# Scaling apps with KEDA

Date - March 16, 2024

By - Vinod Kumar Nair
Designation - Principal Engineer (AWS)
Email - vinod827@yahoo.com

# Who am I

- Completed B.E. (Information Technology) - 2004-2008
- About 16 years of experience in software development
- Currently working as Principal Engineer with Arcesium India Private Limited, Gurgaon
- Certified in AWS, GCP and in Kubernetes (CKAD, CKA, CKS, KCNA & KCSA)
- Open Source contributor in CNCF Projects like KEDA and Terraform from HashiCorp

# Agenda

- Monolithic applications & its deployment approach
- Microservice based application & its deployment approach
- Autoscaling in Kubernetes and its types
- Horizontal Pod Autoscaler (HPA) with demo
- Deep dive into Kubernetes Event Driven Autoscaling (KEDA) with demo
- Key Takeaways
- Reference Links
- Q & A

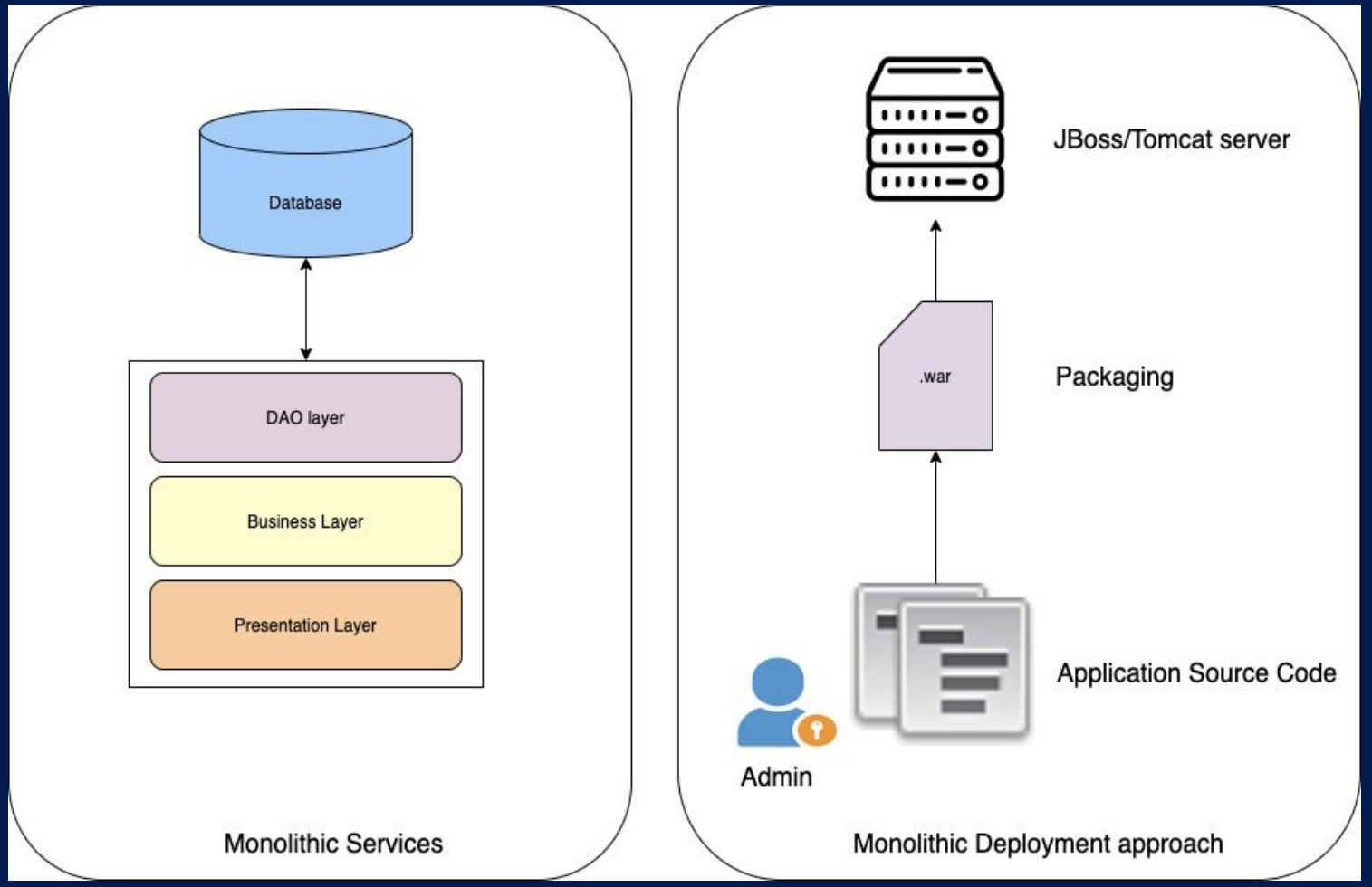
# Monolithic applications & its deployment approach

Monolithic applications are built as a single and indivisible unit.

- Simple and easy to manage applications

**Problems**:-

1) Difficult to Scale
2) Resource wastage
3) Complete outage of the application during any failure



# Microservice based application & its deployment approach

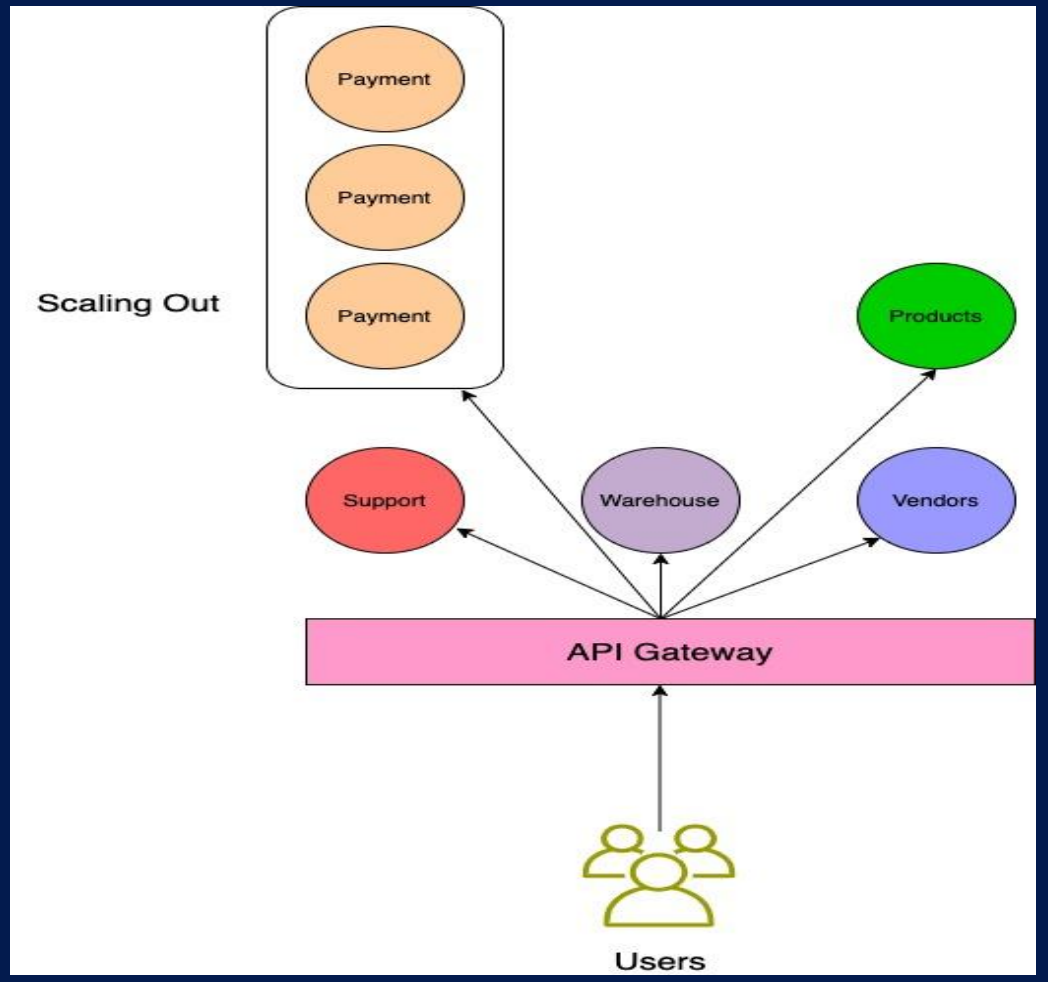
A Microservice architecture encourages every services or business units of an application to be small that are developed and run independently of other services.

**Advantages:-**

1) Scaling out/in is easy
2) Resources can be run with a pre-defined quota
3) Near zero downtime

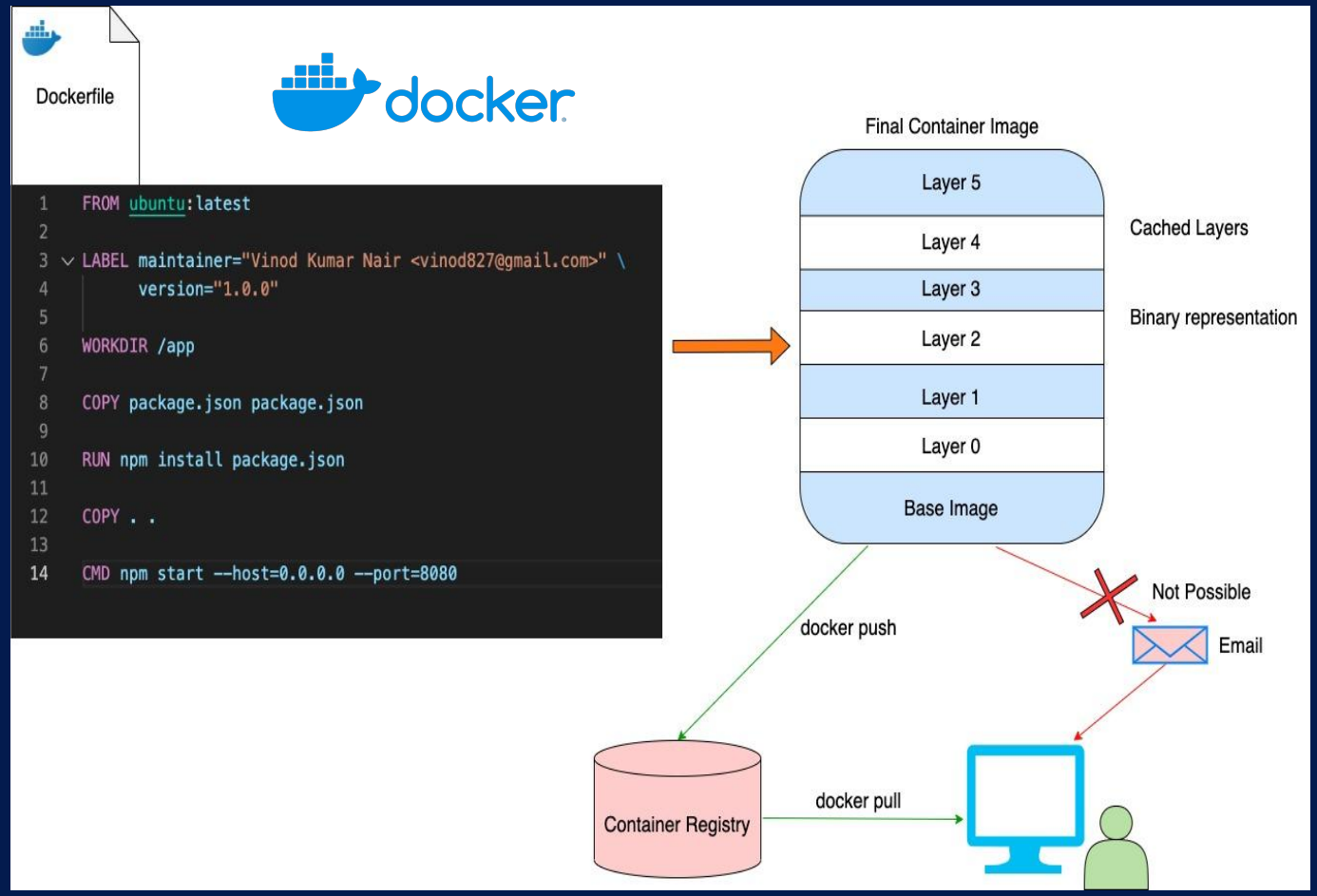
**Disadvantages:-**:-

1) Costly
2) Logging and Maintenance overhead



# Microservice based application & its deployment approach (Cont..)

- Write Dockerfile for the application to create the Docker Image
- Packages application dependencies, system libraries, etc together
- Binary representation of the application and each layer is cached unless there is a change



# Microservice based application & its deployment approach (Cont..)

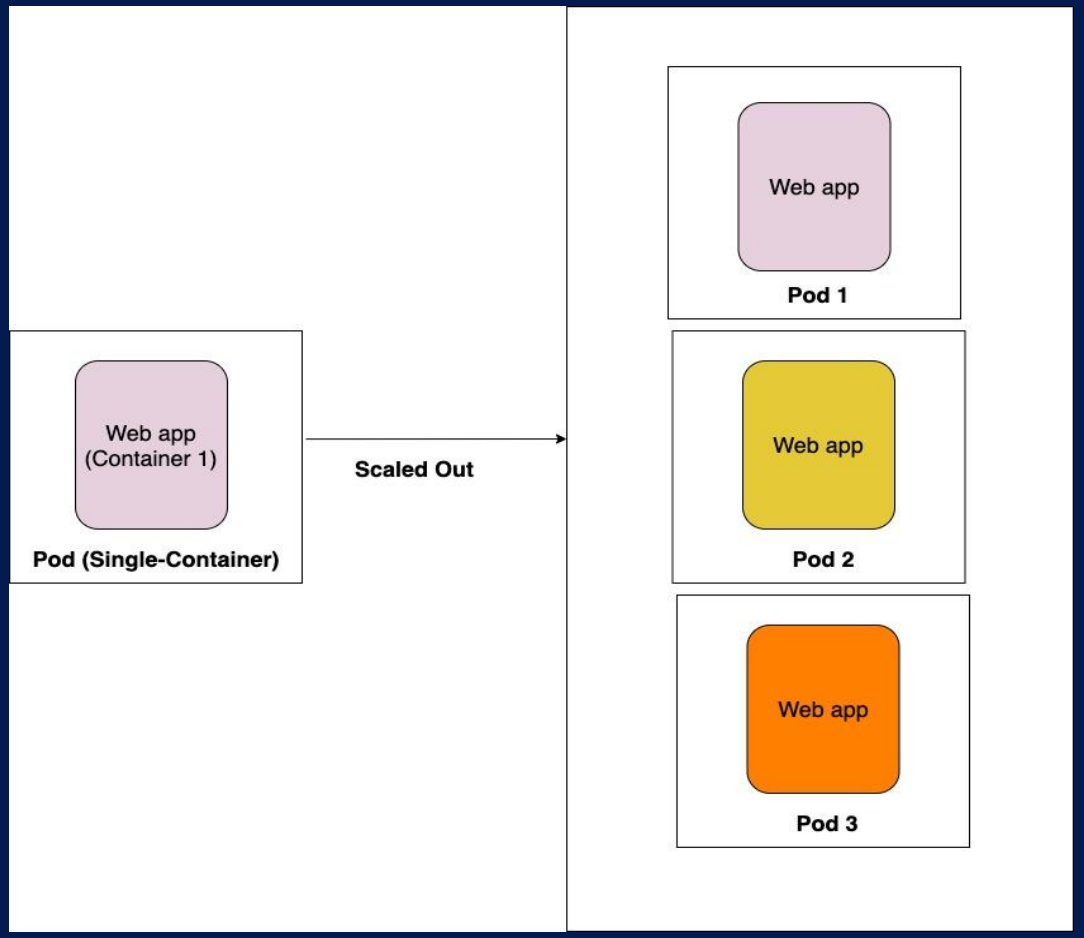
**Container Orchestration tools like Kubernetes (k8s) to deploy the application:-**

- To automate the scheduling of containers in the form of Pods and Deployments
- Scale out/in the apps easily
- Apply resource quotas
- Enforces best security practices including RBAC, Node Restrictions to protects the data from unauthorized access
- Enforces audit policy for all API calls

# Microservice based application & its deployment approach (Cont..)

**Kubernetes Deployment**

- A Kubernetes deployment encapsulates Pod and a Pod can contain 1 or more containers running the actual application
- Usually, 1 pod = 1 container unless we use the sidecar design pattern
- We define a desired state (or replica count) in the deployment manifest file and the Controller Manager maintains the actual state with respect to the desired state of the application



# Autoscaling in Kubernetes & its types

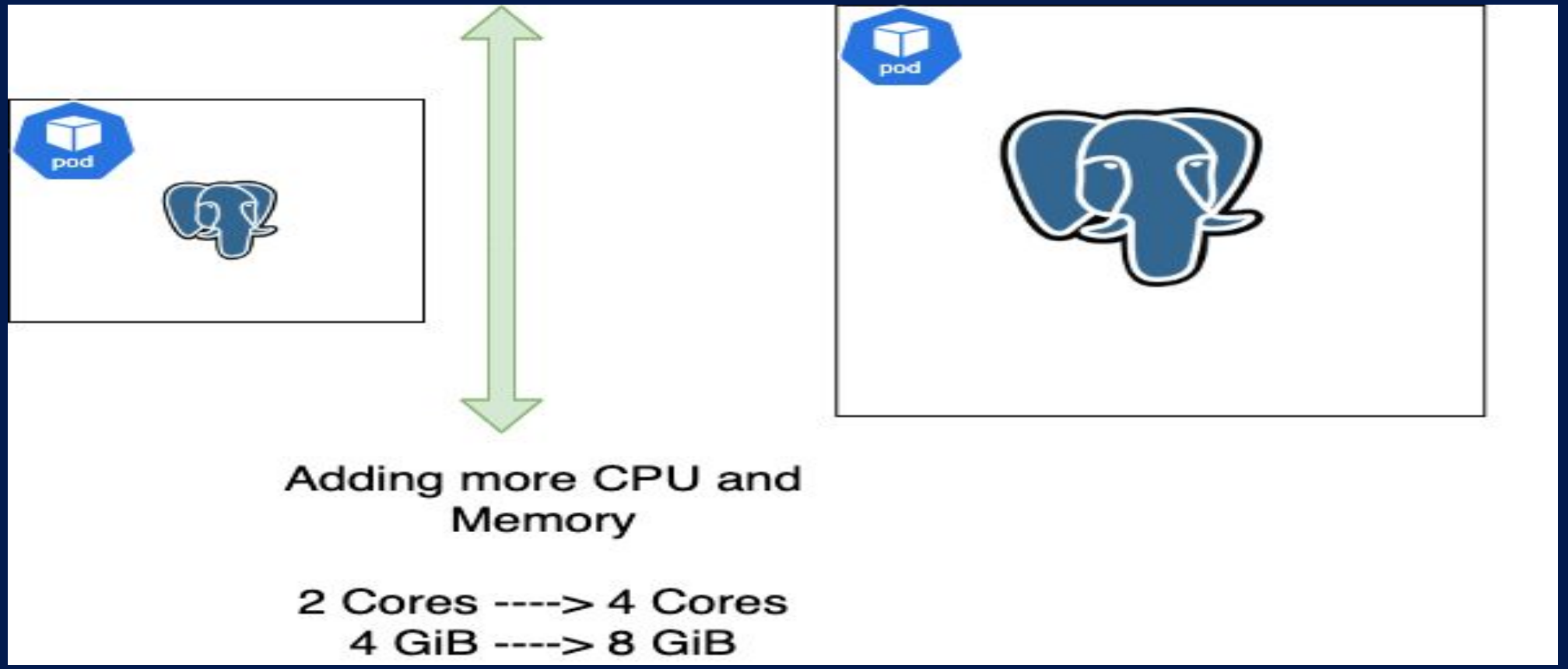
Kubernetes is intelligent in scaling the application (pods) & its cluster to meet the network demand. Kubernetes can scale the application both horizontally and vertically. It can even add additional underlying nodes to meet the demand.

Three types of Scaling support in Kubernetes:-

- **VPA (Vertical Pod Autoscaler)**
- **CA (Cluster Autoscaler)**
- **HPA (Horizontal Pod Autoscaler)**

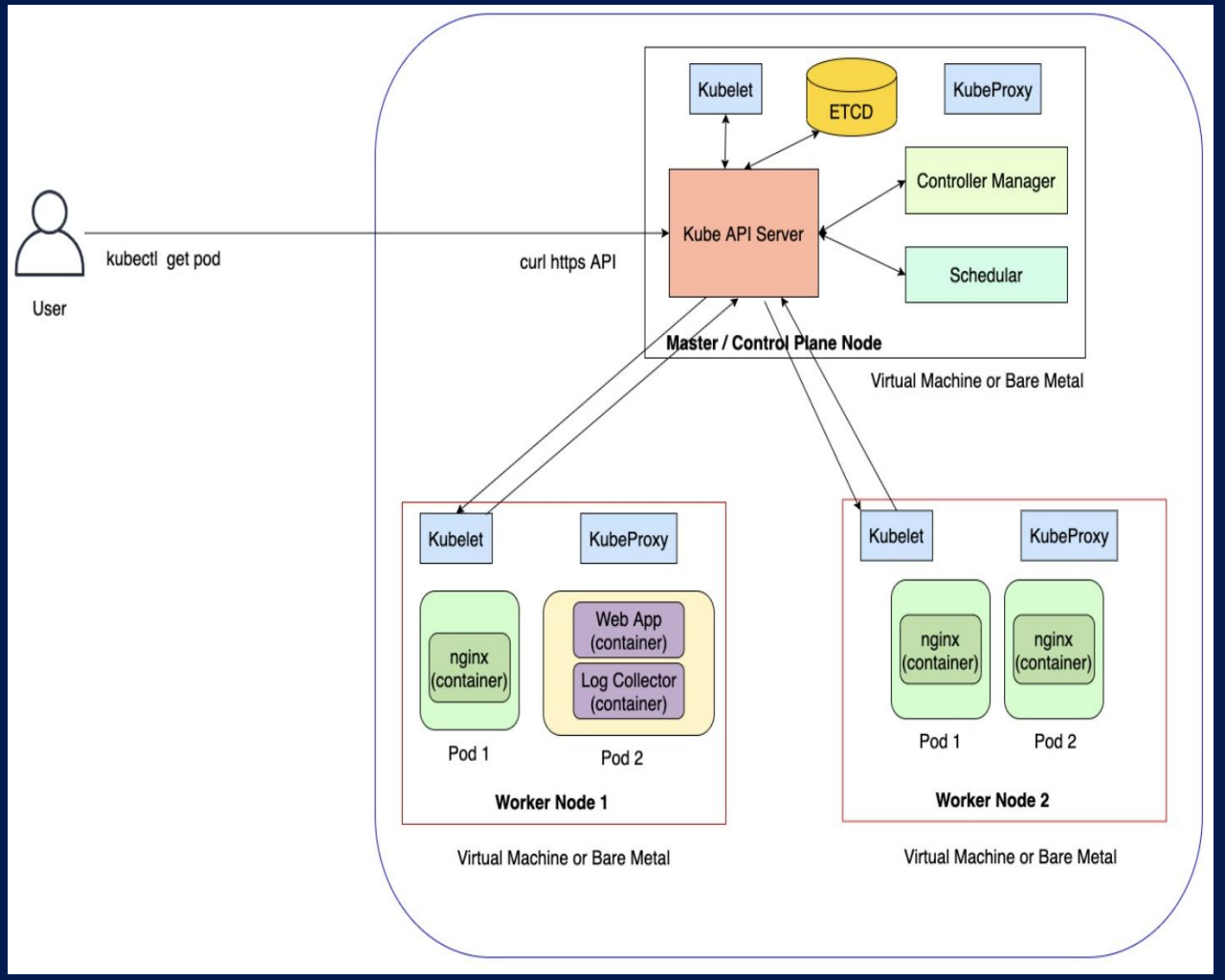
# Autoscaling in Kubernetes & its types (Cont…)

**VPA (Vertical Pod Autoscaler) -** Any stateful application which is difficult to scale horizontally and can make use of VPA to scale up/down by adding more resources to it like CPU/Memory using the appropriate VPA Mode (like Recreate, Initial, Off, etc).



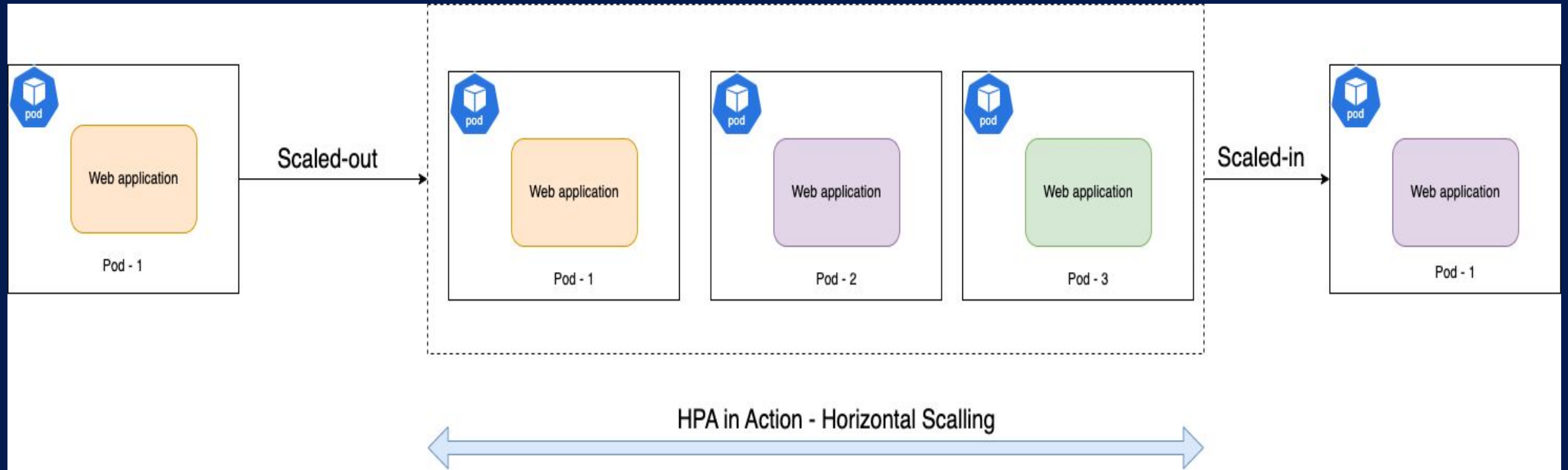
# Autoscaling in Kubernetes & its types (Cont…)

**CA (Cluster Autoscaler)** - The Scheduler component in Kubernetes is responsible for scheduling the pods to the right node of the cluster. If there are not much resources available, Pod goes into Pending State.The CA is responsible for adding more nodes to the cluster to make room for the new pods to be scheduled.



# HPA with demo

**HPA (Horizontal Pod Autoscaler) -** Any application can be scaled-out or scaled-in horizontally to meet the traffic. When HPA kicks in, it add more Pods to maintain the scalability of the application. HPA looks for CPU/Memory metrics to meet the demand of the traffic by scaling out additional pods.



# KEDA (Kubernetes Event-Driven Autoscaling)

**KEDA** or **Kubernetes Event-Driven Autoscaling** is a light weight Kubernetes component that provides autoscaling of apps horizontally with the support for numerous scalers that can be used as a source of events. It can also work along with the native HPA from Kubernetes.

The major difference or limitation when it comes to using the native HPA is we cannot scale down our application to zero (completely down) however KEDA overcome this limitation.

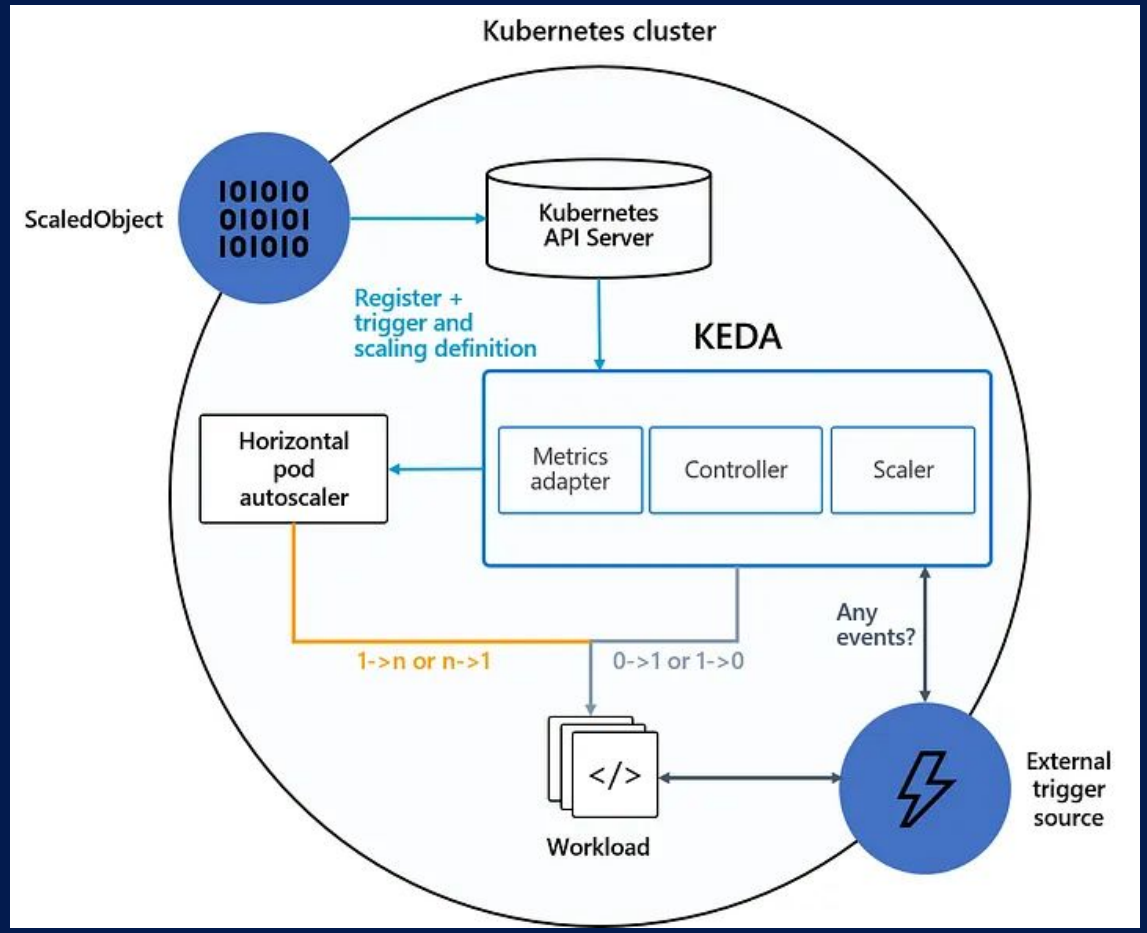
**Key tenets of using KEDA:-**

- Zero pod scaling thus saving the compute cost
- Support for different scalers (64) for source of events like AWS SQS Queue, RabbitMQ, Redis, Prometheus, etc for scaling the application
- Support for native scaling using CPU & Memory

# KEDA (Kubernetes Event Driven Autoscaling)

**Architecture of KEDA:-**

- Metric adapter
- Controller
- Scaler

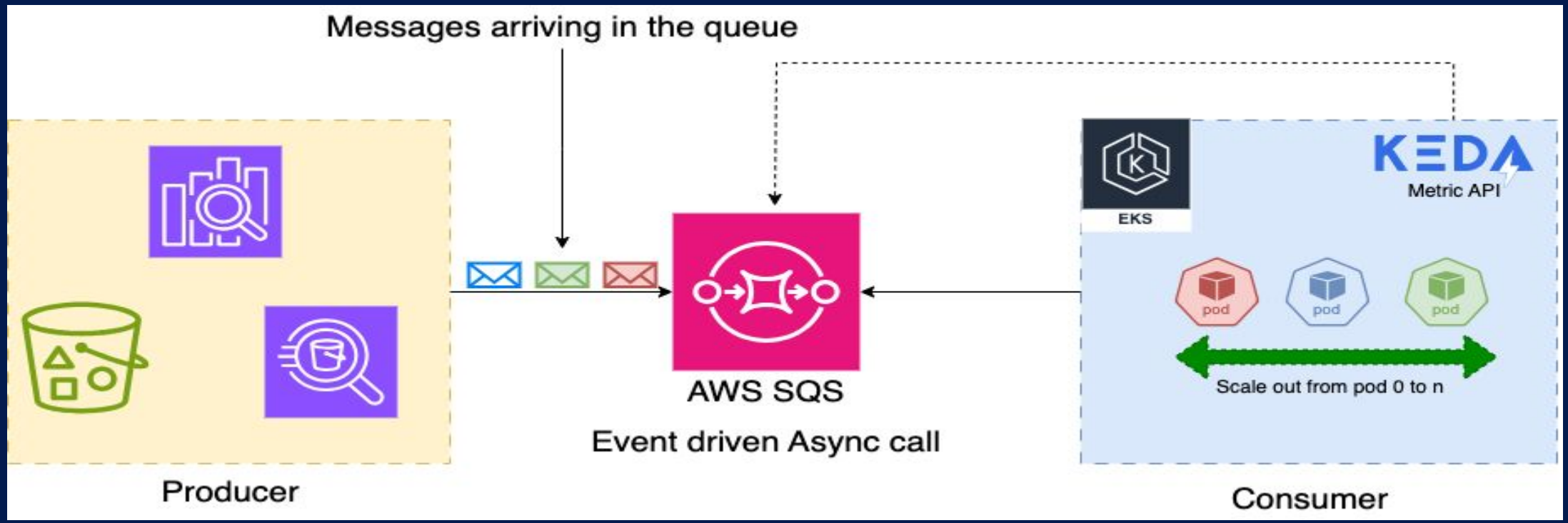



**Use cases of KEDA:-**

- Batch processing jobs
- Serverless workloads
- Scaling microservices app, etc

# KEDA demo using AWS SQS Scaler

Keda demo using AWS SQS as an external source of events bringing the down the application to zero and scale-out to n replica count



# Key Takeaways

- Learned the importance of scaling an application
- Different auto scaling techniques supported by Kubernetes
- Deep dive into HPA (Horizontal Pod Autoscaler) with CPU/Memory based scaling demo
- Architecture of KEDA (Kubernetes Event-Driven Autoscaling)
- Benefits of using KEDA like support for numerous scalers, zero pod scaling saving cost, etc with demo using AWS SQS Queue based Scaler

# Reference Links

https://keda.sh/

https://github.com/kedacore/keda

https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/

https://github.com/vinod827/k8s-nest

https://www.linkedin.com/in/vinod827/

Thank you for attending the session

Question & Answers