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

AIM407-R1

Choosing the right ML instance for training and inference on AWS

Samir Araujo (he/him)

Principal AI/ML Solutions Architect
AWS

Raghu Ramesha (he/him)

ML Solutions Architect
AWS

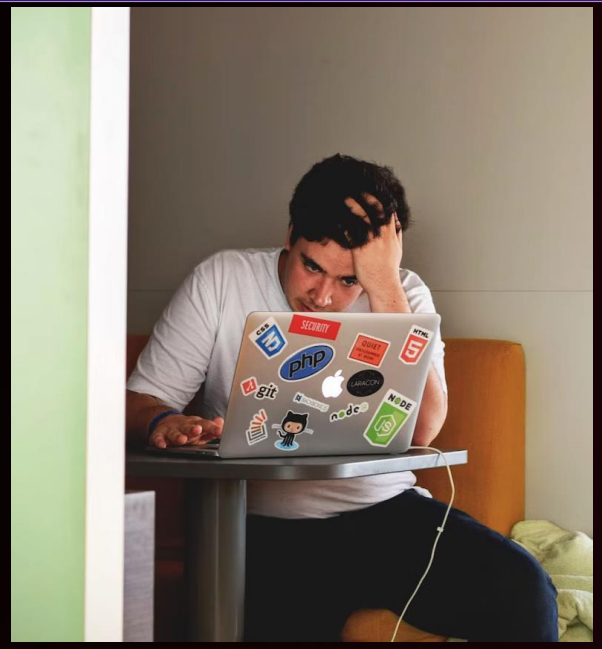
Agenda

- Common scenarios for training and inference
- Challenges
- Picking the right training instances
- Picking the right Inference instances
- Discussion and Q/A

ML training common scenarios

Small
Small models
Classic machine learning (ML)/ shallow neural networks
Small/toy datasets

Intermediate/big
Medium/big models
DL/big trees/clusters
Intermediate/big datasets



Huge
Transformers
Models with 10Bi+ params
Huge datasets/TB

Photos, https://unsplash.com/photos/1K9T5YiZ2WU; https://unsplash.com/photos/-2vD8lIhdnw / Unsplash

Challenges

Data scientists need to build/deploy different types/sizes of ML models in less time, paying less

Best cost performance hardware configuration to build/deploy ML models

Choosing the best hardware setup that provides best cost-performance for ML model building/deploying

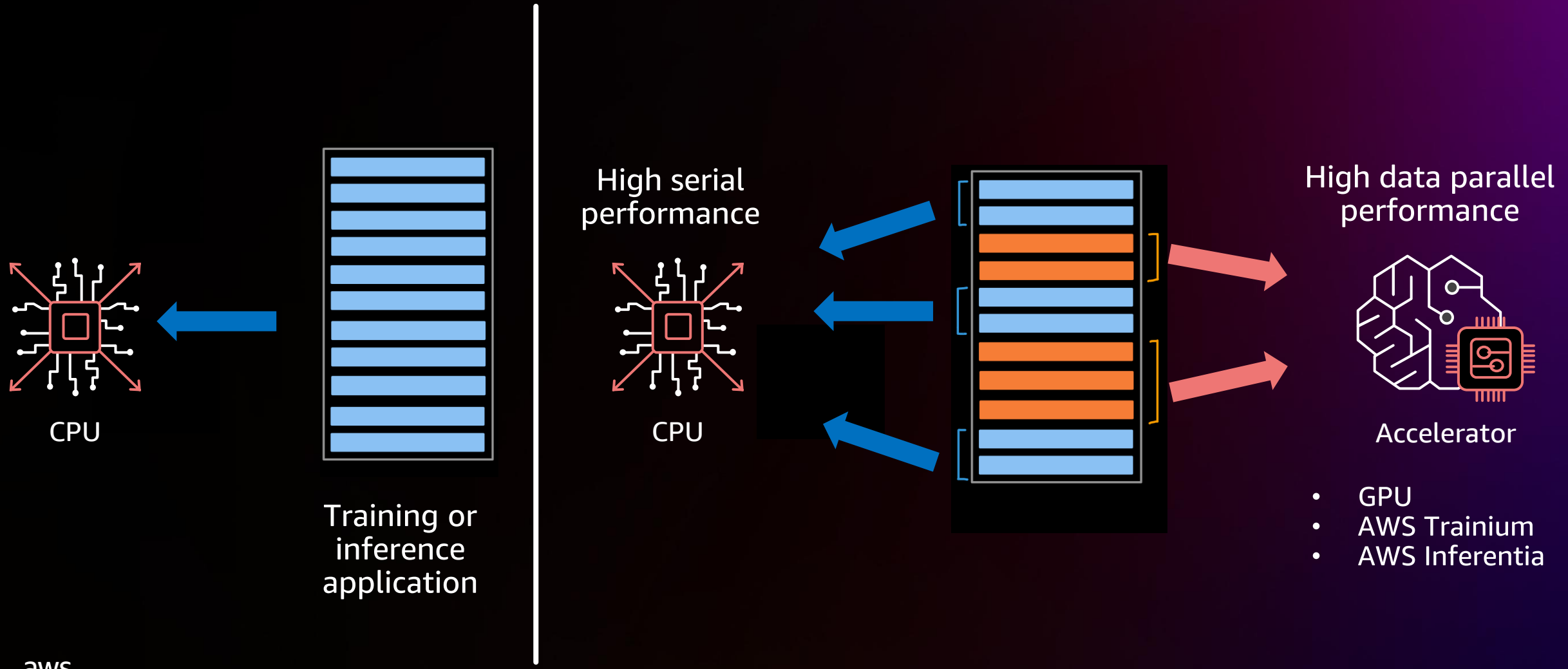
Photo, https://unsplash.com/photos/2gYsZUmockw / Unsplash

Common questions

- Shall I use CPU or hardware accelerator?
- Do I need distributed training or is single host training enough?
- Which filesystem is the best option to store the dataset?
- How do I optimize data transfer from the filesystem to the training host?

- My model has 10B+ params and the dataset has terabytes. What should I do?
  - Spoiler: Amazon EC2 UltraClusters

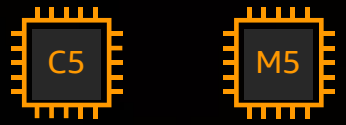
Choosing: CPU-only or CPU + ML accelerators?



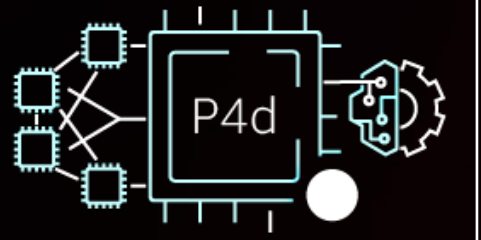
Training instances

Instances for ML training

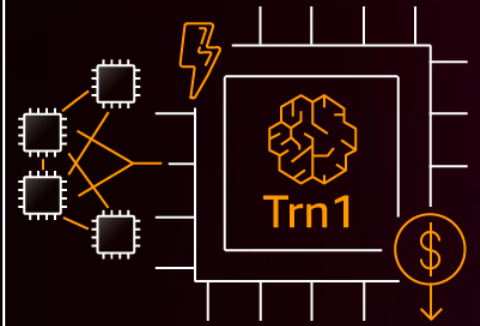
CPU instances



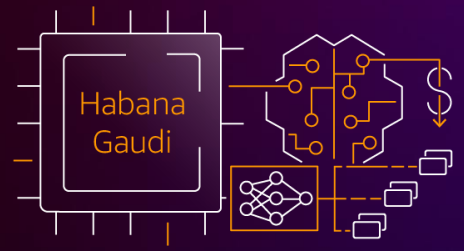
GPU instances



AWS Trainium



Habana Gaudi-based instances



Amazon EC2 only

https://aws.amazon.com/ec2/instance-types/

CPU instances for training

When to consider

- Traditional ML models (Scikit-learn)
- Prototyping DL models
- Training smaller DL models
- Fine-tuning on a smaller dataset
- Non-time-sensitive training, trade off time for cost

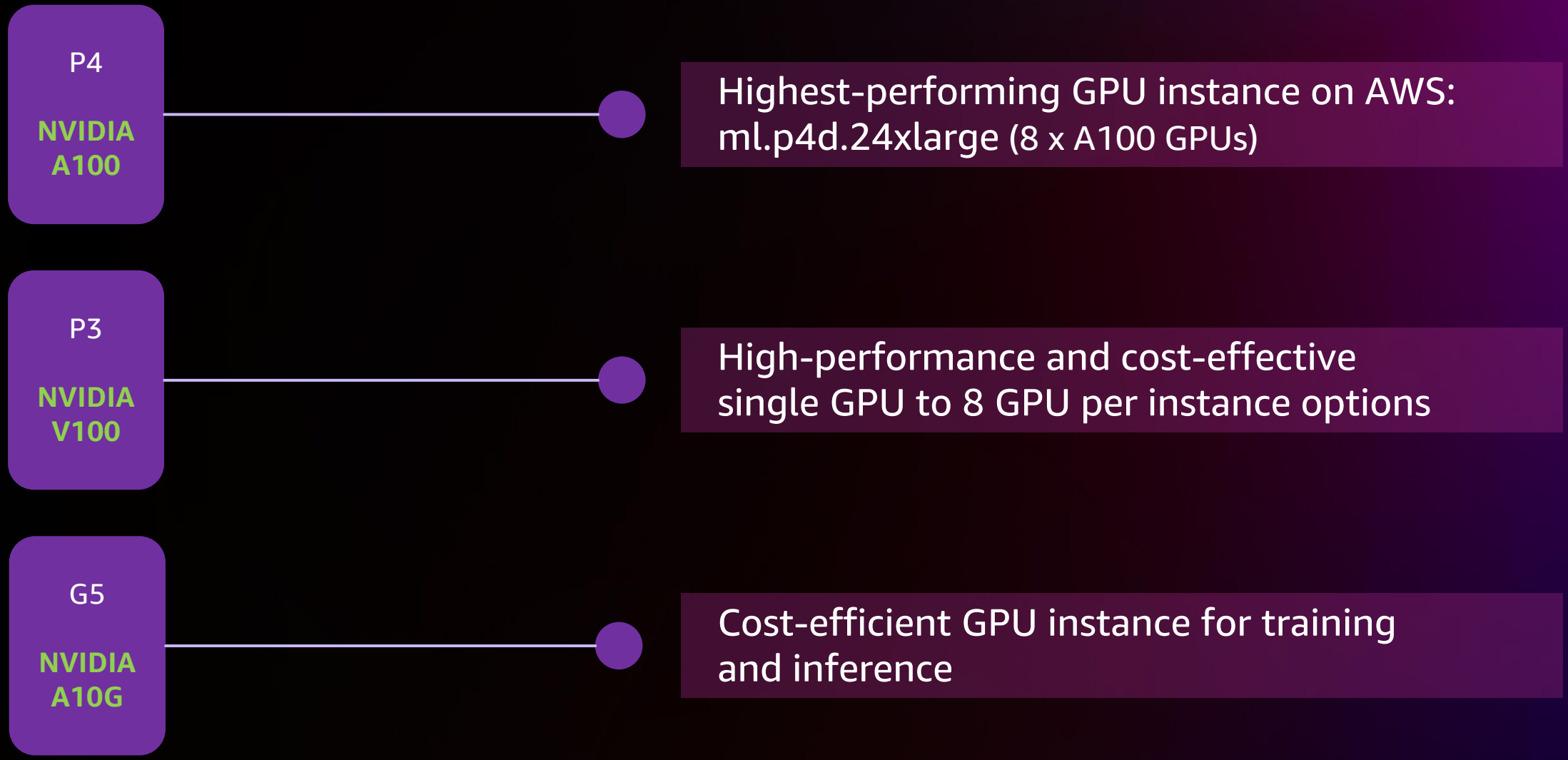
- CPU options with **2–128 vCPUs** for multi-threaded workloads
- System memory from **4–256 GiB**
- C5/6 (featuring on Intel CPUs), C5/6a (featuring AMD CPU)
- Multi-threaded and CPU-optimized Amazon SageMaker built-in algorithms and ML frameworks: **TensorFlow, PyTorch, MXNet, XGBoost, others**

GPU instances for training

P4 instances	NVIDIA A100 GPU memory: 40 GB
P3 instances	NVIDIA V100 GPU memory: 16 GB, 32 GB
P2 instances	NVIDIA K80 GPU memory: 12 GB
G5 instances	NVIDIA A10G GPU memory: 16 GB
G4 instances	NVIDIA T4 GPU memory: 16 GB

- Reduced precision types: FP64, FP32, FP16, Tensor Cores (mixed-precision)
- GPU memory: Up to 40 GB
- Fast GPU interconnect: NVLink high-bandwidth interconnect
- GPU-accelerated machine learning frameworks: TensorFlow, PyTorch, MXNet, XGBoost, others

How do I choose the right GPU instance?



Amazon EC2 Trn1/Trn1n instances

THE MOST COST-EFFICIENT HIGH-PERFORMANCE TRAINING INSTANCE

Trn1(n)

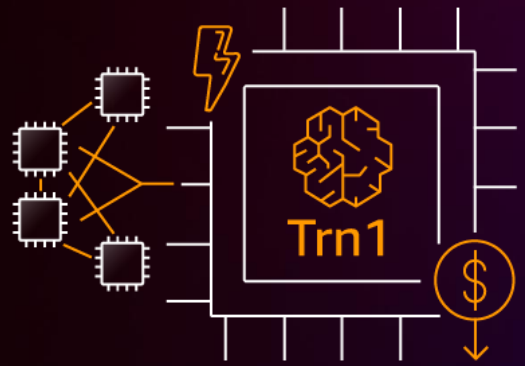
BF16/FP16	TF32	FP32
3.4 PFLOPS	3.4 PFLOPS	840 TFLOPS

AGGREGATE ACCELERATOR MEMORY	PEAK MEMORY BANDWIDTH
512 GB	13.1 TB/sec

EFA NETWORK CONNECTIVITY	NEURON-CORE V2 NEURON-LINK V2
800/1600 Gbps	

AWS Trainium

High performance machine learning training chip, purpose-built by AWS

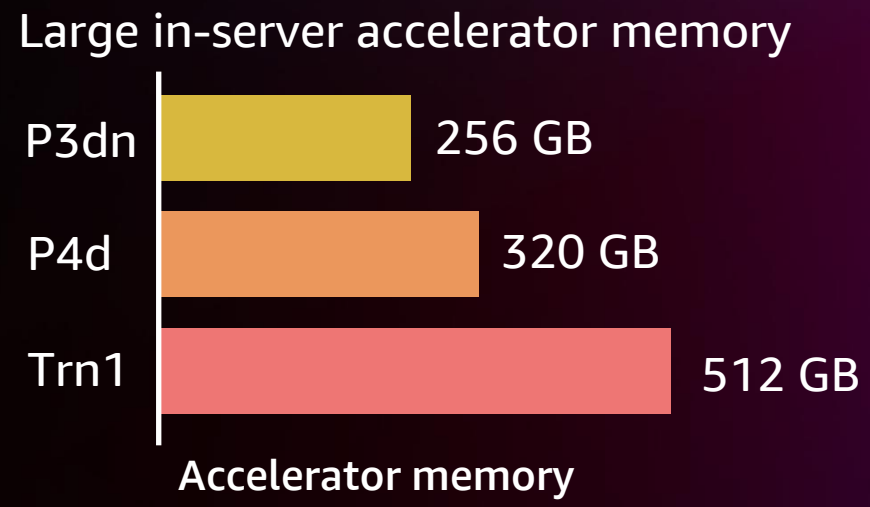


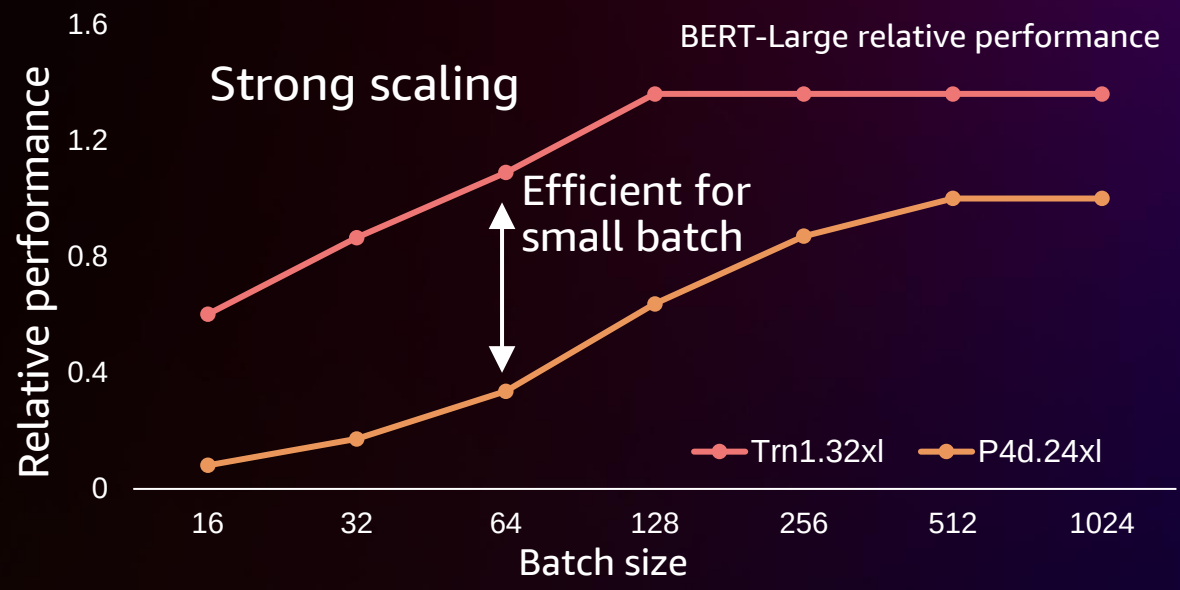
EC2 Trn1/Trn1n Instances

The most cost-efficient high-performance training instance

Trn1: Built for scale out

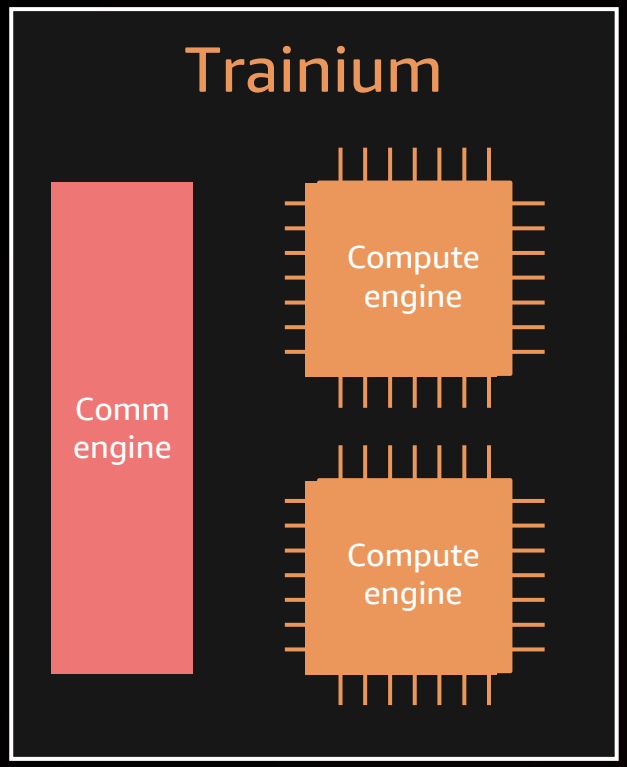
- 60% higher accelerator memory vs. P4d
- 2x network bandwidth vs. P4d
- Native support for PyTorch and TensorFlow
- Train on Trn1 and deploy anywhere



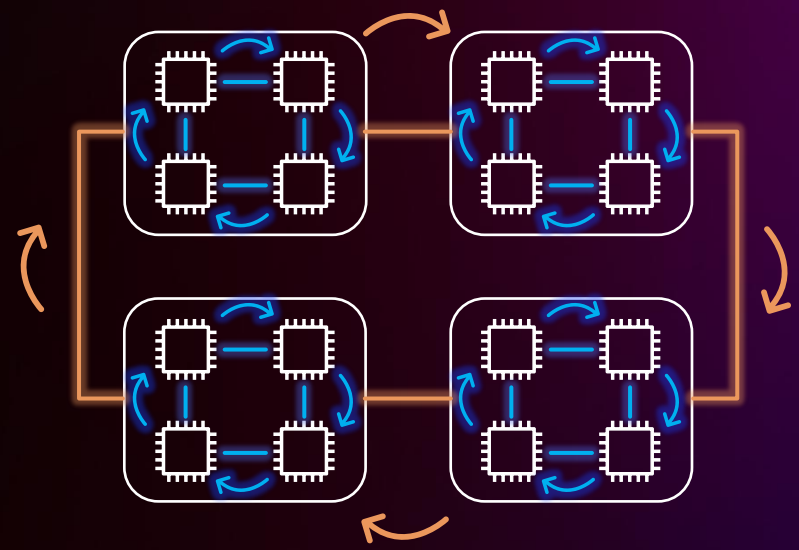


Trn1 – Built for scale-out

PARALLELIZED COMPUTATION AND COMMUNICATION

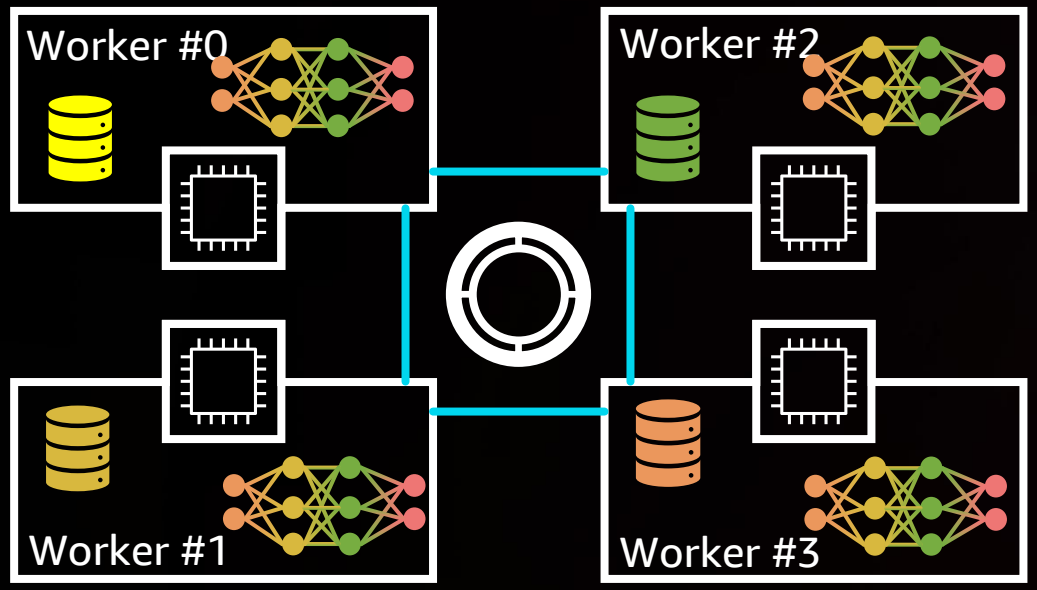


2D-RING TOPOLOGY FOR COLLECTIVE COMMUNICATION

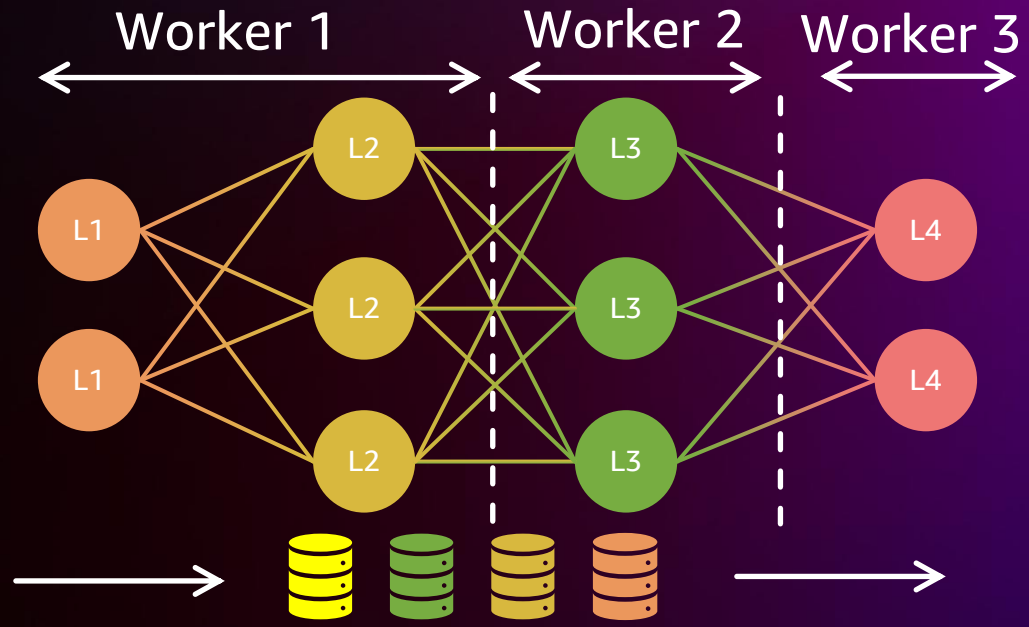


— HIGH SPEED LOCAL INTERCONNECT
— EFAv2

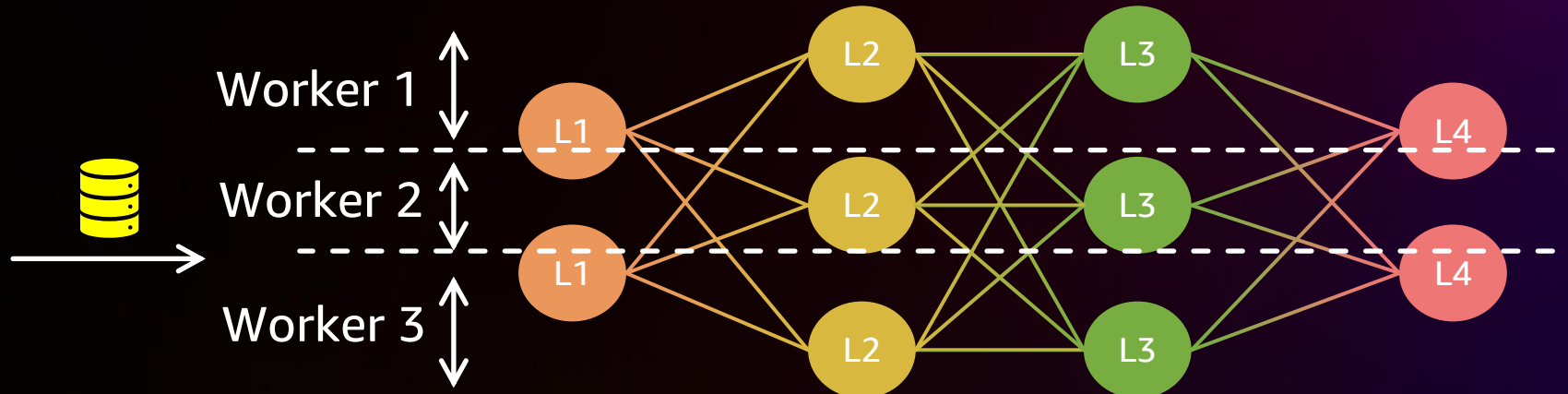
Distributed Training



Data Paralelism



Pipeline Paralelism



Tensor Paralelism

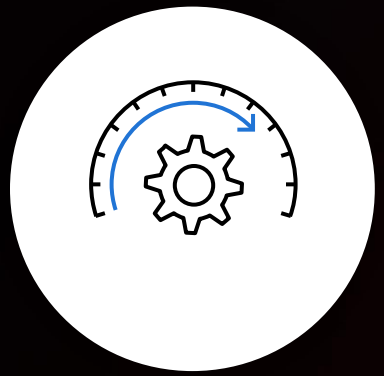
Amazon SageMaker

AMONG THE MOST COMPLETE END-TO-END ML SERVICES

Prepare → Build → Train & tune → Deploy & manage →

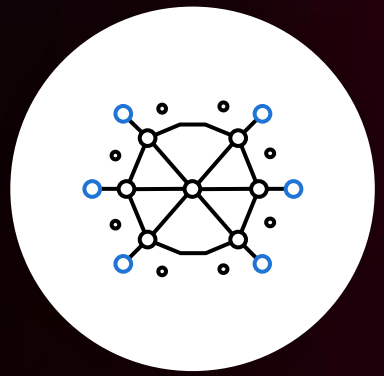
Democratize ML innovation

Empower more groups of people, including business analysts



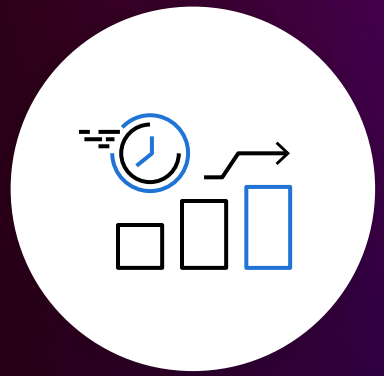
Accelerate the ML lifecycle

Reduce training time from hours to minutes



Prepare data at scale

Access, label, and process structured and unstructured data

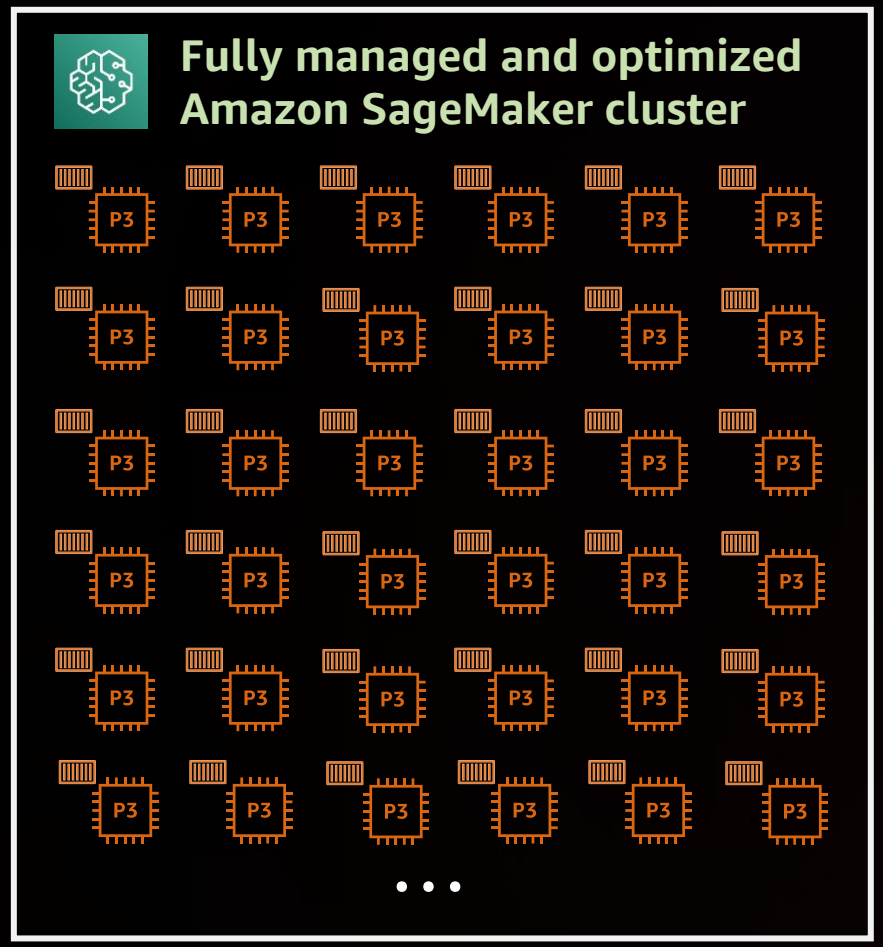


Streamline ML processes

Automate and standardize MLOps processes

https://aws.amazon.com/sagemaker/pricing/

Large training datasets: What are my options?



1. Moderate and large datasets

Amazon S3

- **File mode:** copy entire dataset to local volume
- **Fast file mode:** stream dataset from Amazon S3

2. Scalable shared file system

Amazon EFS

- No downloading or streaming
- Share file system with other services

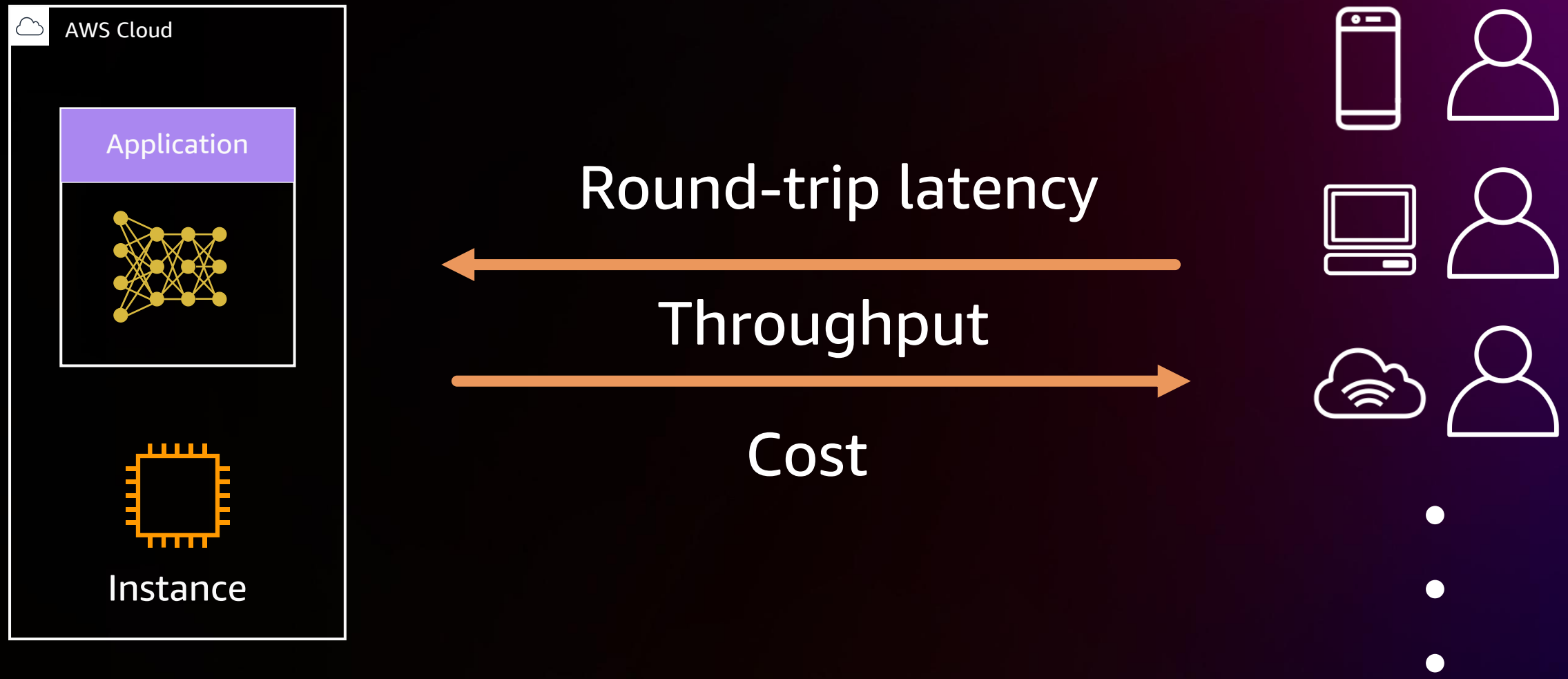
3. High-performance file system

Amazon FSx for Lustre file system

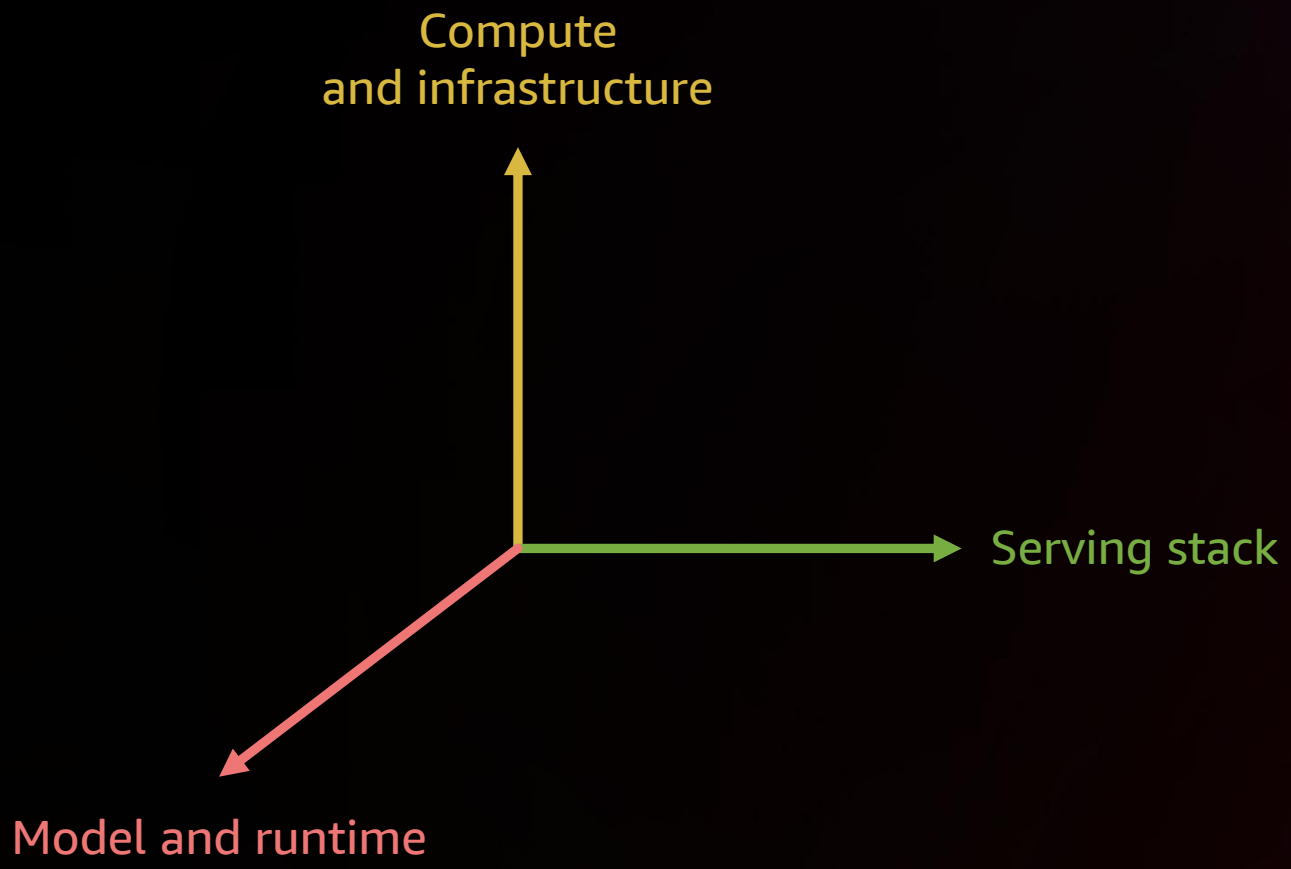
- Optimized for high-performance computing
- Natively integrated with Amazon S3

Inference instances

Inference performance affects customer experience



Optimization Dimensions

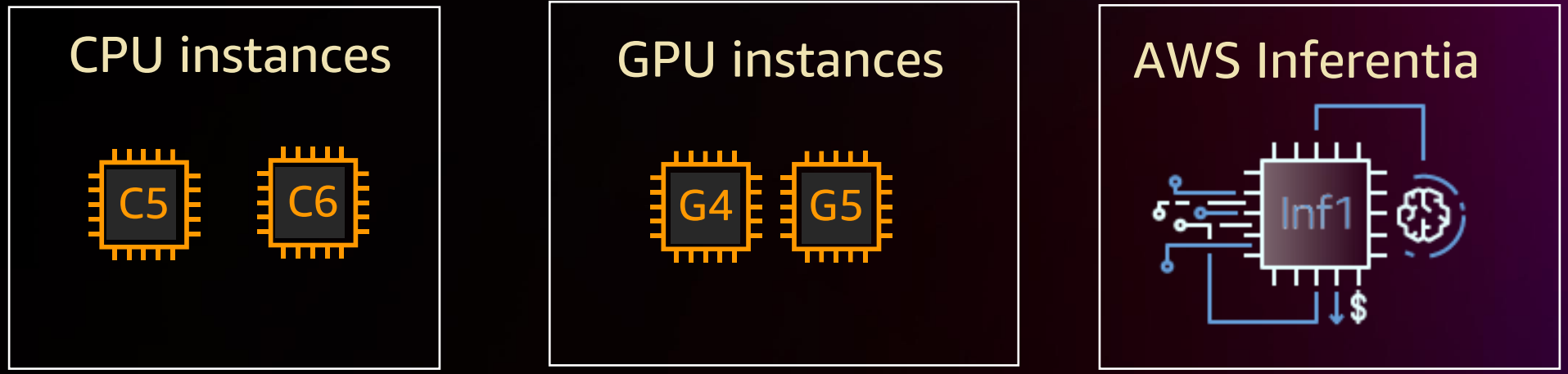


CPU/GPU Instances, custom chips (AWS Inferentia)Networking configuration
SageMaker fully-managed deployment options (multi-model, multi-container, etc.)

Custom stack (e.g. Nginx > Gunicorn > Flask)
TorchServe, TFS, MMS, Nvidia Triton
Configure dynamic batching, # of workers, etc.

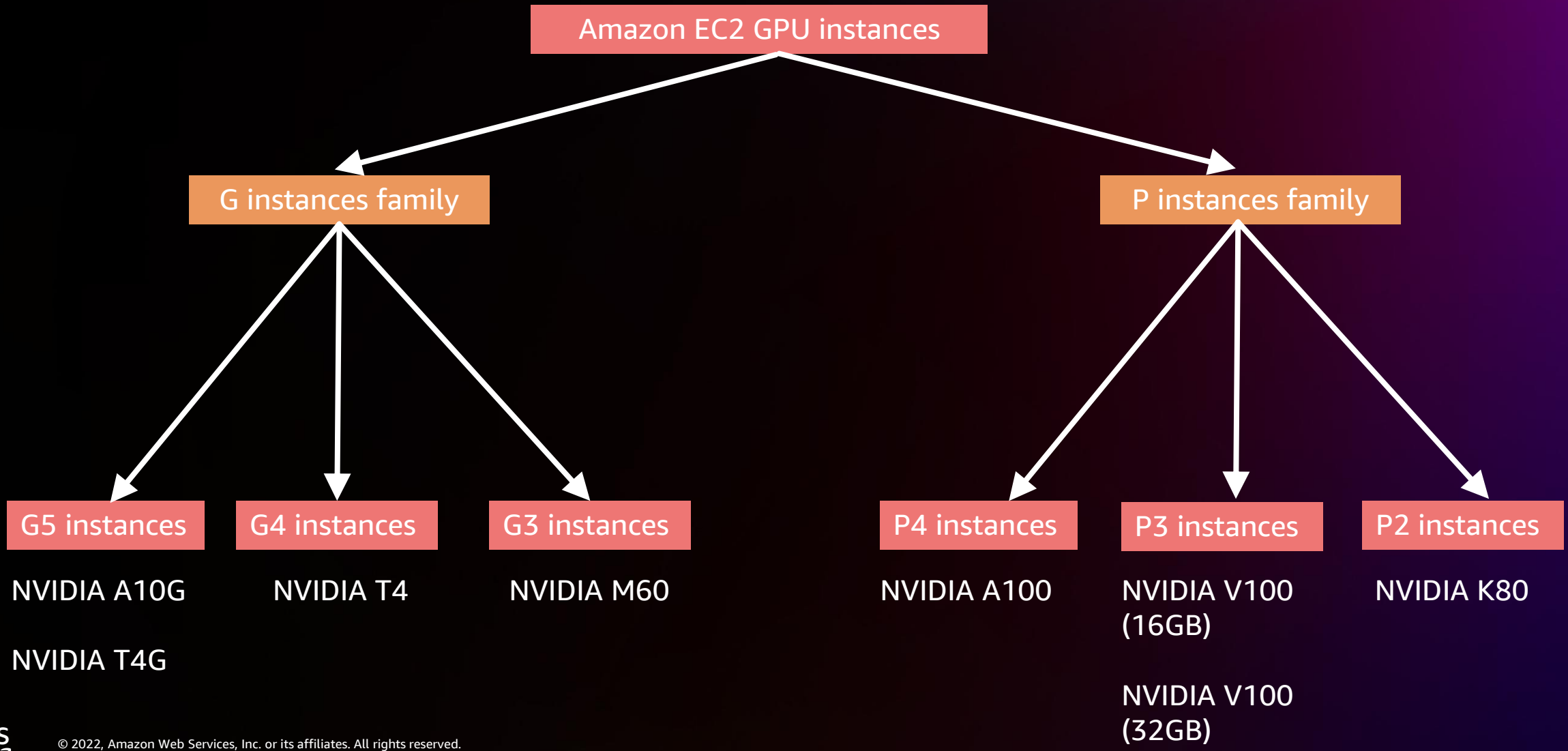
Model compression (pruning, quantization, etc.)
Model compilation (TVM, TreeLite, TensorRT, AWS Neuron, Amazon SageMaker Neo)

Instances and accelerators for ML inference



1. Target performance
2. Cost efficiency
3. Model and framework support
4. Ease of use and deployment

GPU instances for Deep Learning



Benefits of GPU-accelerated inference

Performance

- GPUs are throughput processors and can deliver high throughput at desired latency.
- https://developer.nvidia.com/deep-learning-performance-training-inference

Cost

- GPUs may be under-utilized for small modes, small batch, or sporadic inference request.
- If GPU utilization is low, cost per inference request goes up.

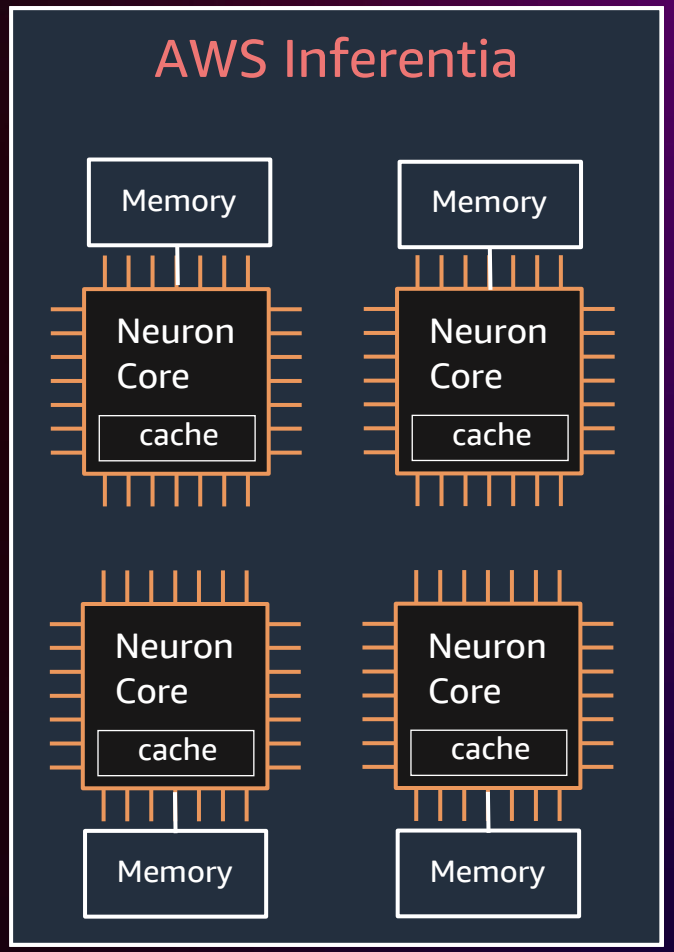
Ease of use

- Native GPU acceleration on all popular deep learning frameworks.
- NVIDIA TensorRT and Amazon SageMaker Neo compilers for easy inference optimization and quantization to FP16, INT8.

AWS Inferentia: Custom silicon for ML inference

FIRST CUSTOM ML CHIP DESIGNED BY AWS

- 4 NeuronCores
- Up to 128 TOPS
- 2-stage memory hierarchy
  Large on-chip cache and commodity DRAM
- Supports FP16, BF16, INT8 data types with mixed precision
- Fast chip-to-chip interconnect



Benefits of AWS Inferentia

Performance

- Amazon Inf1 instances can deliver high throughput and at lower cost compared to GPUs
- Ideal option if your model is supported by AWS Neuron SDK that meets your target latency and throughput goals

Cost

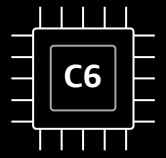
- Inf1 instances delivers lower cost vs GPU for popular models such as YOLOv4, OpenPose, BERT and SSD for TensorFlow, MXNet and PyTorch

Ease of use

- AWS Neuron SDK offers a compiler and runtime as well as profiling tools and support for TensorBoard
- SDK offers higher control over Neuron Cores by implementing Python threads

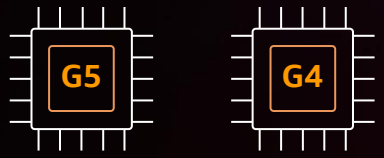
Instances and accelerators for ML inference

CPU INSTANCES



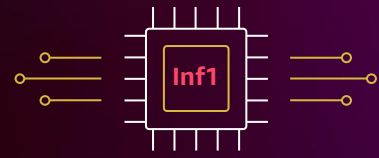
3rd generation Intel Xeon Scalable
Support Intel AVX-512 VNNI

GPU INSTANCES



G5 NVIDIA A10G (24GB) – Up to 8
G4 NVIDIA T4 (16GB) – Up to 4

CUSTOM CHIP

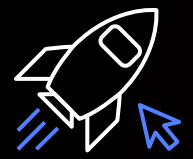


AWS Inferentia chip – Up to 16
Best price/performance for ML inference

Inference accelerator Instance type	Throughput	Latency	Cost efficiency	Model support, Programmability	Ease of use	Framework support
CPU-only C6 instance type	○	○	● Smaller models	●	●	●
GPU G5, G4 instance type	●	●	● High utilization	●	◑	●
AWS Inferentia Inf1 instance type	●	●	●	◑	◑	◑

Amazon SageMaker Inference

SageMaker Real-time Inference



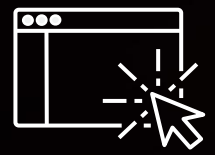
Create a long-running microservice

Instant response for payload up to 6MB

Accessible from an external application

Autoscaling

SageMaker Serverless Inference

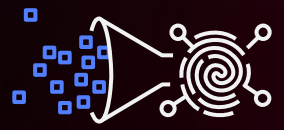


Ideal for unpredictable prediction traffic

Workload tolerable to cold start

Autoscaling (down to 0 instance)

SageMaker Asynchronous Inference



Ideal for large payload up to 1GB

Longer processing timeout up to 15 min

Autoscaling (down to 0 instance)

Suitable for CV/NLP use cases

SageMaker Batch Transform

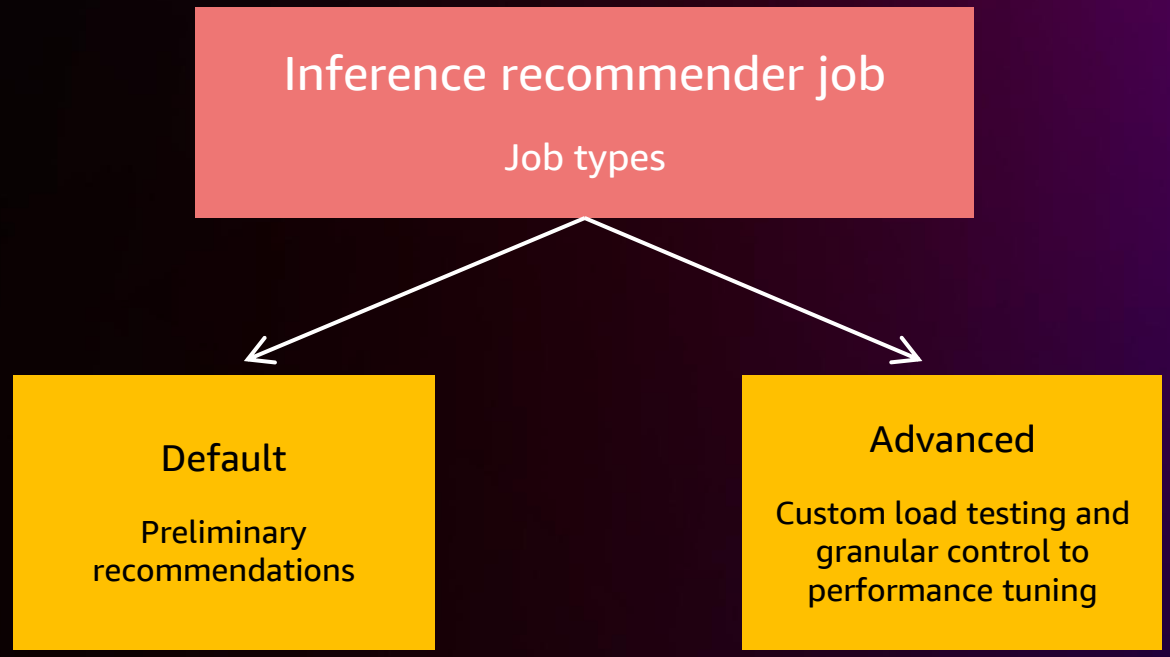
Fully managed mini-batching or large data

Pay only for what you use

Suitable for periodic arrival of large data

Inference Recommender

- Get instance type recommendations (based on throughput, latency, and cost)
- Get parameter tuning suggestions for your model
- Run extensive load test benchmarks
- Integrate with model registry
- Review performance metrics from SageMaker Studio
- Customize your load tests
- Fine-tune your model, model server, and containers
- Get detailed metrics from CloudWatch



Discussion/Q&A

Thank you!

Samir Araujo
arsamir@amazon.nl

Raghu Ramesha
ragmesh@amazon.com

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