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

CMP210

Modern applications: An infrastructure perspective

Yael Grossman

Sr. Solutions Architect, Flexible Compute
AWS

Jeremy Atlan

Ground Truth Infrastructure Architect
Mobileye

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Modernization  Optimization

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Agenda

Tenets of a modern application

Selecting, provisioning, and scaling Amazon EC2 capacity

Architectural practices for modern infrastructure

Performant, scalable, and cost-efficient data applications at Mobileye

Key takeaways

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

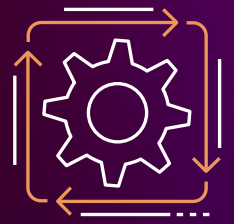
Tenets of modern applications



Responds in milliseconds



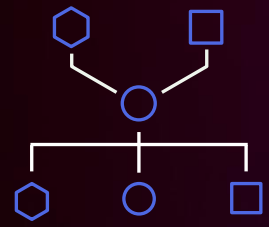
Scales to millions of users



Frequent changes



Fault tolerant



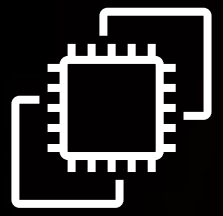
Modular components

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Modern applications need flexible infrastructure

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Different options to run modern applications on AWS



Amazon EC2

Virtual server instances in the cloud



Amazon ECS, Amazon EKS

Container management service for running Docker on a managed Amazon EC2 cluster



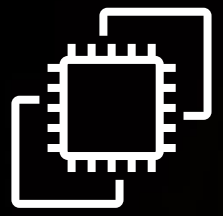
AWS Lambda

Serverless compute for running stateless code in response to triggers

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Different options to run modern applications on AWS



Amazon EC2

Virtual server instances in the cloud



Amazon ECS, Amazon EKS

Container management service for running Docker on a managed Amazon EC2 cluster



AWS Lambda

Serverless compute for running stateless code in response to triggers

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

A compute platform for virtually every workload

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Work backwards from your compute requirements

550+

Instance types

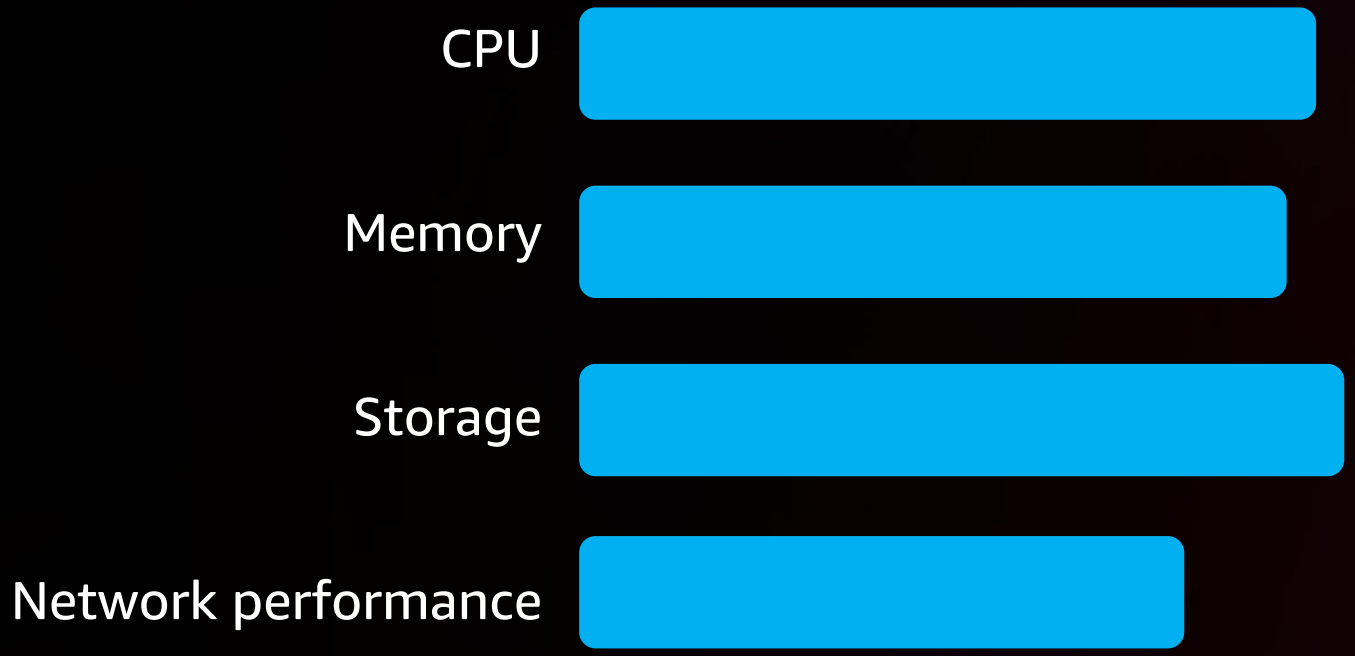
for virtually every workload and business need

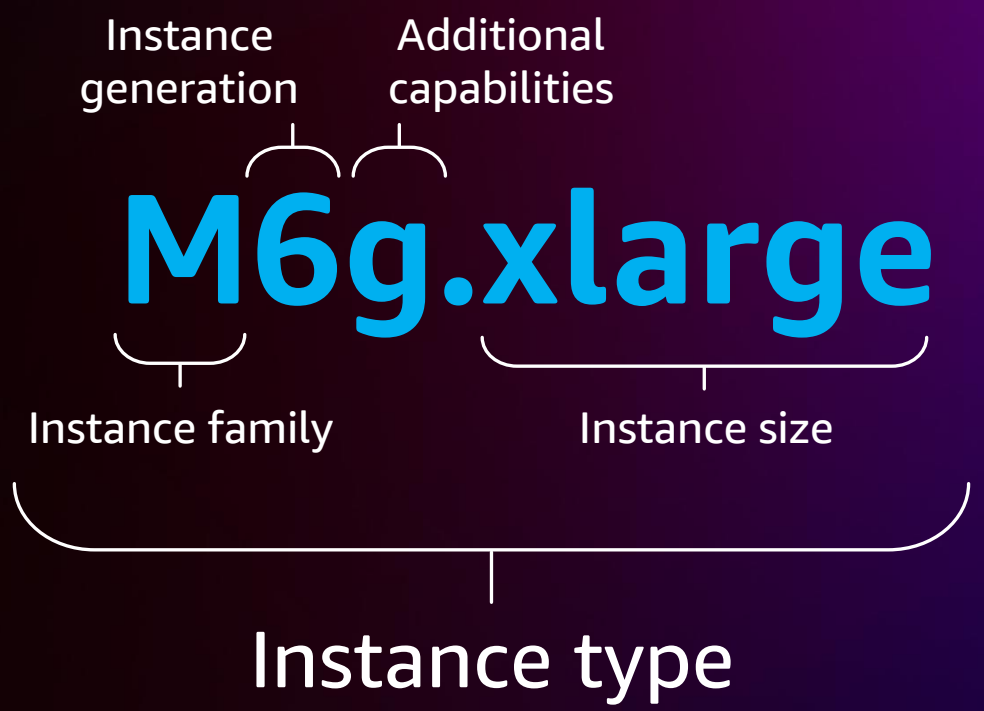
1. What processors can my workload use?
2. What are my workload's performance requirements?
3. What is my workload's consumption pattern?

→ A flexible list of instance types that fit **your workload**

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon EC2 instance characteristics





© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

The latest choice of **processors**



and **accelerators**



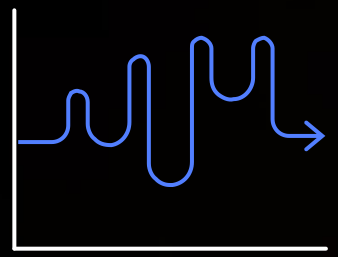
aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multiple purchase options to optimize costs

On-demand

Pay-for-what you use with no long-term commitments



Stateful spiky workloads

Savings plans

Up to **72%** savings for 1 or 3 year hourly usage commitments



Committed and steady-state usage

Spot

Spare capacity at up to **90%** off On-Demand prices



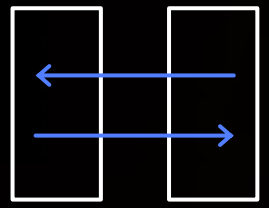
Fault-tolerant, flexible, stateless workloads

The best practice is to combine all three purchase options

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Spot Instances for interruptible workloads



Same infrastructure

Spare Amazon EC2 capacity from the same infrastructure as On-Demand



Capacity

AWS can reclaim with a 2-minute notice; interruptions happen when Amazon EC2 needs the spare capacity back

Workloads Spot is ideal for



Fault-tolerant



Flexible



Loosely coupled



Stateless

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Infrastructure components

One size does not fit all, but usually multiple do

Compute options

Instance categories
(General purpose, compute-intensive, memory-intensive, etc.)

Size
(Nano to 32xlarge)

Choice of processor
(AWS, Intel, AMD)

Accelerated computing
(GPUs and FPGA)

Instance storage
(HDD and NVMe)

Networking
(up to 100 Gbps)

Bare metal

Purchase options
(On-Demand, Saving Plans, Spot)

+

Launch services

Auto Scaling groups

Amazon EC2 fleet – instant mode

Allocation strategies

Attribute-based selection

+

Scaling behavior

Amazon EKS managed node groups and Kubernetes Cluster Autoscaler

Karpenter

AWS Fargate

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Provisioning and scaling Amazon EC2 capacity



Auto Scaling groups





Amazon EKS managed node groups

and

Kubernetes Cluster Autoscaler



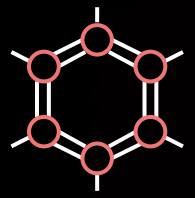
Karpenter

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Auto Scaling groups

Contains a collection of Amazon EC2 instances that are treated as a logical grouping for the purposes of automatic scaling and management

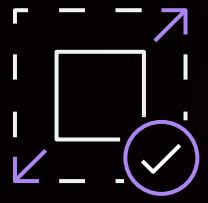


Size

Launch instances to meet desired capacity

Periodic health checks on instances

Maintain a fixed number of instances



Scaling policy

Manual, scheduled, predictive, on-demand

Dynamically change size of group to meet changing conditions

Size is determined by min./max. capacity



Purchase/Arch.

On-Demand, Savings Plans, and Amazon EC2 Spot Instances

x86, arm64, or mixed architecture group



Integrations

Lifecycle hooks

Instance health checks

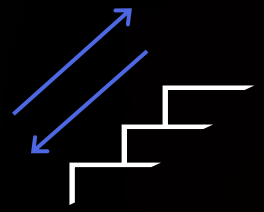
Load balancer integration

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Scaling policies with Amazon EC2 Auto Scaling

Dynamic scaling

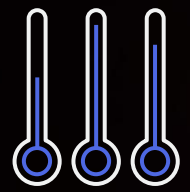


Simple/step scaling

Monitors metrics and adds/removes instances as per steps defined by the customers

Manually calculate capacity

Reactive in nature



Target tracking

Thermostat-like control mechanism that automatically adds or removes instances to maintain metrics at a customer-defined target

Automated

Reactive in nature

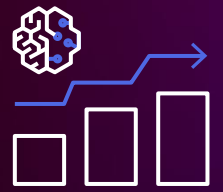


Scheduled scaling

Launch/terminate instances as defined by customer on a schedule

Manually calculate capacity

Proactive in nature



Predictive scaling

Proactively launch capacity based on historic trends

Automated

Proactive in nature

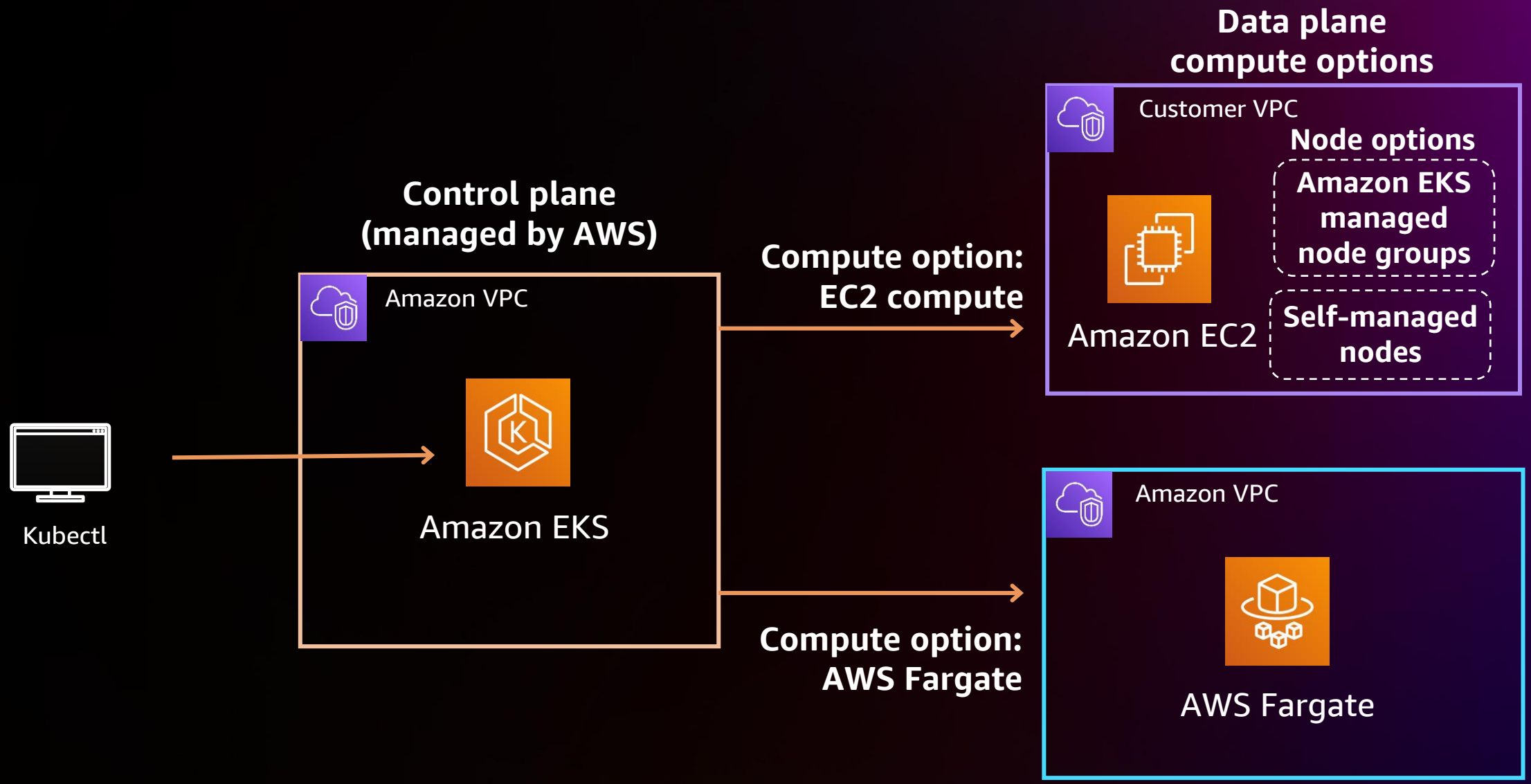
aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

In a Kubernetes world ...

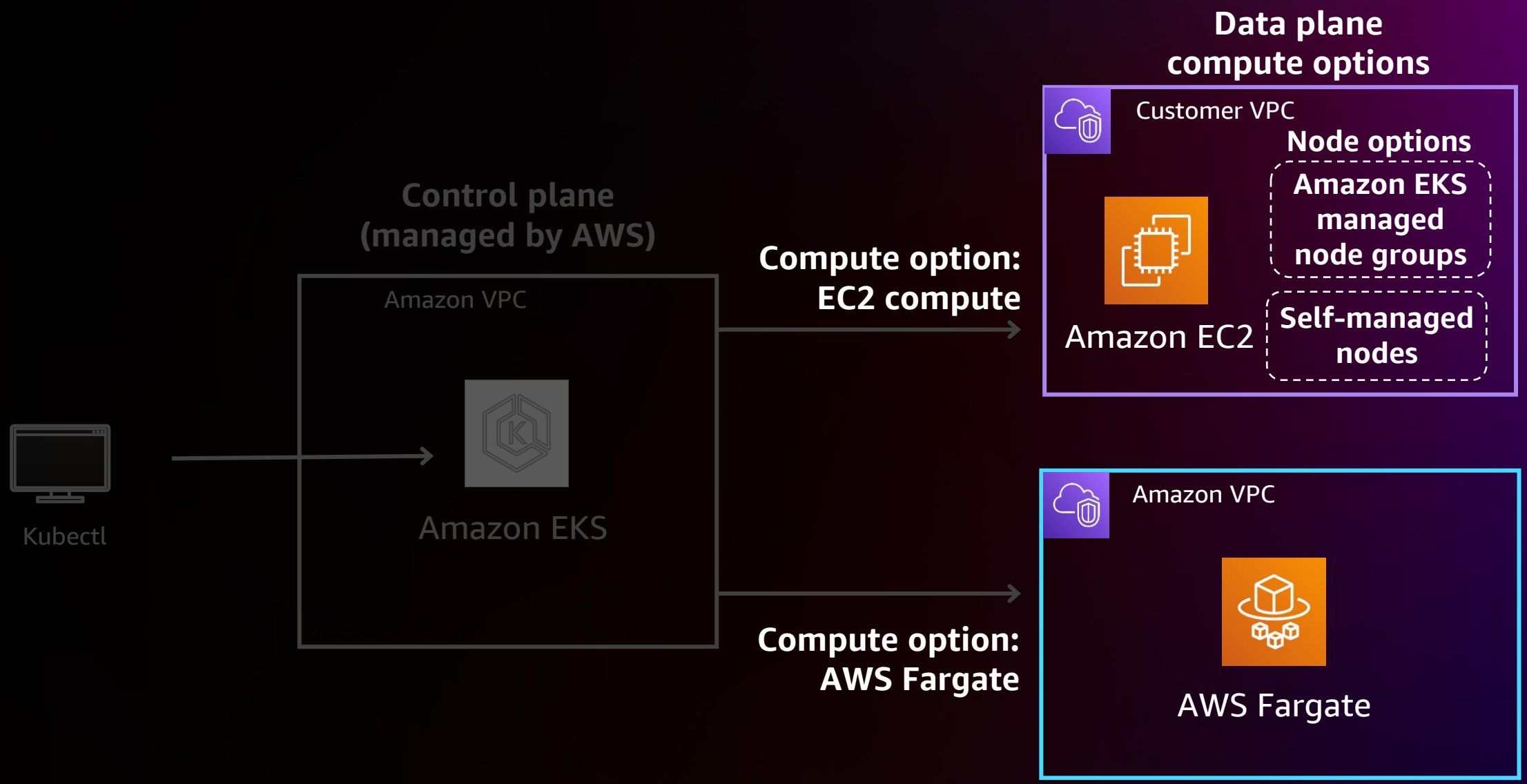
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Compute options for Amazon EKS



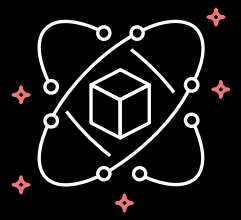
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Compute options for Amazon EKS



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

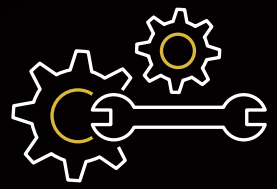
Amazon EKS managed node groups



Integrated with Amazon EC2 Auto Scaling

Support for **mixed instance types**

Amazon EC2 Auto Scaling group provisioning with **capacity-optimized** allocation strategy



High availability included

Managed **node draining** during node termination and updates



Scaling managed by Cluster Autoscaler

Managed Spot Instance draining integrated with **Amazon EC2 Auto Scaling Capacity Rebalancing**



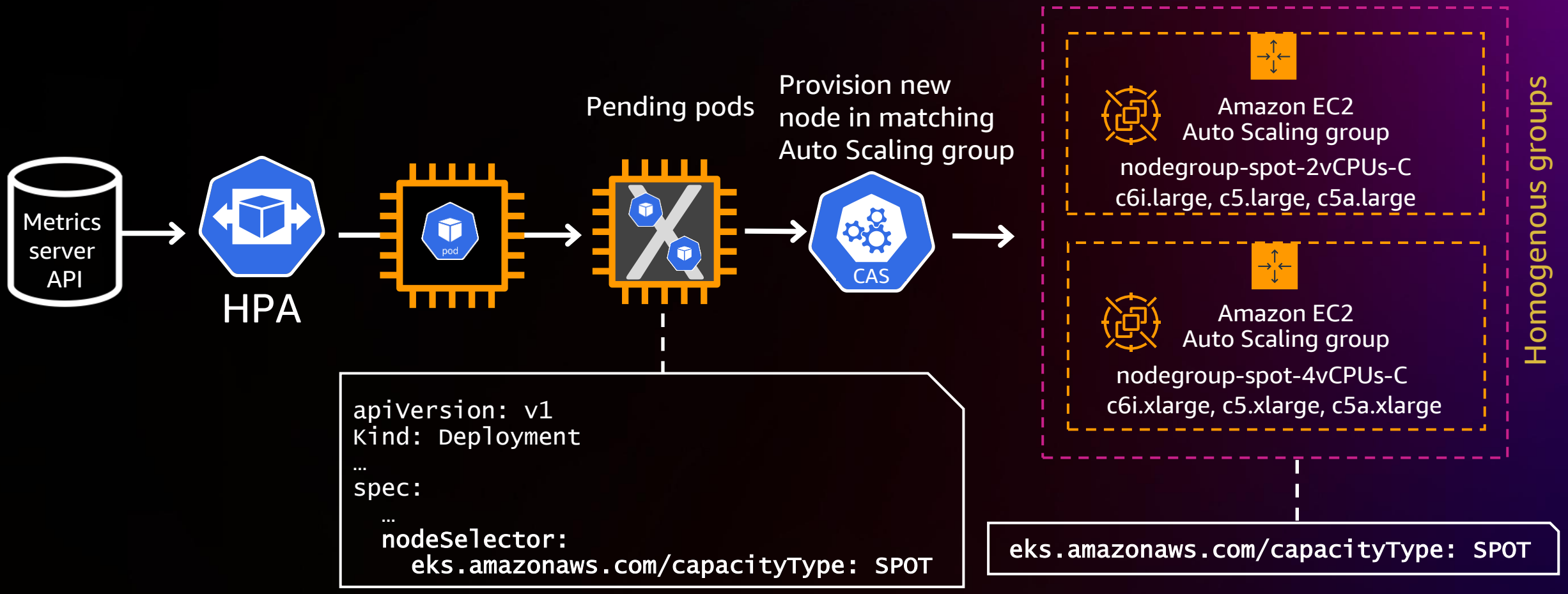
Up to date with Amazon EKS optimized AMI

Custom configuration available with custom AMIs

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Kubernetes Cluster Autoscaler



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Kubernetes Cluster Autoscaler

Scales worker nodes within any specified Auto Scaling groups

- Add nodes to Auto Scaling groups when pods fail to launch due to lack of resources
- Reschedule pods and evict underutilized nodes
- Bridges Kubernetes abstractions into AWS abstractions (Auto Scaling groups and Amazon EKS managed node groups)

Considerations

Uses available RAM and CPU on instances for its scaling calculations

- Each node group should have instance types with the same amount of RAM and CPU and the same labels and taints

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What is Karpenter?

Karpenter is an **open-source**, **flexible**, and **high-performance** Kubernetes Cluster Autoscaler



Open source and native to Kubernetes



Dynamic, groupless node provisioning



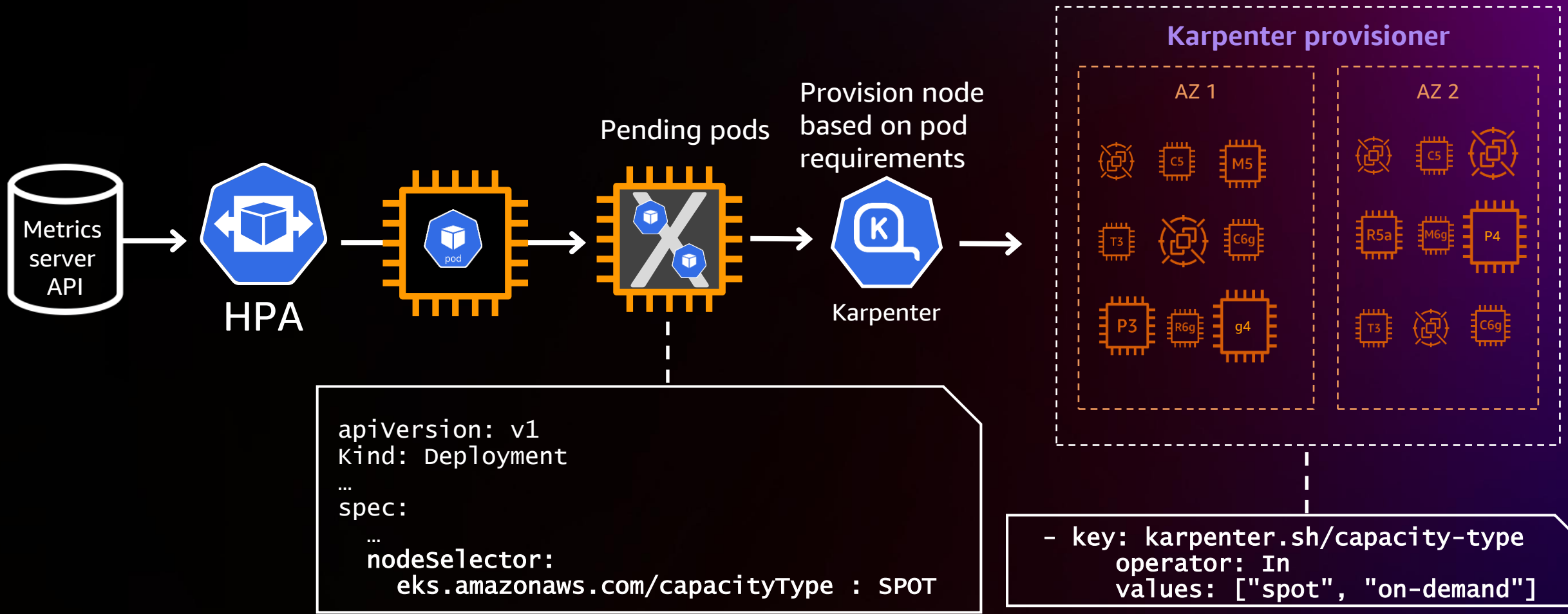
Automatic node sizing



Rapid scaling

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Karpenter for groupless provisioning

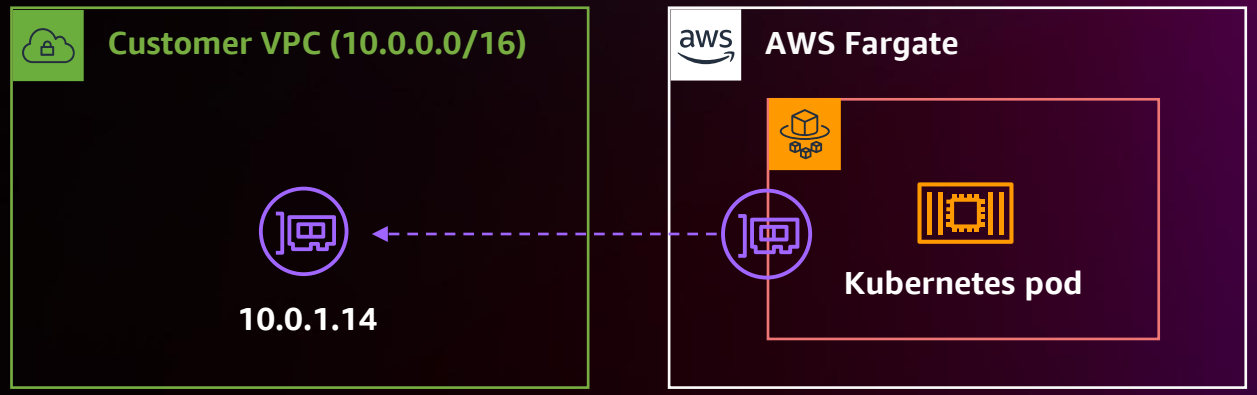


© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Fargate – Serverless compute



AWS Fargate



- Rightsized instances–based CPU and memory requests within the pod specification
- AWS manages the patching and upgrading of the underlying infrastructure; you manage the container
- Secure isolation of tasks by running a dedicated kernel without shared CPU, memory, or storage

Considerations

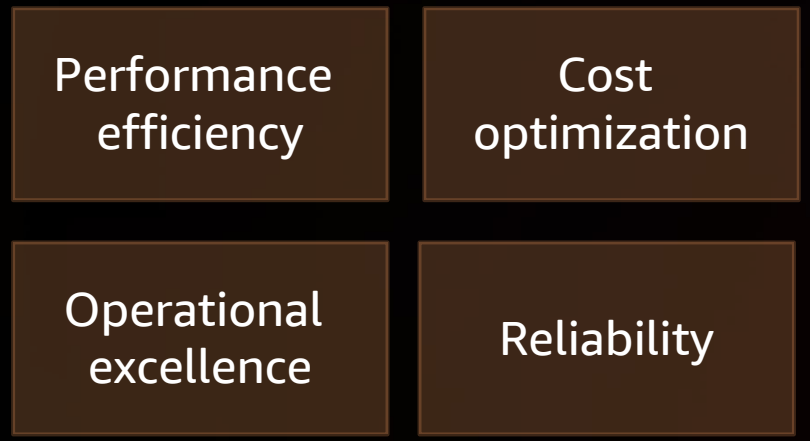
- Pods are limited to private subnets
- DaemonSets are not supported
- Privileged containers are not supported

aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Architectural practices for modern infrastructure

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Compute flexibility



- Take advantage of cost-efficient compute
- Run diverse types of workloads with different requirements in a single cluster
- Support adding new workloads
- Prepare for scenarios where lack of capacity on your ideal instance type could be an issue

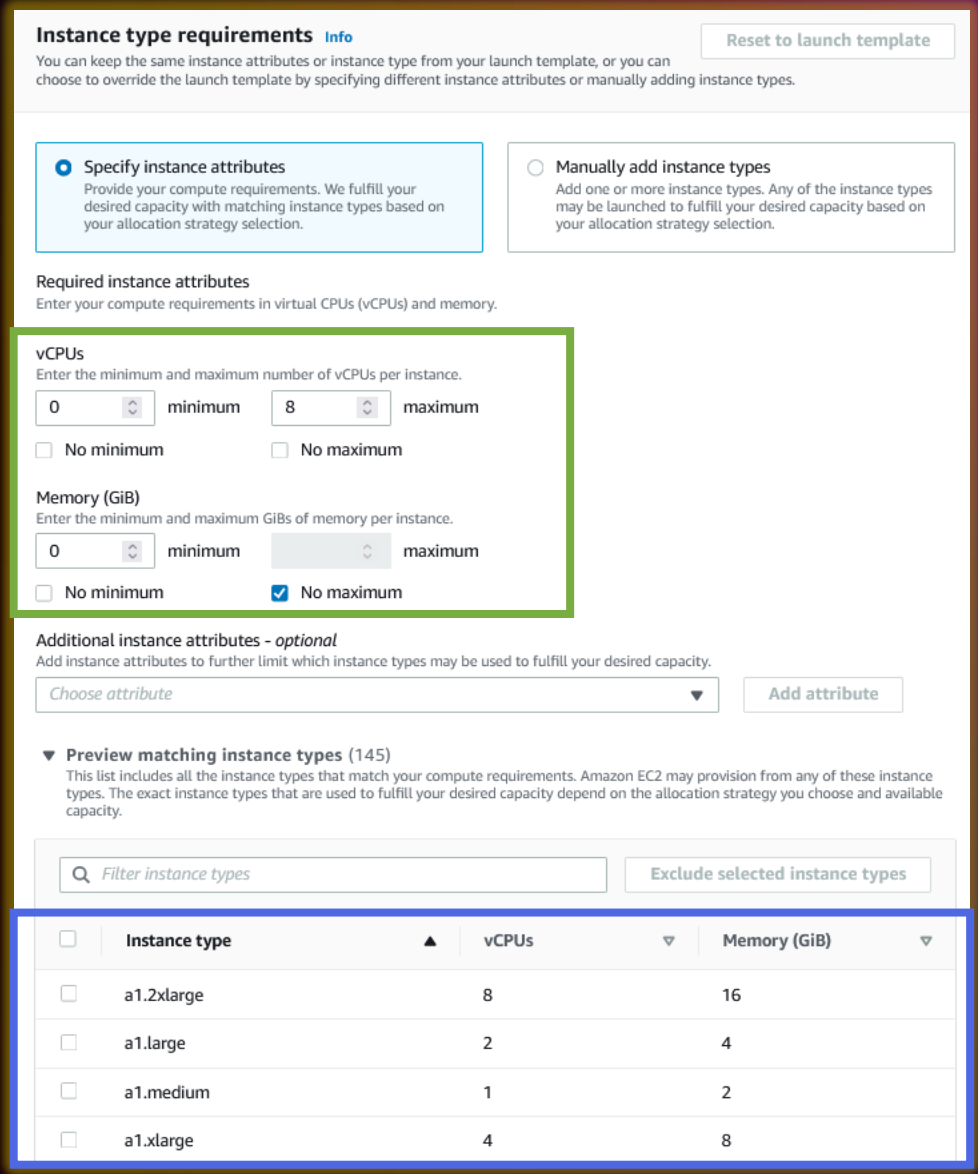
aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Take an attribute-based approach

- Provision capacity based on instance attributes depending on application requirements
- Supported by Auto Scaling groups, managed node groups, Terraform, and more

Benefits

- Simplified provisioning
- Instance flexibility
- Price protected
- Future proof



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Deploy across Availability Zones

- Achieve high scalability, high availability, and high fault tolerance in case a major failure occurs in an entire Availability Zone
- Use K8s native spreading through topology spread constraints
- Consider additional costs due to cross-zone communication



aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Instance lifecycle management

Operational excellence

Reliability

- Respond to instance health checks status and underlying hardware failures
- Respond to changes that affect the availability of Spot Instances
- Support for infrastructure updates through automatic deployments

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Choosing an Auto Scaling approach



Amazon EC2 Auto Scaling



Managed node groups and Cluster Autoscaler



Karpenter



Fargate

Highly flexible Amazon EC2 instance lifecycle management and scaling

Provisioning and scaling for Amazon EKS worker nodes via an integration with Auto Scaling groups

Fully automated, flexible node management for Kubernetes clusters

Serverless compute

Focus on Karpenter for high-scale, spiky, or diverse Kubernetes workloads

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Performant, scalable, cost-efficient data applications at Mobileye

Jeremy Atlan

Ground Truth Infrastructure Architect
Mobileye

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Agenda

Who we are

Mobileye's diverse data applications on AWS

Multi-solution architecture



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Mobileye: Who we are







- Leader in **Collision Avoidance** & **Autonomous Driving** Technology
- Over **300** Car Models with Mobileye Inside
- Over **100M** Vehicles with Mobileye Technology

aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AI Algorithms at Mobileye

- Identify all **road artifacts** and **road users**
- Predict the motion of all **road users**
- Determine a safe and comfortable **driving policy**



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data is a key ingredient of algo development

Types of data

- RGB images
- Raw data collected by sensors
- Maps
- Ground truth data

What is it used for

- Auto-label DataSet for ML
- Train ML Models
- Validation DataSet for KPI
- and much more

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

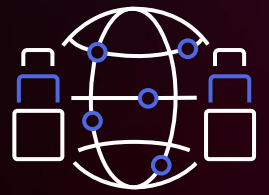
Data applications – One size does not fit all

Application needs vary

- Duration
- Dependencies
- Scale
- Resource requirements
- SLA

The challenge

What is the best way to run each application?



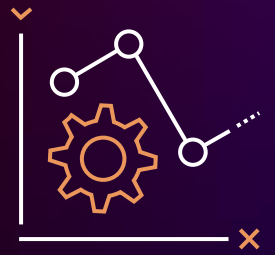
Reliability



Cost optimization



Security



Performance efficiency

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

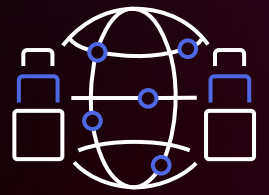
Data applications – One size does not fit all

Application needs vary

- Duration
- Dependencies
- Scale
- Resource requirements
- SLA

The challenge

What is the best way to run each application?



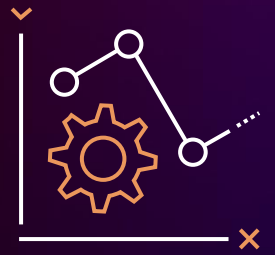
Reliability



Cost optimization



Security



Performance efficiency

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



#1 use case – ML model training on Amazon SageMaker

- Non-flexible compute
- Low scale
- Specific frameworks and configuration

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



#2 use case – Data processing on AWS Batch

- Monolithic jobs
- High scale
- Basic dependency management
- Compute flexibility
  - Job requirements vary across vCPU, RAM, and GPU

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



#3 use case – Data applications on Amazon EKS

- Workloads
  - Stateful with local Amazon EBS
  - Stateless applications
- Pod auto scaling using HPA
- Multi-AZ for resiliency and fault tolerance
- Combine On-Demand and Spot

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



#3 use case – Data applications on Amazon EKS

- Workloads
  - Stateful with local Amazon EBS
  - Stateless applications
- Pod auto scaling using HPA
- Multi-AZ for resiliency and fault tolerance
- Combine On-Demand and Spot

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



#4 use case – Data analytics using Apache Spark on Amazon EKS

- Large-scale data transformation using map/reduce functions
- Apache Spark K8s operator
- Data hosted in memory, data shuffling using PVC
- Compute orchestrated with Karpenter
  - Different instance types
  - Spot Instances
  - High resource efficiency
  - Multi-AZ allows high scale

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



#4 use case – Data analytics using Apache Spark on Amazon EKS

- Large-scale data transformation using map/reduce functions
- Apache Spark K8s operator
- Data hosted in memory, data shuffling using PVC
- Compute orchestrated with Karpenter
  - Different instance types
  - Spot Instances
  - High resource efficiency
  - Multi-AZ allows high scale

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



#5 use case – DAG applications with Argo Workflows on Amazon EKS

- Orchestrate complex workflows, state monitoring, and retry policies
- Dependencies and resource management
- Checkpoints are exported to Amazon S3
- Compute orchestrated by Karpenter
  - Different instance types
  - Spot Instances
  - High resource efficiency
  - Multi-AZ allows high scale

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



#5 use case – DAG applications with Argo Workflows on Amazon EKS

- Orchestrate complex workflows, state monitoring, and retry policies
- Dependencies and resource management
- Checkpoints are exported to Amazon S3
- Compute orchestrated by Karpenter
  - Different instance types
  - Spot Instances
  - High resource efficiency
  - Multi-AZ allows high scale

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Simply run a complex workflow

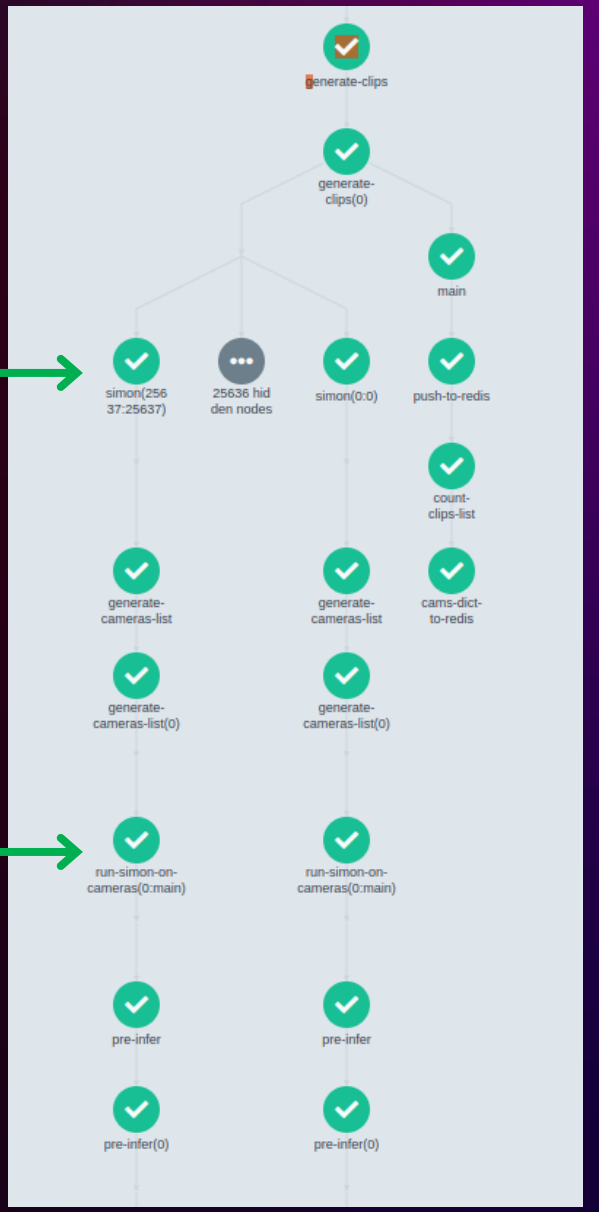
- Each job in the workflow is a container
- Dependencies are captured using a graph (DAG)
- Executors run in nested parallelism

What this allows

- Faster runtime
- Checkpointing
- Easy recovery

```
resources:
        limits:
          cpu: 4000m
          memory: 8Gi
        requests:
          cpu: 4000m
          memory: 8Gi
```

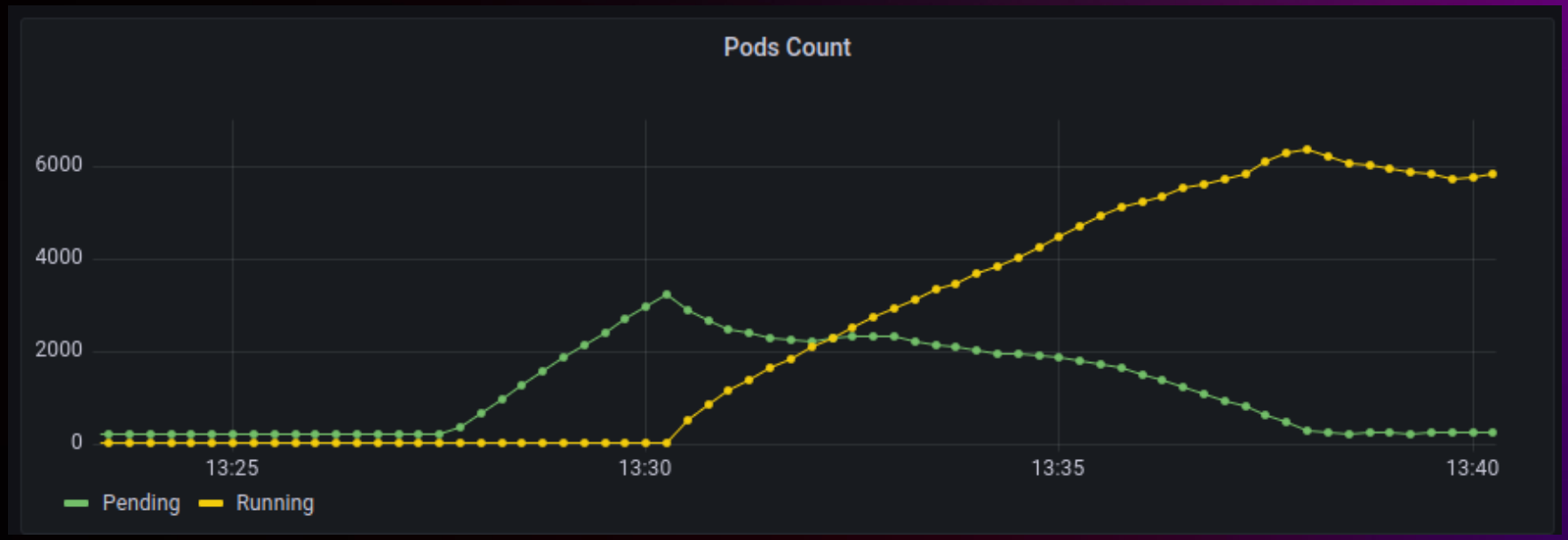
```
resources:
        limits:
          cpu: 11900m
          memory: 45G
          nvidia.com/gpu: '1'
        requests:
          cpu: 11900m
          memory: 45G
          nvidia.com/gpu: '1'
```

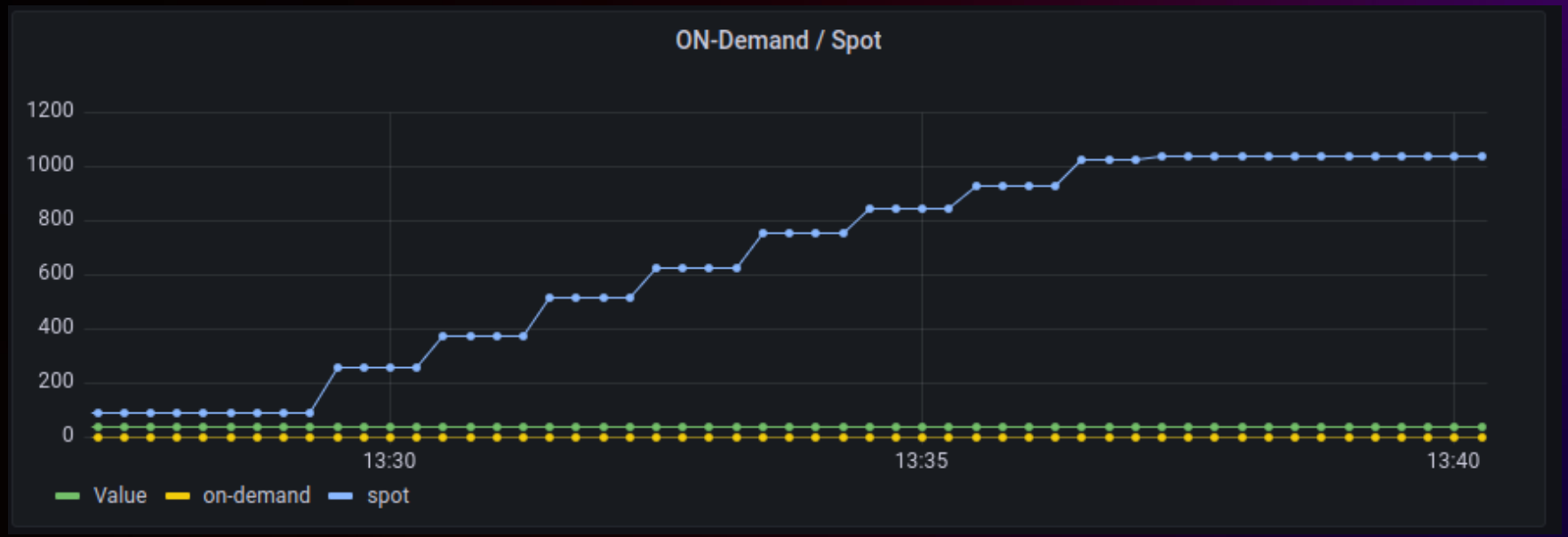


© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Key metrics

A single workflow:
6,000 pods
1,200 Spot nodes
10 minutes





© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data services on Amazon EKS – A simple experience for our data engineers



Data engineer



Amazon EKS operator

Data frameworks



Spark



ArgoCD

Orchestration



Amazon EKS



Karpenter

Compute



Amazon EC2



Spot



Accelerated computing



Memory optimized

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

One size does not fit all – A different solution for different types of data application



Amazon SageMaker

Machine learning training



AWS Batch

Large-scale data processing

Amazon EKS

Orchestrate data applications and underlying infrastructure



Apache Spark on Amazon EKS

Map reduce

Argo Workflows on Amazon EKS

Complex workflow management

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Summary

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Key takeaways

1. Different workloads need a different compute approach
2. **Compute flexibility is key!** Take advantage of the broad choices available by Amazon EC2 in order to achieve better elasticity, availability, and performance and lower prices
3. Understanding a **workload's performance requirements and consumption patterns** will help identify what instances can fit your needs
4. **Deploy across Availability Zones** to achieve high scalability and fault tolerance
5. Automate **instance lifecycles** to be able to handle failures and updates
6. **Take advantage of tools** that provision, manage, and scale compute to reduce overhead and increase resource efficiency

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Yael Grossman
yaelgr@amazon.com

Jeremy Atlan
Jeremy.Atlan@mobileye.com



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

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.