Kubernetes virtually anywhere, for everyone

Barry Cooks (he/him)
Vice President, Kubernetes
Amazon Web Services

Sharmila Ramar
Global Head of Cloud and DevOps Engineering
MassMutual

Sheetal Joshi
Senior Developer Advocate
Amazon Web Services
Our journey today

• Application modernization – the problem and technology solutions
• Why customers choose AWS for Kubernetes
• Our customers and what are they building
• What we’ve been up to
• How to get going
CIOs say that 80% of developers’ time is spent on the operations and maintenance of applications and only 20% of the time is actually spent on innovation.

Deloitte
Customer needs

- Get to market faster
- Lower total cost of ownership
- High performance and scalability
- Security and isolation by design
How do you innovate?

Experiment

Idea

Innovation flywheel

Feedback
Why modernize with containers?

- **Portability**: Everything you need to run your application

- **Infrastructure efficiency**: Lightweight – higher density

- **Performance**: Fast start/stop times
  Easy unit of scale

- **Operational efficiency**: Focus on business logic instead of infrastructure
Why Kubernetes?

Ease
- Declarative and self-reconciling
- Flexible and extensible

Consistency
- Same API, regardless of where you run or at what scale

Ecosystem
- Hundreds of solutions across the CNCF ecosystem

Community
- De facto standard with numerous enterprises helping chart the future

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Why choose AWS for Kubernetes?
Principles

- Security first
- Built for production
- Seamless cloud integrations
- Native and upstream
- Committed to open source
Security patching, including embargoed CVEs

Amazon EKS supports 4 Kubernetes versions at any given time; follows upstream release/deprecation

Supports automatic upgrades of worker nodes and control plane

Region-spanning, highly available architecture for all clusters by default
Seamless cloud integrations

KUBERNETES ACCESS TO AWS

Higher-level services

Amazon EMR

Amazon GuardDuty

Amazon EKS

Supporting services

Amazon SQS

Amazon DynamoDB

Amazon RDS

Infrastructure services

Amazon EC2

AWS KMS

IAM
Native and upstream

If it runs on Kubernetes, it runs on Amazon EKS – that includes Istio, Knative, Kubeflow, etc.
Committed to open source

- cortex
- fluentbit
- containerd
- kubernetes
- OpenTelemetry
- etcd
- tinkerbell

What we do

- New code
- Testing
- Bug fixes
- Documentation
- Best practices
- Security response
AWS is pushing the boundaries with AWS Outposts, AWS Wavelength, AWS Local Zones, and now on-premises, edge, and hybrid capabilities.
Amazon EKS Anywhere infrastructure options

- VMware (GA)
- Bare metal (GA)
- Apache CloudStack (GA)
- Nutanix (Preview)

OS:
- Bottlerocket
- Ubuntu 8
- RHEL
AWS global reach

30 geographic Regions
96 Availability Zones
21 Local Zones
29 Wavelength Zones

Coming soon
5 new geographic Regions
15 new Availability Zones
30 new Local Zones
More than 2/3 of containers in the Cloud run on AWS.

CNCF survey
What customers are building

**Legacy app modernization**
- .NET apps
- Legacy homegrown Linux apps
- Monoliths

**AI/ML**
- Autonomous vehicles (object tracking, sensor fusion)
- Robotics (vision, grasping, motion control)
- Modeling, training, and inference

**Data processing**
- Real time
- MapReduce
- Batch

**Backends**
- Apps and services
- Mobile
- IoT

**Web applications**
- Static websites
- Complex web apps
Internet gaming

**Situation**

Needed a compute service that could handle immense scale, rapid deployments, and global reach

**Results**

VALORANT (14M monthly active players) and Legends of Runeterra are now running on EKS

Riot is running EKS clusters in most AWS Regions
AI/ML

Situation
To develop the Aurora Driver, a self-driving platform, Aurora needs a range of compute-intensive workloads for machine learning, computer vision, and simulation. Aurora also needs a compute platform that can scale.

Results
- System can now scale to 10 million tasks a day
- In the future, it will handle a billion tasks a day
Financial services

**Situation**
Wanted to enable internal teams with 15K technologists running over 250K containers to realize benefits of the cloud and the agility and innovation that come with it while avoiding cloud infrastructure management.

**Results**
Technologists can now explore new & exciting technology solutions in a safe and scalable manner. Modern application development is now the norm with working groups, demo days, exploratory POCs & MVPs.

“Containerization is not an answer looking for a problem, it’s a way of blending both innovation & scalability into successful effectiveness!”

Amr Abdelhalem
SVP, Head of Cloud Platforms, Fidelity Investments
Travel services

Situation

Like many enterprises, M&A activity at Expedia Group increases the breadth of compute platforms that engineers need to manage and has driven a need for standardization.

Results

RCP manages 400 EKS clusters and orchestrates the distribution of workloads across the platform.

Over 9,000 applications have been identified for migration to RCP.
Founded in 1851, Massachusetts Mutual Life Insurance Co. is one of the largest US insurers.

Our company has been continually guided by one consistent purpose:

“We help people secure their future and protect the ones they love.”

MassMutual’s Vision is “to provide financial wellbeing to all Americans.”
MassMutual cloud-first strategy

The MassMutual AWS journey began in 2015

• Kicked off the move away from bespoke on-premises private cloud solution
• Early adopters included digital & data science departments

Enterprise Data Analytics Platform (EDAP) – an AWS Cloud native solution to replace legacy data warehouse platforms

EDAP success led us to invest in the cloud-first strategy in 2019
MassMutual path to modernization

**Strategy**
- Replatform
- Refactor
- Build new

**Pilot**
- Move to manage as containers in EKS
- Build new with serverless

**Define**
- Development practices
- Operational expertise
- Deployment best practices
- Cost management & governance

**Optimize**
- Decrease build & deployment time
- Reduce TCO and increase ROI
- Increase time to market

**Scale**
- Organization-wide
- Critical workloads
- Global reach
MassMutual: What’s next?

Our cloud-first strategy addresses . . .

- Standardization
- Security and risk management
- Cloud operating model

**Results**

- Reduced overall costs and operational complexity
  - Running 110 EKS clusters and 100 business applications
  - Migrating an additional 150+ services and APIs to Amazon EKS
What we’ve been up to

COST OPTIMIZATION
Customer cost challenges

Allocate costs across teams and departments

Chargeback/showback

Reporting, budget forecasting, and cost optimization
In-cluster EKS cost management with Kubecost

• Licensed Kubecost features available to EKS customers at no additional cost
  • Granular cost breakdown by deployment, pod, namespace, and other Kubernetes concepts
  • Support included
  • Multi-cluster cost visibility and unified reporting (roadmap)

• Integration with AWS Cost and Usage Reports for accurate pricing

• AWS Marketplace integration
Karpenter – Cost-efficient compute for Kubernetes

Karpenter is an *intelligent* and *high-performance* Kubernetes compute provisioning and management solution.

Karpenter lets you take full advantage of AWS with its deep integration between Kubernetes and Amazon EC2.
Why Karpenter?

Responds in seconds when application load changes
improve availability

Kubernetes-native CRDs and opinionated defaults
minimize operational overhead

Intelligently chooses instance types and consolidates pods to
lower compute costs
Karpenter in action

Sheetal Joshi
Senior Developer Advocate
AWS
bash-5.1$ kubectl get node
NAME STAIUS ROLES AGE VERSION
ip-192-168-84-35.us-west-2.compute.internal Ready <none> 9d v1.23.9-eks-ba74326
bash-5.1$ kubectl get pod -A
NAMESPACE NAME READY STATUS RESTARTS AGE
karpenter karpenter-65f6584977-tqbr5 2/2 Running 0 8m27s
karpenter karpenter-65f6584977-xjzk7 2/2 Running 0 8m27s
kube-system aws-load-balancer-controller-5dc48f896d-vrf6b 1/1 Running 1 (8d ago) 8d
kube-system aws-load-balancer-controller-5dc48f896d-vrv2p 1/1 Running 0 8d
kube-system aws-node-76tzc 1/1 Running 0 9d
kube-system coredns-85d5b4454c-fhxwb 1/1 Running 0 8d
kube-system coredns-85d5b4454c-m4cpp 1/1 Running 0 8d
kube-system efs-csi-controller-69bcf9b47-swml4 3/3 Running 1 (8d ago) 8d
kube-system efs-csi-controller-69bcf9b47-x87xz 3/3 Running 0 8d
kube-system kube-proxy-qd8dj 1/1 Running 0 9d
kube-system metrics-server-64cf6869bd-v29ln 1/1 Running 0 8d
monitoring prometheus-node-exporter-2b7rl 1/1 Running 0 9d
bash-5.1$ vi inflate.yaml
bash-5.1$
apiVersion: apps/v1
kind: Deployment
metadata:
  name: inflate
spec:
  replicas: 0
  selector:
    matchLabels:
      app: inflate
  template:
    metadata:
      labels:
        app: inflate
    spec:
      terminationGracePeriodSeconds: 0
      containers:
      - name: inflate
        image: public.ecr.aws/eks-distro/kubernetes/pause:3.2
        resources:
          requests:
            cpu: 250m
<table>
<thead>
<tr>
<th>NAME</th>
<th>STATUS</th>
<th>ROLES</th>
<th>AGE</th>
<th>VERSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-192-168-84-35.us-west-2.compute.internal</td>
<td>Ready</td>
<td>&lt;none&gt;</td>
<td>9d</td>
<td>v1.23.9-eks-ba74326</td>
</tr>
</tbody>
</table>

bash-5.1$ kubectl get pod -A

<table>
<thead>
<tr>
<th>NAMESPACE</th>
<th>NAME</th>
<th>READY</th>
<th>STATUS</th>
<th>RESTARTS</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>karpenter</td>
<td>karpenter-65f6584977-tqbr5</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>8m27s</td>
</tr>
<tr>
<td>karpenter</td>
<td>karpenter-65f6584977-xjzk7</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>8m27s</td>
</tr>
<tr>
<td>kube-system</td>
<td>aws-load-balancer-controller-5dc48f896d-vrf6b</td>
<td>1/1</td>
<td>Running</td>
<td>1 (8d ago)</td>
<td>8d</td>
</tr>
<tr>
<td>kube-system</td>
<td>aws-load-balancer-controller-5dc48f896d-vrv2p</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>kube-system</td>
<td>aws-node-76tzc</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>9d</td>
</tr>
<tr>
<td>kube-system</td>
<td>coredns-85d5b4454c-fhxwb</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>kube-system</td>
<td>coredns-85d5b4454c-m4cpp</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>kube-system</td>
<td>efs-csi-controller-69bcf9b47-swml4</td>
<td>3/3</td>
<td>Running</td>
<td>1 (8d ago)</td>
<td>8d</td>
</tr>
<tr>
<td>kube-system</td>
<td>efs-csi-controller-69bcf9b47-x87xz</td>
<td>3/3</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>kube-system</td>
<td>kube-proxy-qd8dj</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>9d</td>
</tr>
<tr>
<td>kube-system</td>
<td>metrics-server-64cf6869bd-v29ln</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>monitoring</td>
<td>prometheus-node-exporter-2b7rl</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>9d</td>
</tr>
</tbody>
</table>
5 nodes 150500m/159250m 94.5% cpu $6.120/hour $4467.600/month
615 pods (0 pending 615 running 615 bound)

Press any key to quit
kind: Provisioner

metadata:
  name: default

spec:
  consolidation:
    enabled: false

limits:
  resources:
    cpu: 5k
    aws.amazon.com/neuron: 0
    nvidia.com/gpu: 0
    amd.com/gpu: 0

requirements:
  - key: kubernetes.io/arch
    operator: In
    values:
      - amd64
  - key: "karpenter.sh/capacity-type"
    operator: In
    values:
      - on-demand
  - key: karpenter.k8s.aws/instance-cpu
    operator: Lt
    values: ["33"]
  - key: "topology.kubernetes.io/zone"
    operator: In
    values: ["us-west-2a", "us-west-2b", "us-west-2c", "us-west-2d"]

provider:
  amiFamily: AL2
  apiVersion: extensions.karpenter.sh/v1alpha1
  kind: AWS
  securityGroupSelector:
    karpenter.sh/sg-discovery: eks-demo
  subnetSelector:
    karpenter.sh/sn-discovery: eks-demo

apiVersion: karpenter.sh/v1alpha5
kind: Provisioner
metadata:
  name: default
spec:
  consolidation:
    enabled: false
  limits:
    resources:
      cpu: 5k
      aws.amazon.com/neuron: 0
      nvidia.com/gpu: 0
      amd.com/gpu: 0
  requirements:
  - key: kubernetes.io/arch
    operator: In
    values:
      - amd64
  - key: "karpenter.sh/capacity-type"
    operator: In
    values:
      - on-demand
  - key: karpenter.k8s.aws/instance-cpu
    operator: Lt
    values: ["33"]
  - key: "topology.kubernetes.io/zone"
    operator: In
    values: ["us-west-2a", "us-west-2b", "us-west-2c", "us-west-2d"]
provider:
  amiFamily: AL2
  apiVersion: extensions.karpenter.sh/v1alpha1
  kind: AWS
securityGroupSelector:
  karpenter.sh/sg-discovery: eks-demo
subnetSelector:
  karpenter.sh/sn-discovery: eks-demo

~
"provisioner.yaml" 36L, 830B
4 nodes 110400m/114400m 99.1% cpu $3.808/hour $2779.840/month
452 pods (0 pending 452 running 452 bound)

ip-192-168-80-218.us-west-2.compute.internal cpu
100% (66 pods) c6g.4xlarge/$0.544 On-Demand ready

ip-192-168-118-126.us-west-2.compute.internal cpu
98% (128 pods) c6g.8xlarge/$1.088 On-Demand ready

ip-192-168-165-17.us-west-2.compute.internal cpu
100% (130 pods) c6g.8xlarge/$1.088 On-Demand ready

ip-192-168-110-36.us-west-2.compute.internal cpu
98% (128 pods) c6g.8xlarge/$1.088 On-Demand ready

Press any key to quit

bash-5.1$ kubectl scale deployment inflate --replicas=600
deployment.apps/inflate scaled
bash-5.1$ kubectl scale deployment inflate --replicas=500
deployment.apps/inflate scaled
bash-5.1$ kubectl scale deployment inflate --replicas=440
deployment.apps/inflate scaled
bash-5.1$
What we’ve been up to

EXTENDING ACCESS TO AWS
AWS Controllers for Kubernetes (ACK)

PROVISION AWS RESOURCES ALONGSIDE KUBERNETES APPLICATIONS

Harness AWS
Create and use AWS resources directly within your cluster; improve reliability and uptime at virtually any scale

Cloud-native control
Kubernetes custom resources and controllers allow you to define the AWS resources your applications need directly within the cluster

Always up to date
ACK generates automatically using the AWS SDKs, this ensures controllers are up to date with the latest features and functionality

github.com/aws-controllers-k8s
AWS Controllers for Kubernetes (ACK)

PROVISION AWS RESOURCES ALONGSIDE KUBERNETES APPLICATIONS

Generally available now
- Amazon Managed Service for Prometheus
- Amazon API Gateway V2
- AWS Application Auto Scaling
- Amazon DynamoDB
- Amazon EC2
- Amazon ECR

Amazon EKS
- Amazon EKS
- AWS KMS
- AWS Lambda
- Amazon RDS
- Amazon S3
- Amazon SageMaker
- AWS Step Functions

Available in preview
- Amazon API Gateway
- Amazon CloudFront
- AWS CloudTrail
- Amazon ElastiCache
- Amazon EMR
- Amazon OpenSearch Service
- IAM
- Amazon MSK
- Amazon Kinesis
- Amazon MemoryDB for Redis
- Amazon MQ
- Amazon SNS

github.com/aws-containers-k8s
AWS Batch for EKS

Features

- Fully managed batch computing
- Multi-cluster aware scheduling
- Workload-aware scaling and scheduling
- Compatible with any EKS cluster

Use preferred monitoring and governance tooling

Batch manages application isolation from other EKS workloads

Genomics & drug discovery
ML training
Data engineering pipelines
Processing logs

AWS Batch

EKS clusters
What we’ve been up to

EASY ACCESS TO PARTNER SOFTWARE
Production-ready clusters are more than your code
AWS Marketplace software with EKS add-ons

Common OSS tools built and vended by AWS

Vendor-provided tools from AWS Marketplace

Launch using EKS add-ons

EKS clusters
ISV catalog

Launch partners

- kubecost
- Teleport
- factor house
- Tetraste Istio Subscription
- dynatrace
- upbound

Coming soon

- portworx
- Lightrun
- Shoreline
- AEROSPIKE
- NeuVector
- Kong
- paloalto
- TRILIO
- nOps
- PARALUS
- sysdig
- LACEWORK
- kaster by Veeam
- DataStax
- new relic
- Rafay
- DataDog
- StormForge
- datree
- nirmata
- HAProxy
Kubernetes for everyone
Our focus areas

- **Community**
  - Contribution
  - Funding
  - Security

- **Technology**
  - Project leadership
  - Operational best practices

- **Product**
  - Global availability
  - Hybrid support
  - AWS innovations like Batch

**Customers**
Managing Kubernetes at scale

MORE SMALLER CLUSTERS

**Improved isolation**
Multiple clusters improve security isolation, tracking, and management capabilities

**Everything scales**
Cluster creation, updates, and other lifecycle actions scale linearly

**Automation and standards are critical**
To enable operations at scale
Managing Kubernetes at scale

**Fully managed control plane**
EKS offers a standard, automatically scaled control plane

**Managed compute**
Automate and standardize compute provisioning for every cluster, including pre- or post-provisioned compute

**Operational tools**
Standardized tooling deployments

**Single pane of glass**
EKS console lets you view and troubleshoot the entire Kubernetes API across all of your clusters from a single location, no matter where they run
Managing Kubernetes at scale

ON THE HORIZON

Simplified actions

Opinionated templates

Reconciling deployments

Improved monitoring and troubleshooting
How to get started
AWS open source technical field community

Leverage AWS expertise for the open source community

AWS is active across open source ecosystem
EKS Blueprints

An open-source framework that allows you to **configure and deploy complete** clusters

- Infrastructure as code with Terraform and AWS CDK
- Based on AWS best practices and recommendations
- Integrated with popular K8s tools and services
- Fully extensible and customizable

Visit EKS Blueprints Quick Start to learn more:
Data on EKS

Open-source patterns that help you deploy **data workloads** on Amazon EKS

- Infrastructure as code (IaC) templates
- Performance benchmark reports
- AWS best practices for data workloads
- Deployment examples and architectures

Visit Data on EKS portal to learn more:
AWS Data Lab for containers

Work backwards from big ideas

Focused, real-world solution building

De-risked path to production

Come with an idea, leave with a solution

Visit the AWS Data Lab for more info:
AWS customer enablement

ACCELERATE TIME TO MARKET FOR YOUR PLATFORM

AWS Professional Services
A global network with deep AWS expertise

AWS Specialists
Deep domain experts who support customer adoption

AWS Technology Partners
Third-Party software to help accelerate your business outcomes

AWS Consulting Partners
On-demand help from AWS Certified third-party experts
Thank you!

Barry Cooks
linkedin.com/in/barryjcooks

Please complete the session survey in the mobile app