CPU, GPU, TPU

ARCHITECTURE OF NVIDIA TRITON SERVER



https://docs.nvidia.com/deeplearning/triton-inference-server/user-guide/docs/_images/arch.jpg

OVERVIEW OF NVIDIA TRITON SERVER

- **Support for Multiple frameworks: Triton can be used to deploy models from all major frameworks. Triton supports TensorFlow GraphDef, TensorFlow SavedModel, ONNX, PyTorch TorchScript, TensorRT, RAPIDS FIL for tree based models, and OpenVINO model formats.**

- **Model pipelines: Triton model ensemble represents a pipeline of one or more models or pre/post processing logic and the connection of input and output tensors between them. A single inference request to an ensemble will trigger the execution of the entire pipeline.**

- **Concurrent model execution: Multiple models can run simultaneously on the same GPU or on multiple GPUs for different model management needs.**

- **Dynamic batching: For models that support batching, Triton has multiple built-in scheduling and batching algorithms that combine individual inference requests together to improve inference throughput. These scheduling and batching decisions are transparent to the client requesting inference.**

DEMO

https://github.com/triton-inference-server/server

NVIDIA GPU Cloud (NGC) Container Registry

https://github.com/NVIDIA/DeepLearningExamples

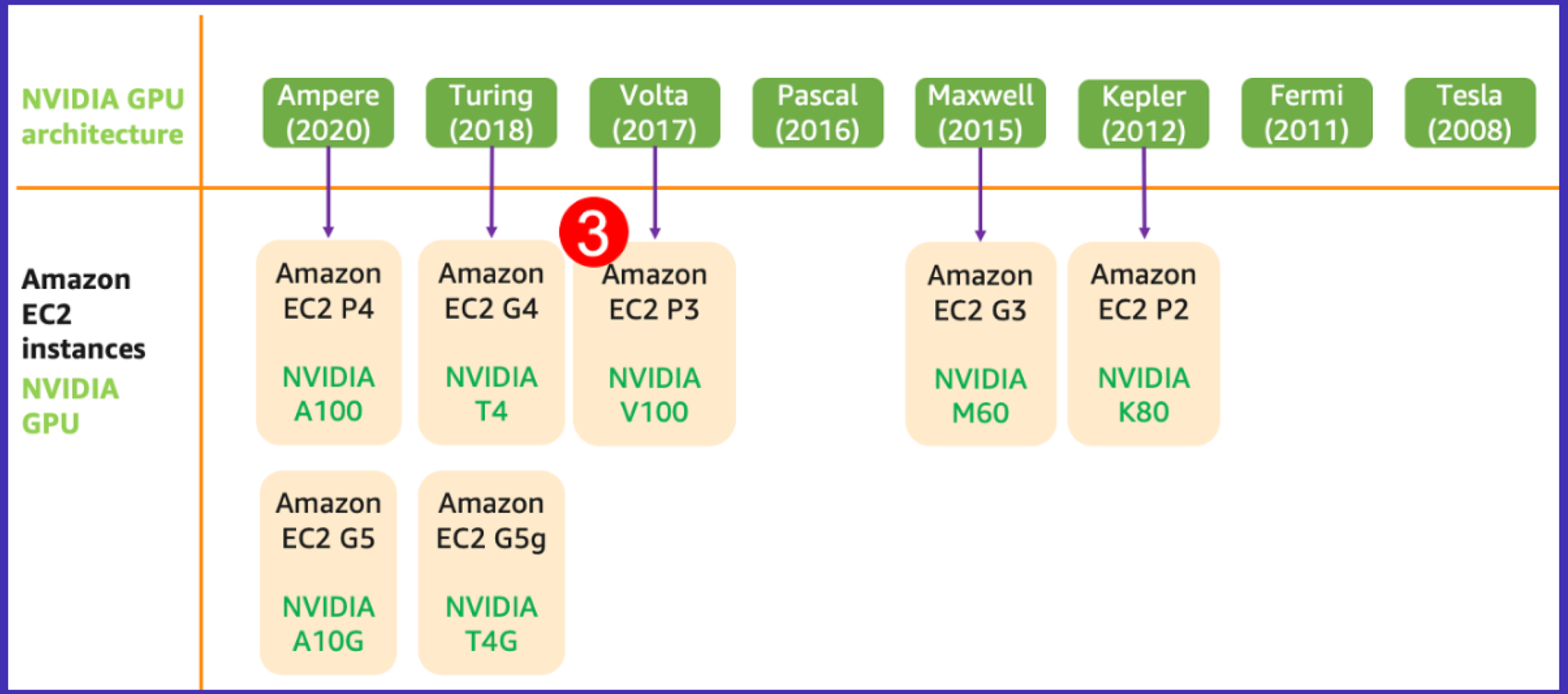
SageMaker Multi-Model endpoints with GPU Support

https://github.com/aws/amazon-sagemaker-examples/blob/main/multi-model-endpoints/mme-on-gpu/cv/resnet50_mme_with_gpu.ipynb

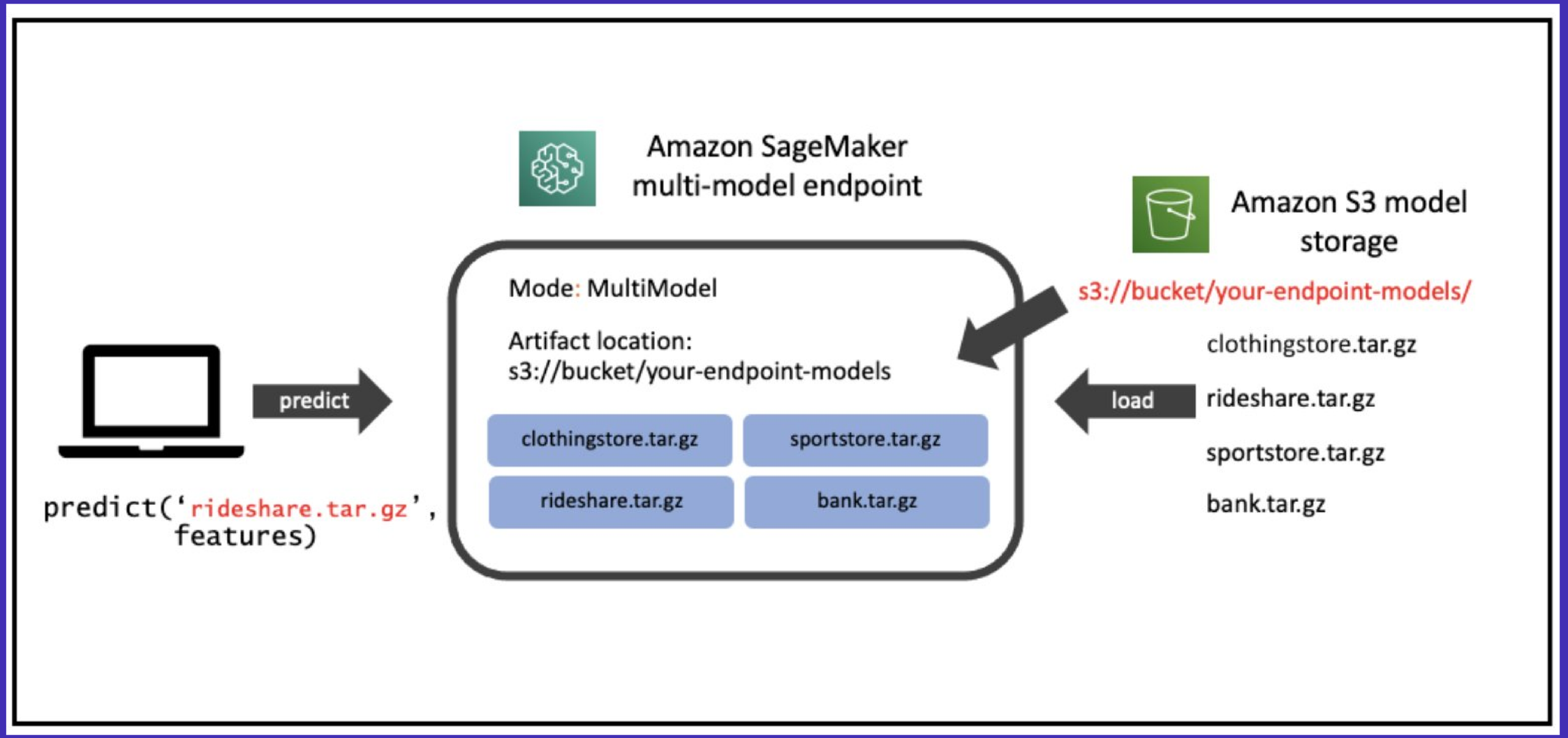
NVIDIA GPU OFFEREING FROM AWS

Instance family	Instance type	vCPUs	GiB of memory per vCPU	GPUs	GPU memory
p2	ml.p2.xlarge	4	15.25	1	12
p3	ml.p3.2xlarge	8	7.62	1	16
g5	ml.g5.xlarge	4	4	1	24
g5	ml.g5.2xlarge	8	4	1	24
g5	ml.g5.4xlarge	16	4	1	24
g5	ml.g5.8xlarge	32	4	1	24
g5	ml.g5.16xlarge	64	4	1	24
g4dn	ml.g4dn.xlarge	4	4	1	16
g4dn	ml.g4dn.2xlarge	8	4	1	16
g4dn	ml.g4dn.4xlarge	16	4	1	16
g4dn	ml.g4dn.8xlarge	32	4	1	16

NVIDIA GPU OFFEREING FROM AWS

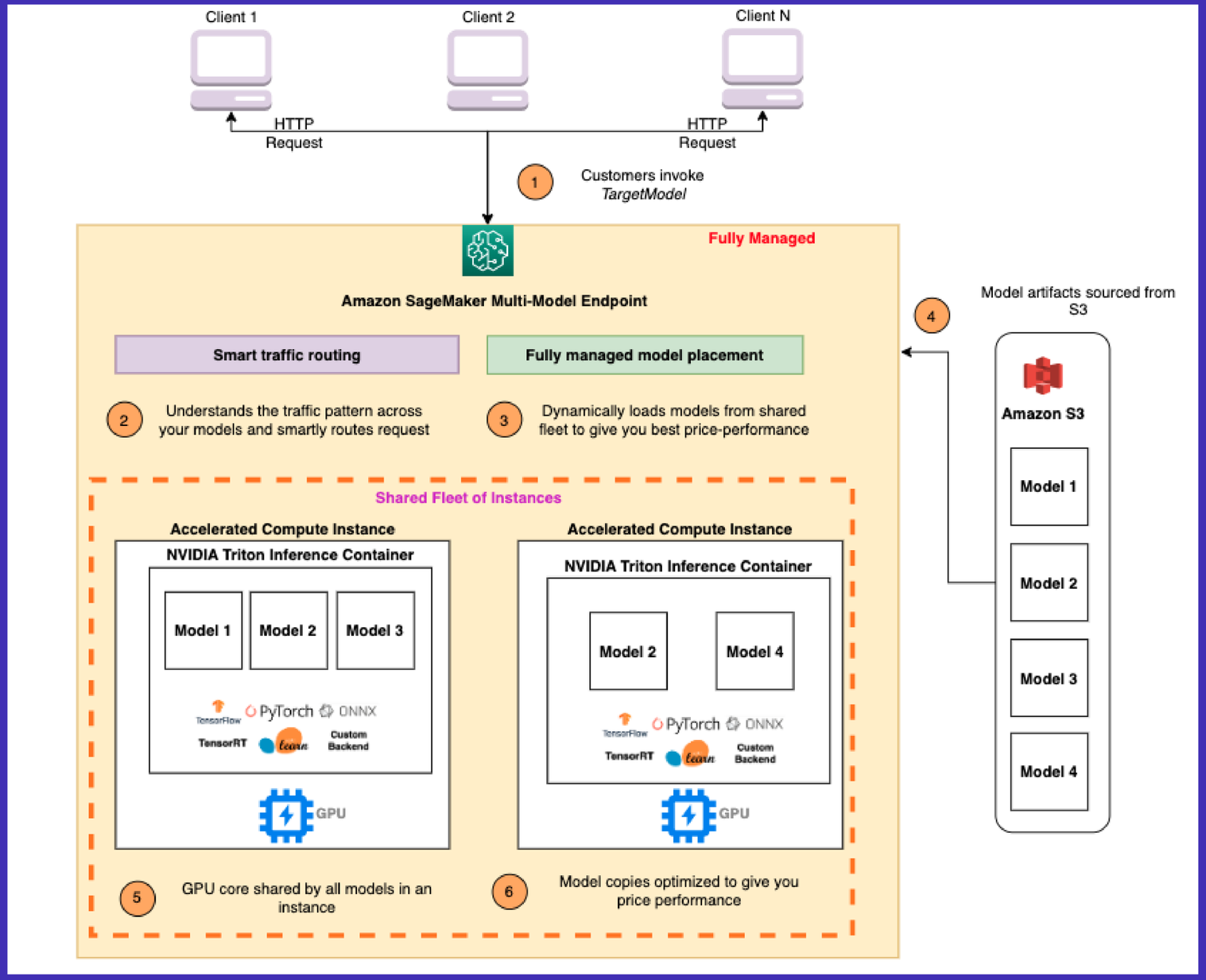


DEMO - MME - MULTI MODEL ENDPOINT - EXAMPLE SUPERAPP



Ref: https://d2908q01vomqb2.cloudfront.net/f1f836cb4ea6efb2a0b1b99f41ad8b103eff4b59/2022/10/25/ML-9791-image001.jpg

DEMO - MME - MULTI MODEL ENDPOINT - LIFECYCLE EXAMPLE



- Use an NVIDIA Triton inference container on SageMaker MMEs, using different Triton model framework backends such and PyTorch and TensorRT
- Convert ResNet-50 models to optimized TensorRT engine format and deploy it with a SageMaker MME.
- Set up auto scaling policies for the MME
- Get insights into instance and invocation metrics using Amazon CloudWatch

Ref: https://d2908q01vomqb2.cloudfront.net/f1f836cb4ea6efb2a0b1b99f41ad8b103eff4b59/2022/10/25/ML-9791-image003.png

DEMO - MME - MULTI MODEL ENDPOINT - LIFECYCLE

GET Request

SageMaker manages the lifecycle of models hosted on multi-model endpoints in the container's memory.

Routes the request to an instance behind the endpoint.

Downloads the model from the S3 bucket to that instance's storage volume

Loads the model to the container's memory (CPU or GPU, depending on whether you have CPU or GPU backed instances) on that accelerated compute instance.

Process and Scale

If the model is already loaded in the container's memory, invocation is faster because SageMaker doesn't need to download and load it.

If the model receives many invocation requests, and there are additional instances for the multi-model endpoint, SageMaker routes some requests to another instance to accommodate the traffic (Cold Start due to load to storage).

Unload and Delete

Memory utilization is high and SageMaker needs to load another model into memory, it unloads unused models from that instance's container to ensure that there is enough memory to load the model.

If the instance's storage volume reaches its capacity, SageMaker deletes any unused models from the storage volume.

To delete a model, stop sending requests and delete it from the S3 bucket