Building and Scaling Inference Workloads on Amazon EKS

Save this link for later

genai.eksworkshop.com

If you want to revisit any of the workshop content later, you can access it here

AWS ecosystem for AI/ML workloads

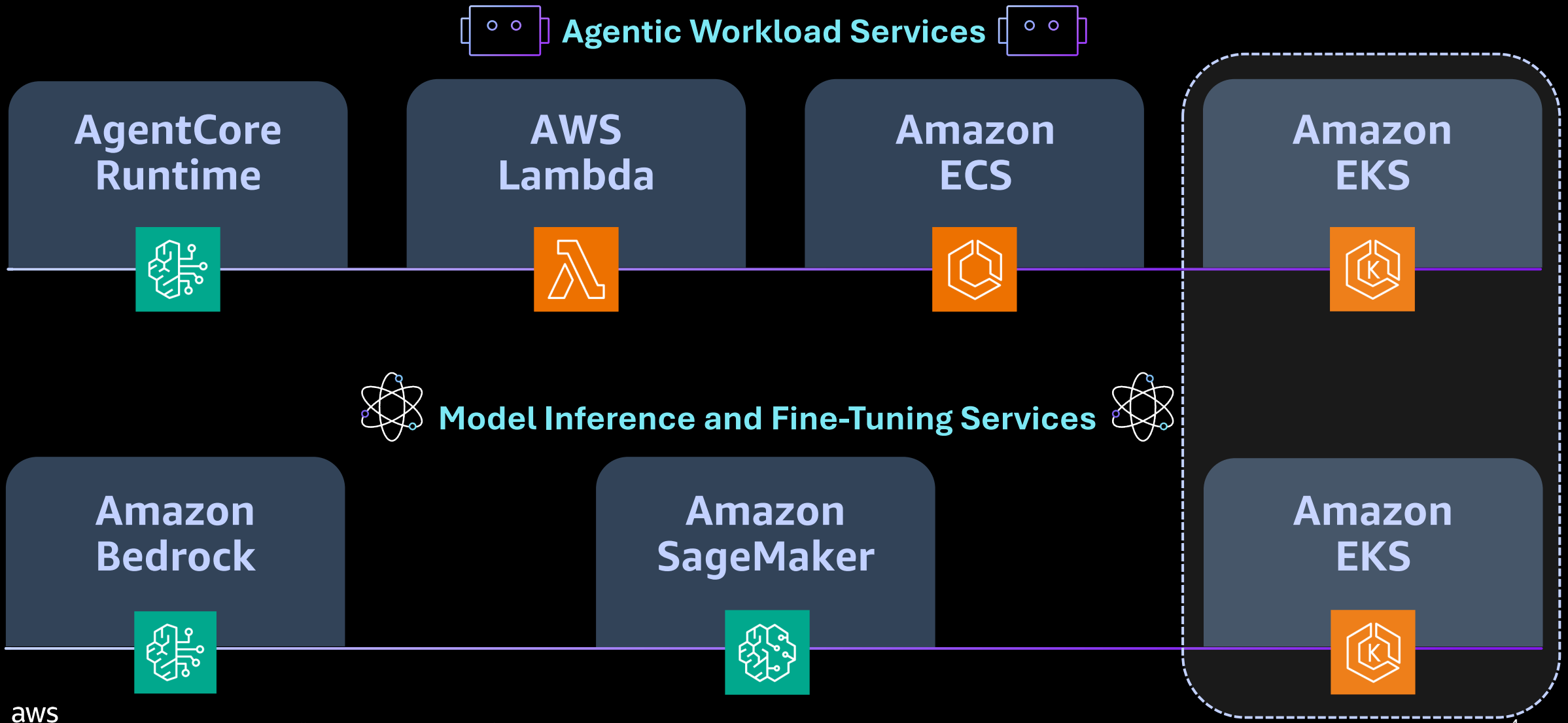
Agentic Workload Services

- AgentCore Runtime
- AWS Lambda
- Amazon ECS
- Amazon EKS

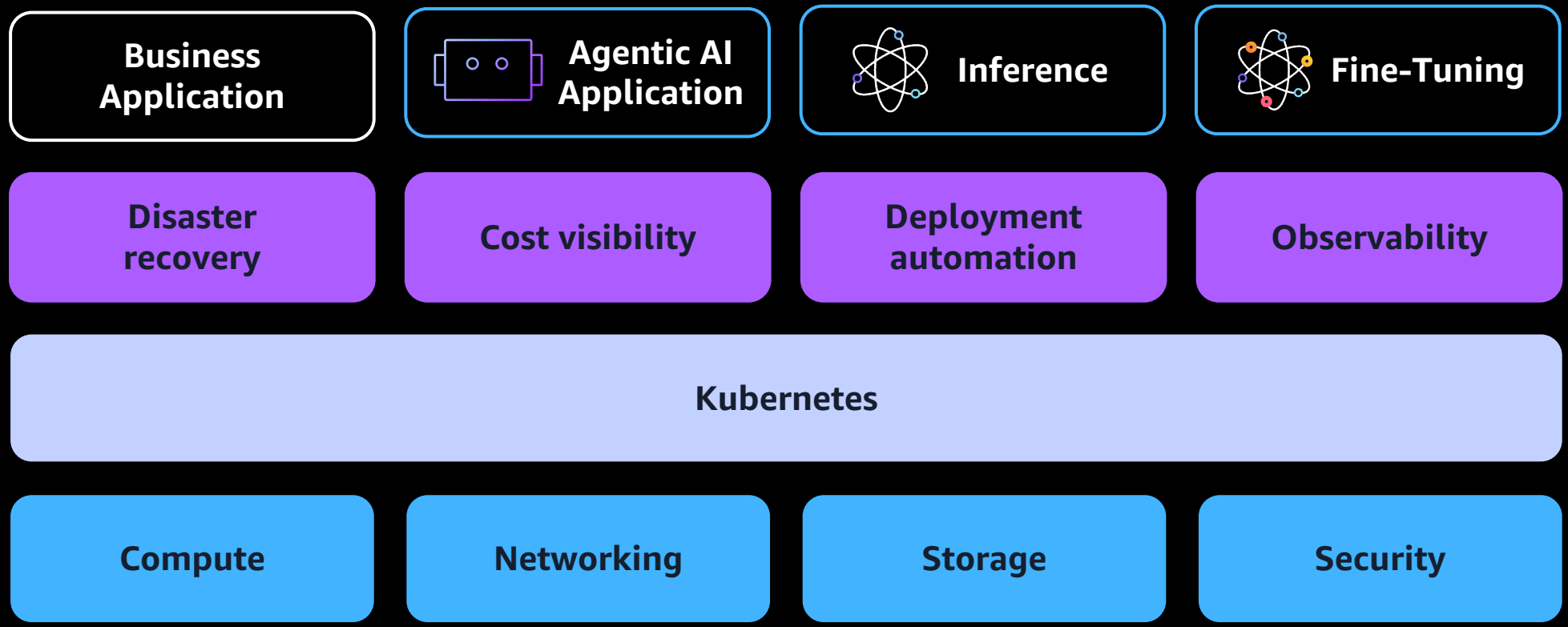
Model Inference and Fine-Tuning Services

- Amazon Bedrock
- Amazon SageMaker
- Amazon EKS

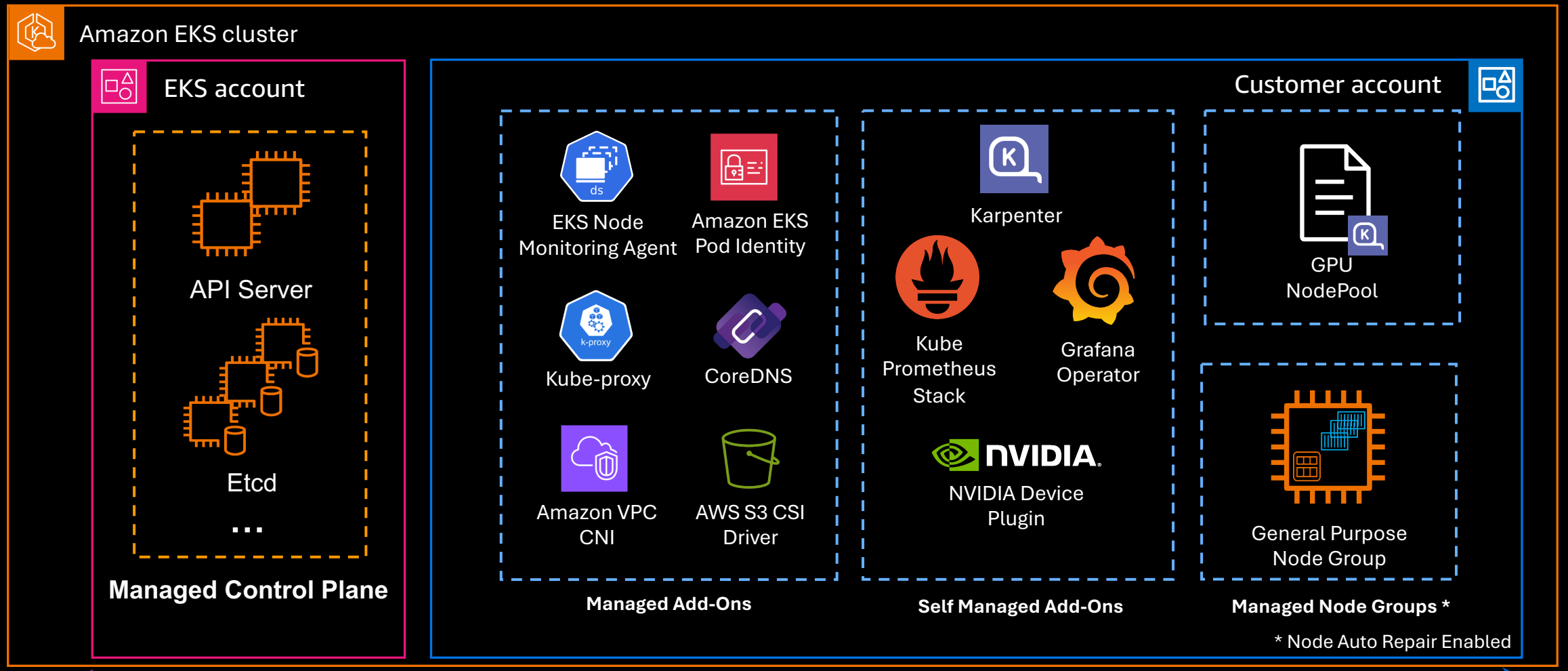
AWS ecosystem for AI/ML workloads



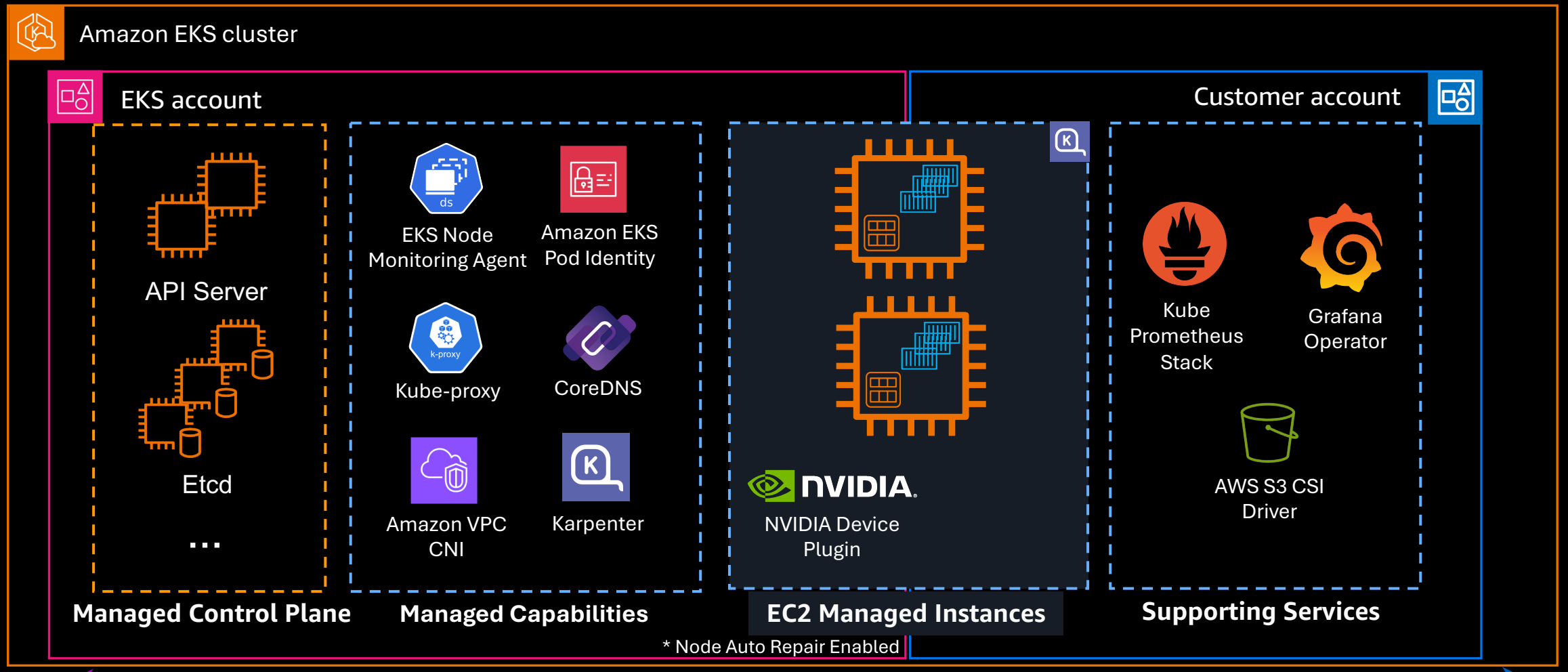
Standardizing on Kubernetes



Amazon EKS Standard



Amazon EKS Auto Mode



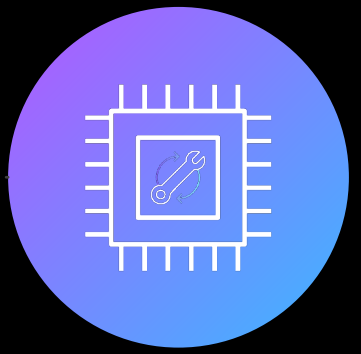
Managed

Self Managed

Node health monitoring and auto-repair for GPU

Supports both GPU and Neuron | Announced December 16, 2024

The Node monitoring agent will **automatically detect** health issues



EKS will **automatically replace** the faulty instance with a healthy one after **10 minutes**

Fast container pulls

Seekable OCI (SOCI) Parallel Pull

Concurrent container downloads over network and unpack on-disk, utilizing available infrastructure capacity

Faster scaling for AI workloads

Training and inference workloads often use large container images for models, artifacts, and dependencies - up to 60% faster container pulls

No changes required

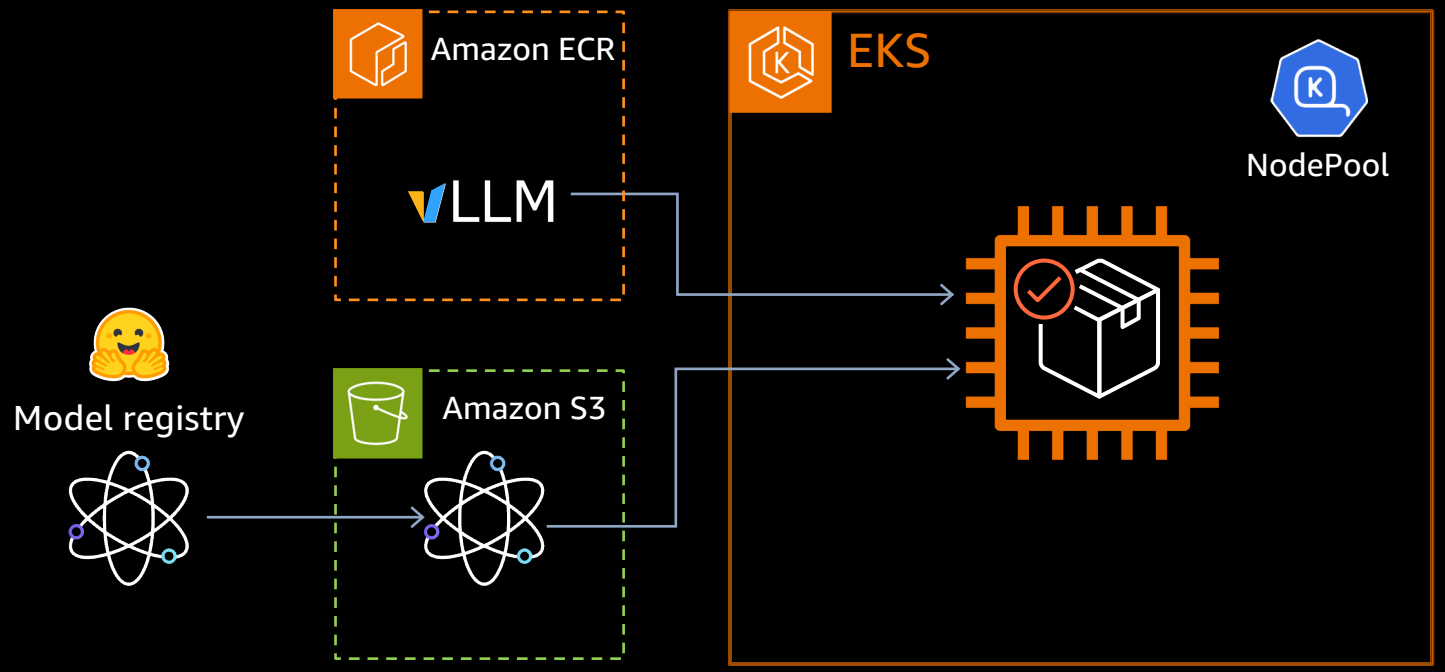
SOCI works with all existing Open Containers Initiative (OCI) container images and without any changes to infrastructure or storage

Batteries included

SOCI is enabled by default when using NVIDIA (G, P) or Trainium instances with EKS Auto Mode for faster startup and scaling by default

Inference on EKS
Building blocks

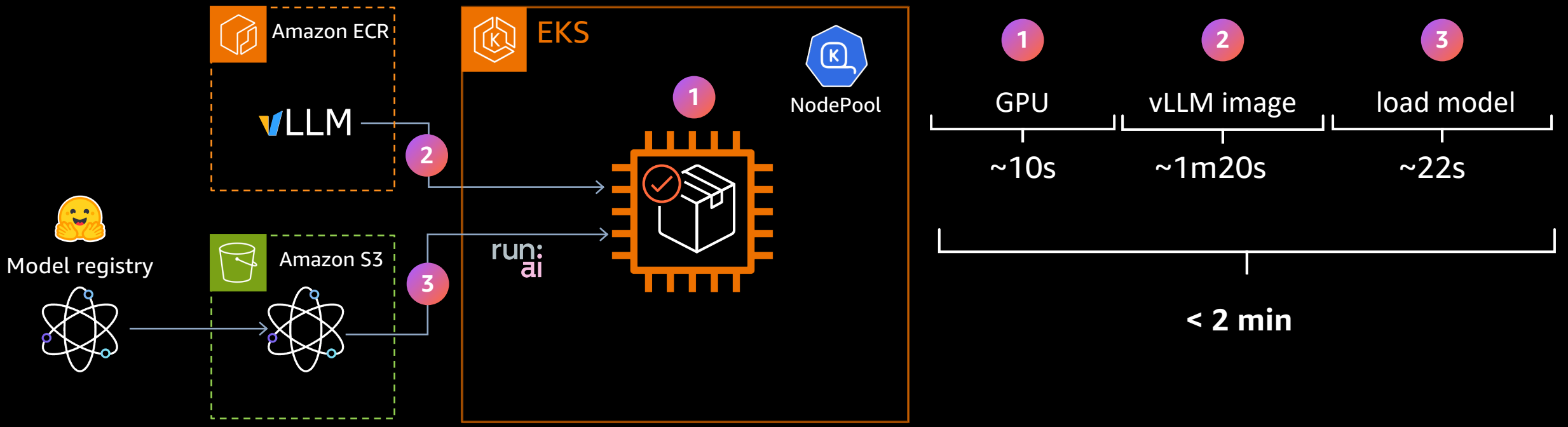
Run workload



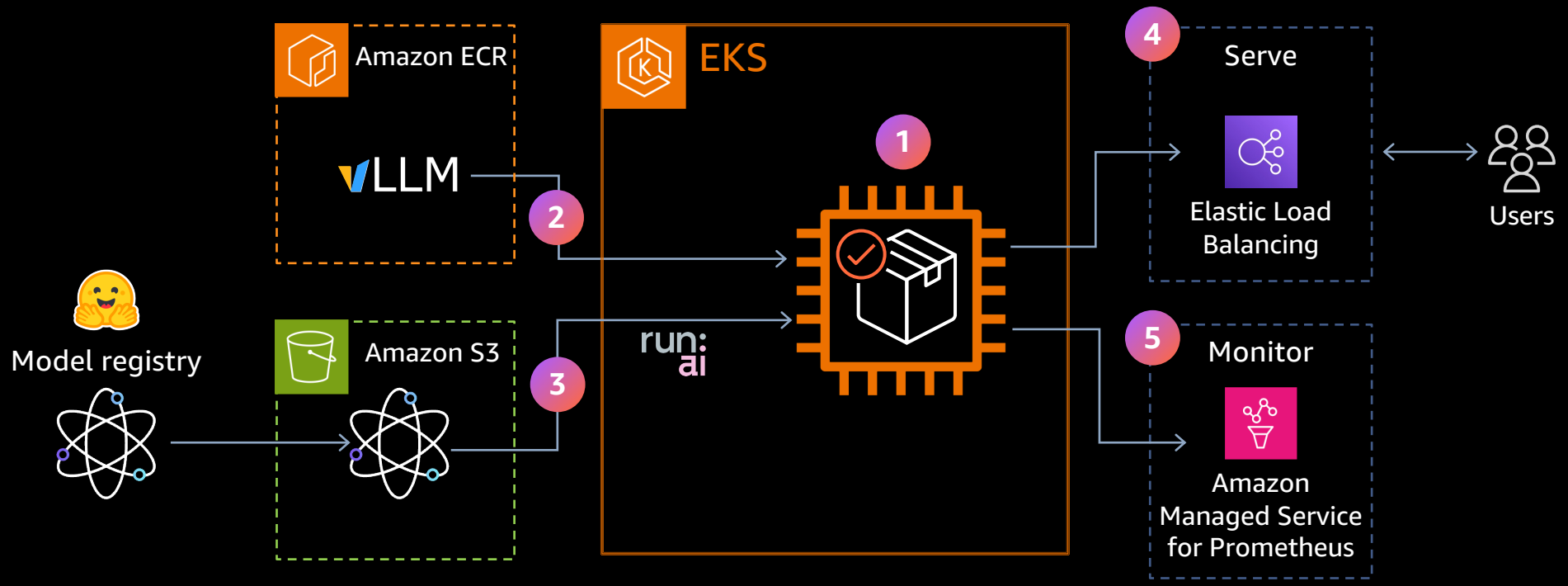

```yaml
apiVersion: apps/v1
kind: Deployment
...
spec:
  template:
    spec:
      containers:
      - name: server
        image: public-ecr/vllm:latest
        args:
          - --model /s3/ministral3-8b
          - port "8080"
        env:
        - name: VLLM_ATTENTION_BACKEND
          value: "TRITON_ATTN_VLLM_V1"
...
        nodeSelector:
          intent: gpu
        tolerations:
        - key: nvidia.com/gpu
          operator: Equal
          value: "true"
          effect: NoSchedule
```

```
> kubectl apply -f deployment.yaml
```

Architecture overview

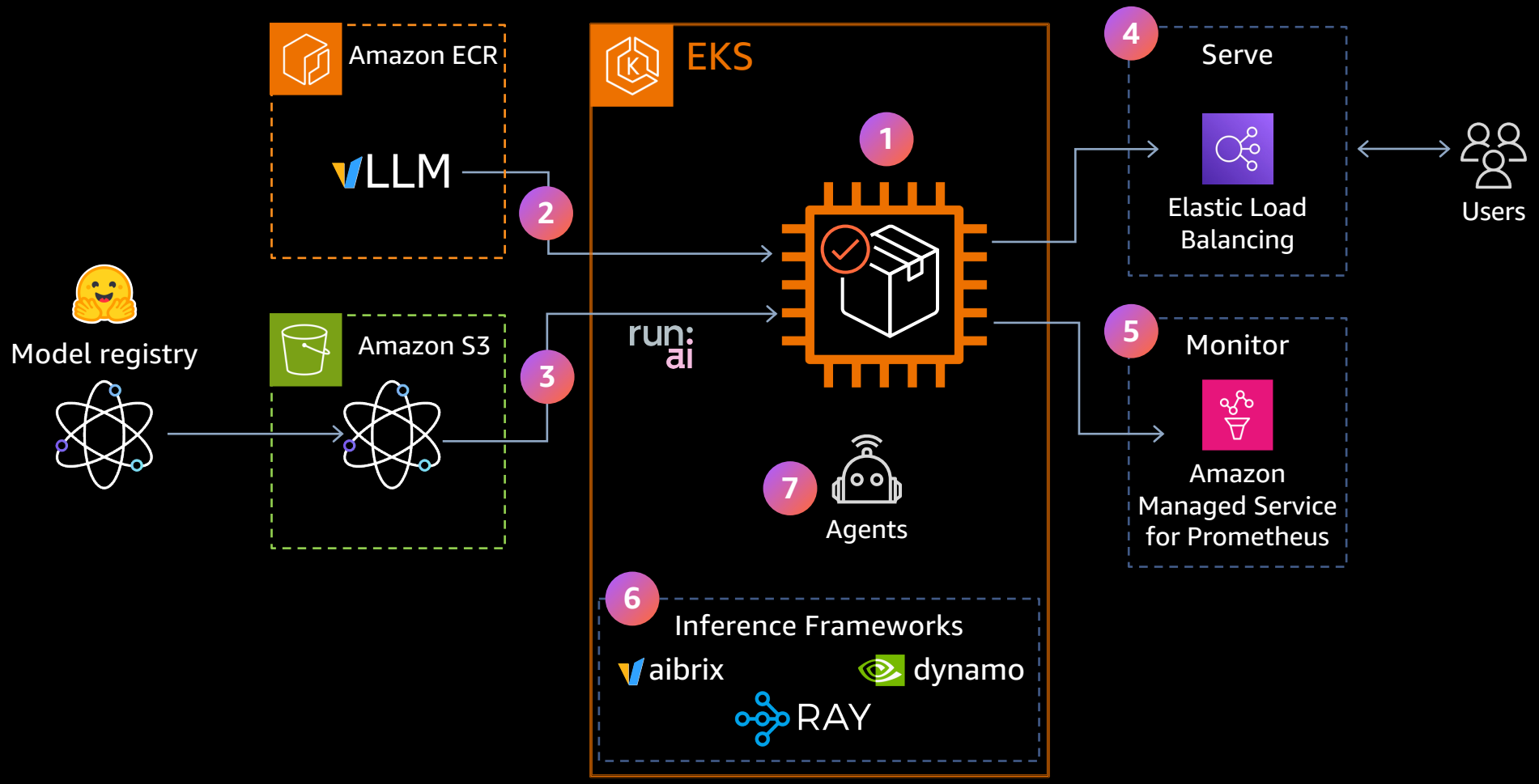


Architecture overview



1. Provision compute
2. Run Container
3. Load Model in GPU Memory
4. Expose an Endpoint and Interact
5. Monitor

Architecture overview



1. Provision compute
2. Run Container
3. Load Model in GPU Memory
4. Expose an Endpoint and Interact
5. Monitor
6. Inference Framework
7. Agentic workloads

Next Steps after the workshop

Continue in your own account

- Use the terraform to spin up the infrastructure in your own account
- Follow our best practice guide for AI/ML workloads
- Visit AI on EKS for more advanced blueprints and patterns
- All links accessible from **genai.eksworkshop.com**

Get support from AWS GenAI on EKS Specialists

- Have a 1:1 with a Solution Architect
- Ask to run this workshop for your team

Get support from your account team

- Ask for PoC credits
- Help with GPU pricing or capacity
- Find a partner to help you build

Thank you!