



Kubernetes Event Driven Autoscaling - KEDA

Kerrigan Lin

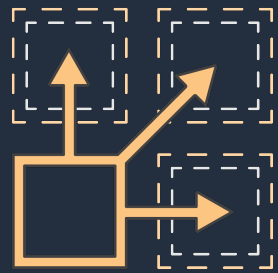
Solutions Architect @AWS Taiwan



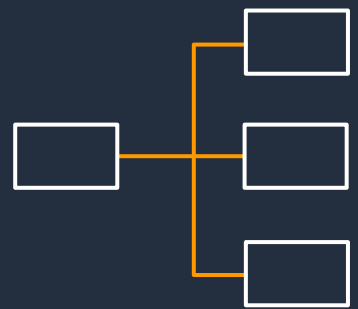
Kerrigan Lin (林彥勳)

- A father
- Cloud Architect & Development Engineer
- Solutions Architect @AWS

Challenges



HPA – Horizontal Pod
Autoscaler – Kubernetes
Component for metrics-based scaling



Microservices designs
mostly pivoted around
event-driven architecture



Writing custom scalers and
operators takes time and reduces
productivity

Features



Event-driven

Intelligently scale your event-driven application



Autoscaling Made Simple

Bring rich scaling to containers in your Amazon EKS cluster



Built-in Scalers

Out-of-the-box scalers for various vendors, databases, messaging systems and more



Multiple Workload Types

Support for variety of workload types such as deployments and jobs



Vendor-Agnostic

Support for triggers across multiple vendors



AWS Fargate support

Run and scale your fargates on Amazon EKS in production workloads

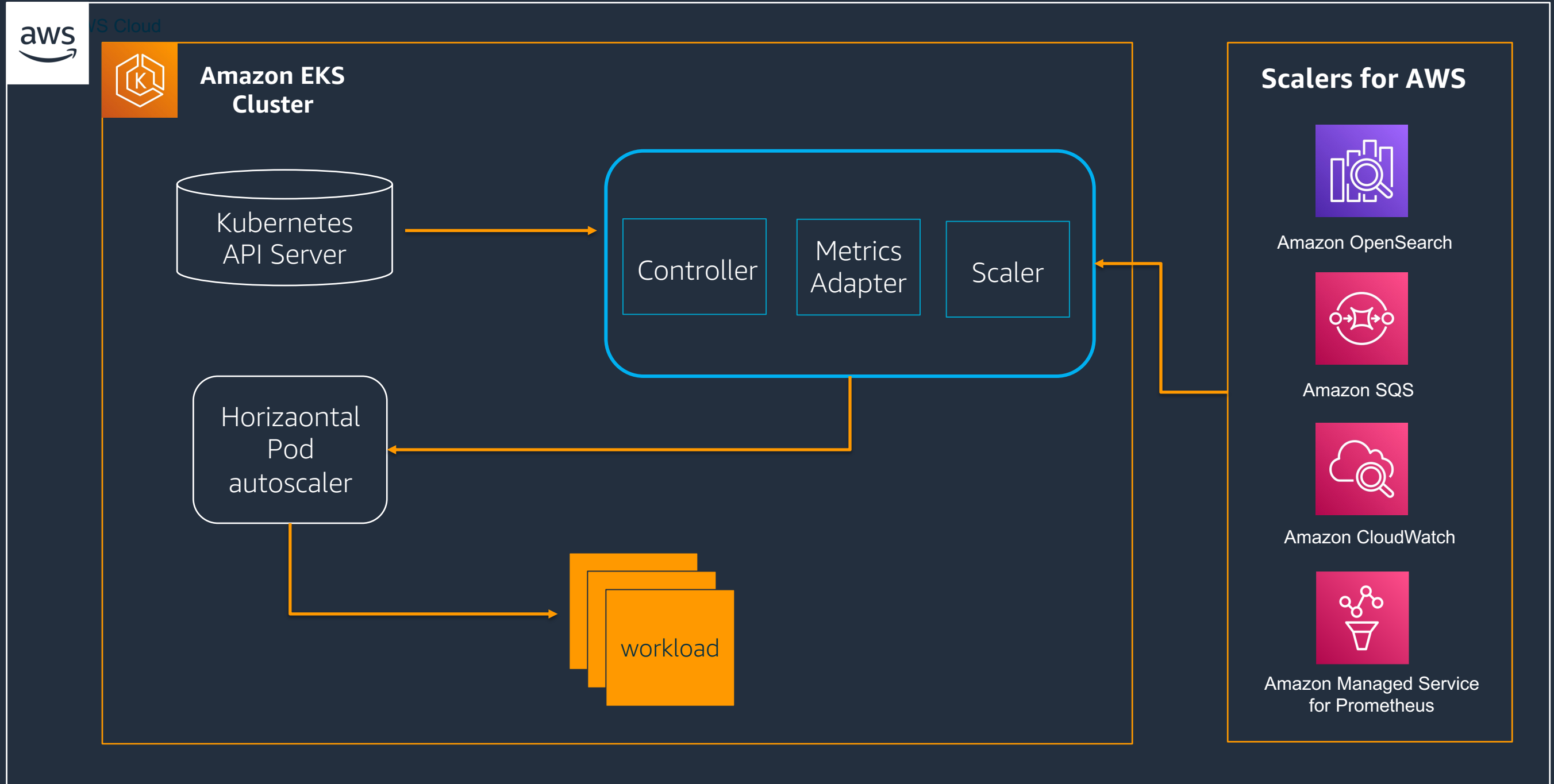
The Community

CNCF Incubating project



Get Involved -> keda.sh/community/

KEDA Architecture in Amazon EKS



More Scalers

ActiveMQ

ActiveMQ Artemis

Apache Kafka

AWS CloudWatch

AWS DynamoDB

AWS Kinesis Stream

AWS SQS Queue

Azure Application Insights

Azure Blob Storage

Azure Data Explorer

Azure Event Hubs

Azure Log Analytics

Azure Monitor

Azure Pipelines

Azure Service Bus

Azure Storage Queue

Cassandra

CPU

Cron

Datadog

Elasticsearch

External

External Push

Google Cloud Platform Stackdriver

Google Cloud Platform Storage

Google Cloud Platform Pub/Sub

Graphite

Huawei Cloudeye

IBM MQ

InfluxDB

Kubernetes Workload

Liiklus Topic

Memory

Metrics API

MongoDB

MSSQL

MySQL

NATS Streaming

New Relic

OpenStack Metric

OpenStack Swift

PostgreSQL

Predictkube

Prometheus

RabbitMQ Queue

Redis Lists

Redis Lists (supports Redis Cluster)

Redis Lists (supports Redis Sentinel)

Redis Streams

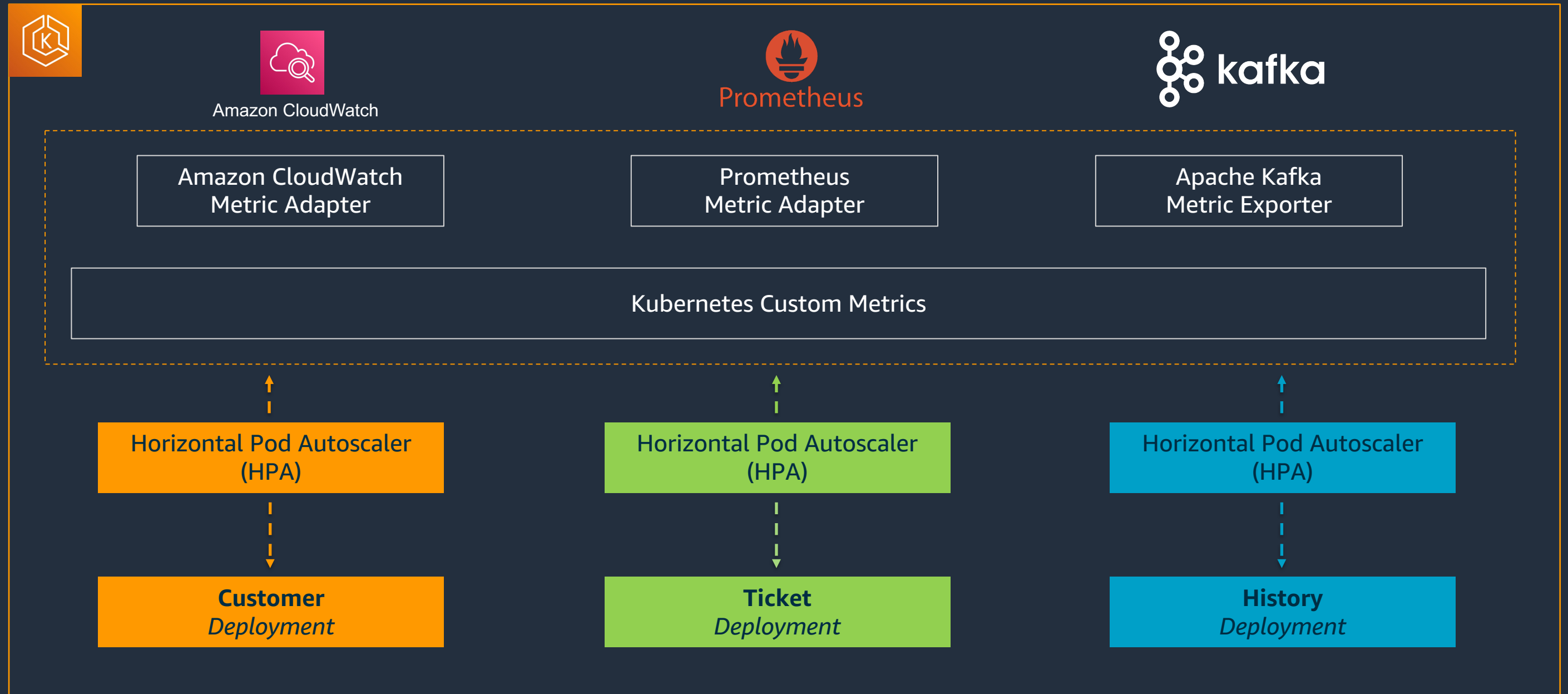
Redis Streams (supports Redis Cluster)

Redis Streams (supports Redis Sentinel)

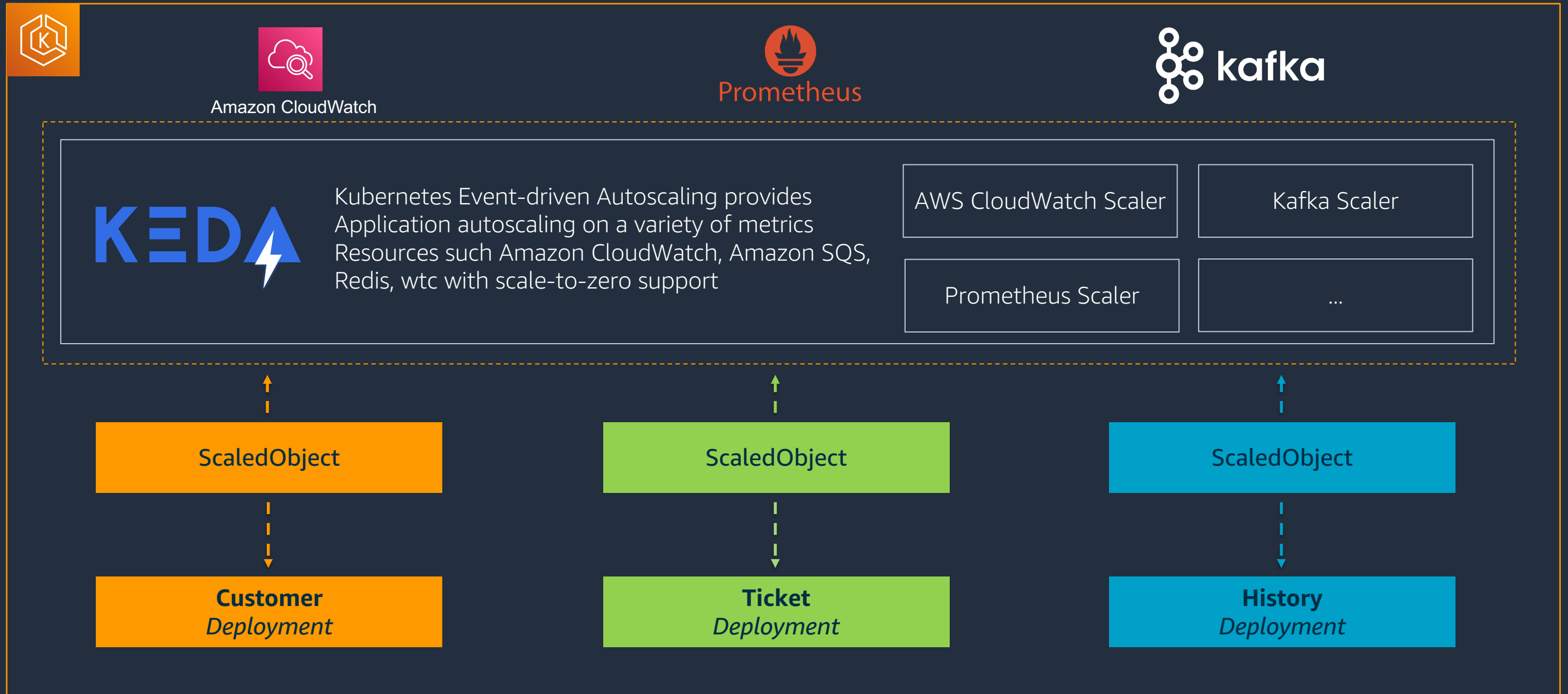
Selenium Grid Scaler

Solace PubSub+ Event Broker

KEDA Architecture Outcome



KEDA Architecture Outcome



ScaledObject Example

```
apiVersion: keda.sh/v1alpha1
kind: TriggerAuthentication
metadata:
  name: keda-trigger-auth-aws-credentials
  namespace: keda
spec:
  secretTargetRef:
    - parameter: awsAccessKeyID      # Required.
      name: test-secrets             # Required.
      key: AWS_ACCESS_KEY_ID        # Required.
    - parameter: awsSecretAccessKey # Required.
      name: test-secrets             # Required.
      key: AWS_SECRET_ACCESS_KEY    # Required.
```

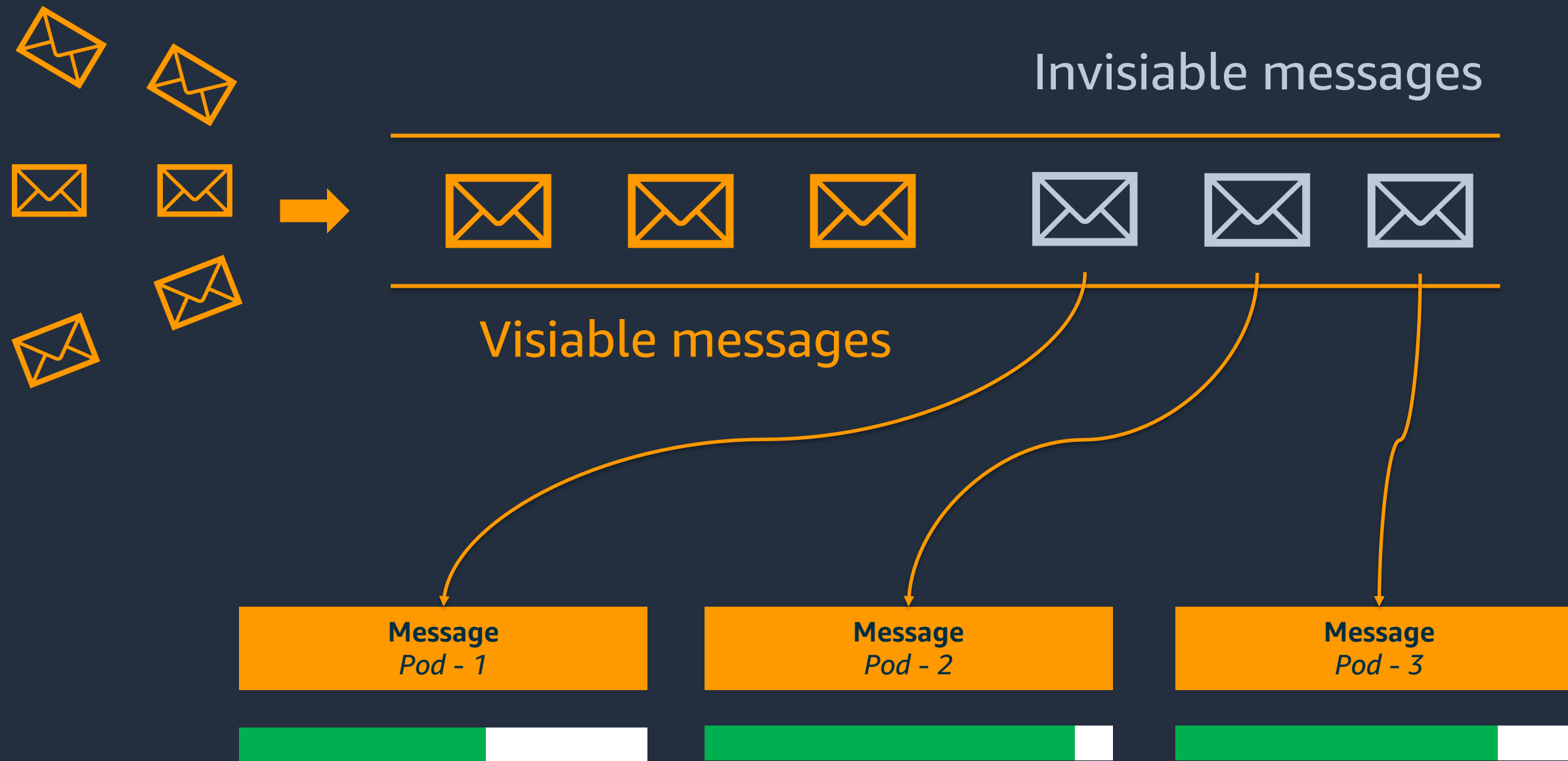
ScaledObject Example

```
apiVersion: keda.sh/v1alpha1
kind: ScaledObject
metadata:
  name: aws-sqs-queue-scaledobject
  namespace: keda
spec:
  scaleTargetRef:
    name: sqs-consumer-deployment
  pollingInterval: 10 # Optional. Default: 30 seconds
  cooldownPeriod: 10 # Optional. Default: 300 seconds
  triggers:
  - type: aws-sqs-queue
    authenticationRef:
      name: keda-trigger-auth-aws-credentials
    metadata:
      queueURL: KEDA_Queue
      queueLength: "2"
      awsRegion: "ap-northeast-1"
```

What about long running executions?



Amazon SQS Scaler for Long Running Execution



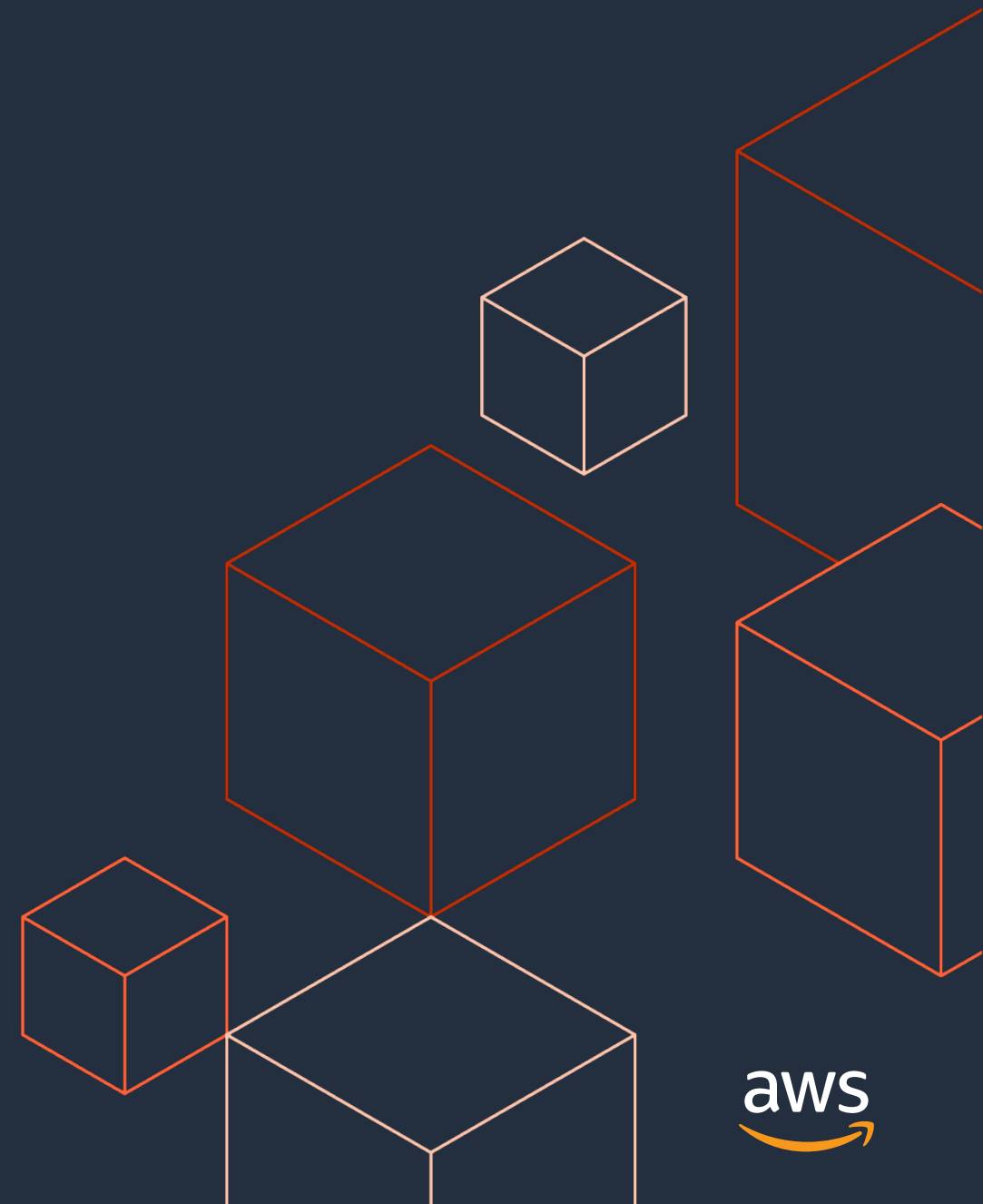
Amazon SQS Scaler for Long Running Execution

Long Running Executions + KEDA Scaling Job

Option 1 : Leverage Kubernetes pod lifecycle

Option 2 : Use the **Scaled Jobs** functionality of **KEDA** to create Kubernetes Job per event

Demo



Demo – Scaling for Amazon SQS Messages



Core Principles

- **Don't rebuild** anything that Kubernetes offers out of the box.
- Single purpose, simple, **non-intrusive**.
- Works with any container and any **workload**.
- Preserve what makes **messaging brokers** powerful.
- Open and built with **community**.

Resources

[KEDA Official Website](#)

[KEDA GitHub](#)

[KEDA Integration with Amazon Managed Service for Prometheus](#)

[KEDA Integration with Amazon SQS](#)

Survey



Thank you!

Kerrigan Lin

@kerriganlin

