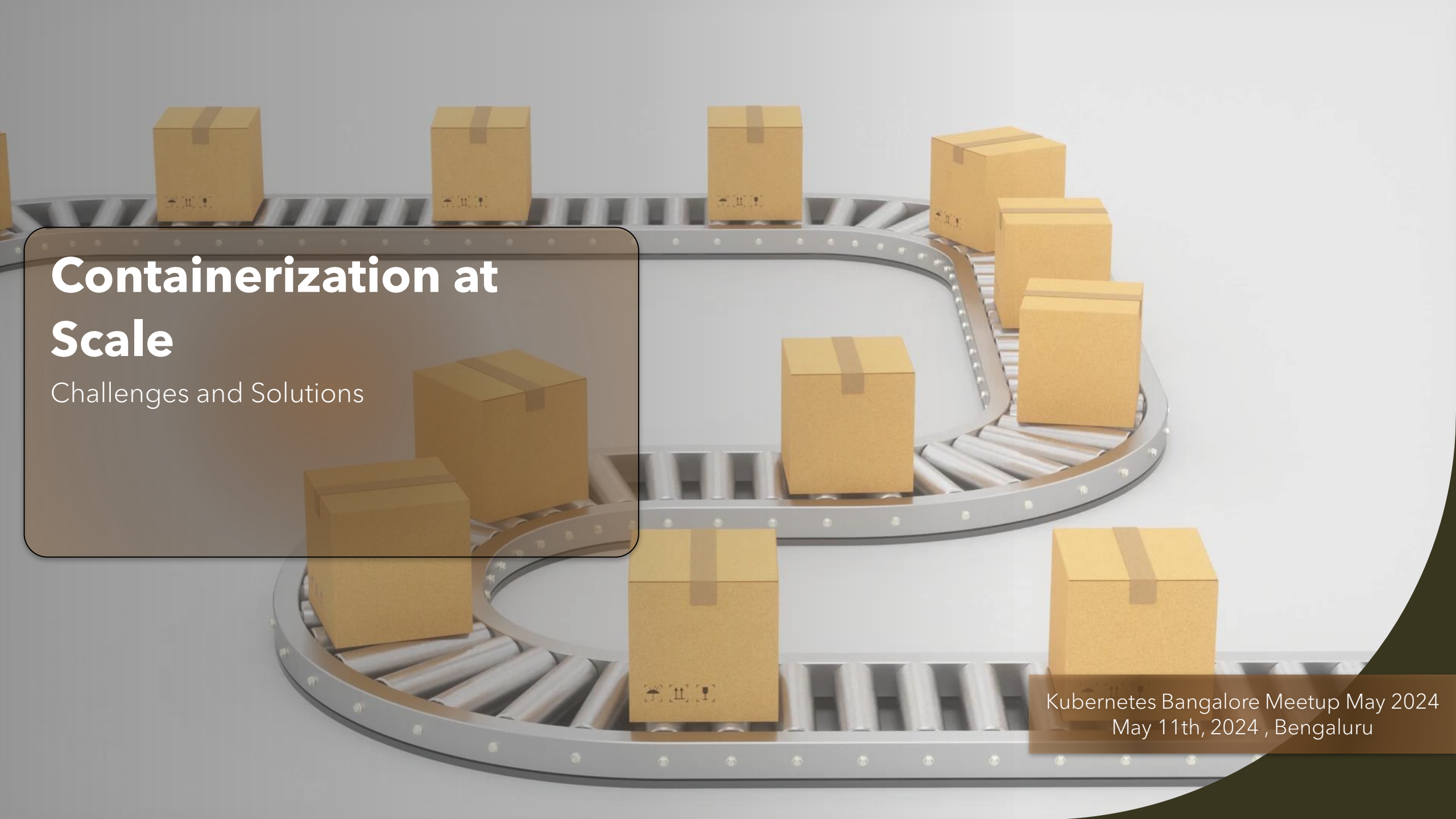


Containerization at Scale

Challenges and Solutions

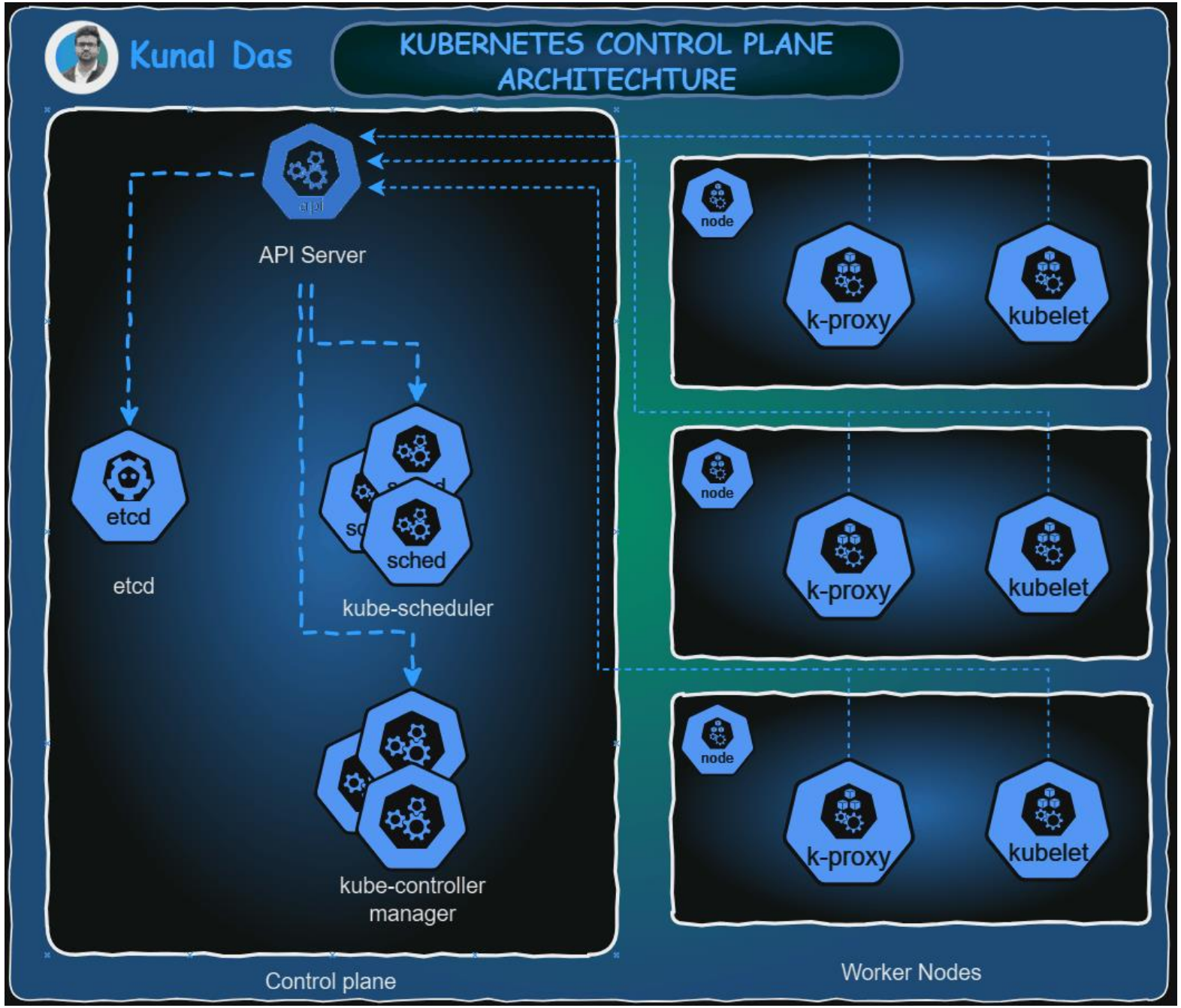
Kubernetes Bangalore Meetup May 2024
May 11th, 2024 , Bengaluru

Kunal Das

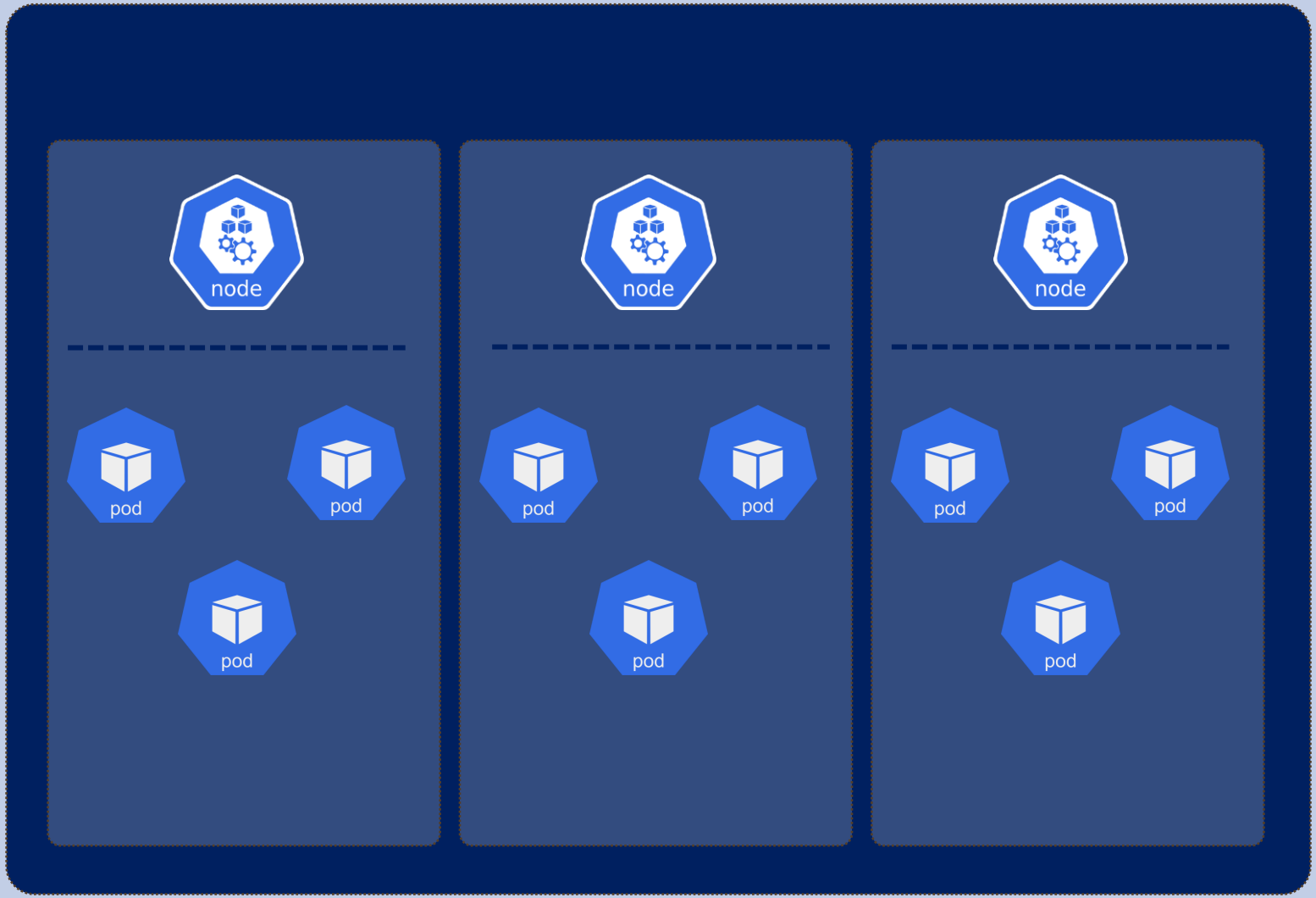
Sr DevOps Engineer @ Cyncly

HashiCorp Certified Terraform Associate

Microsoft Certified DevOps Expert


Kunal Das
KUBERNETES CONTROL PLANE ARCHITECHTURE
api
API Server
etcd
etcd
sched
kube-scheduler
kube-controller manager
Control plane
node
k-proxy
kubelet
node
k-proxy
kubelet
node
k-proxy
kubelet
Worker Nodes

Node Pool



Auto Scaling

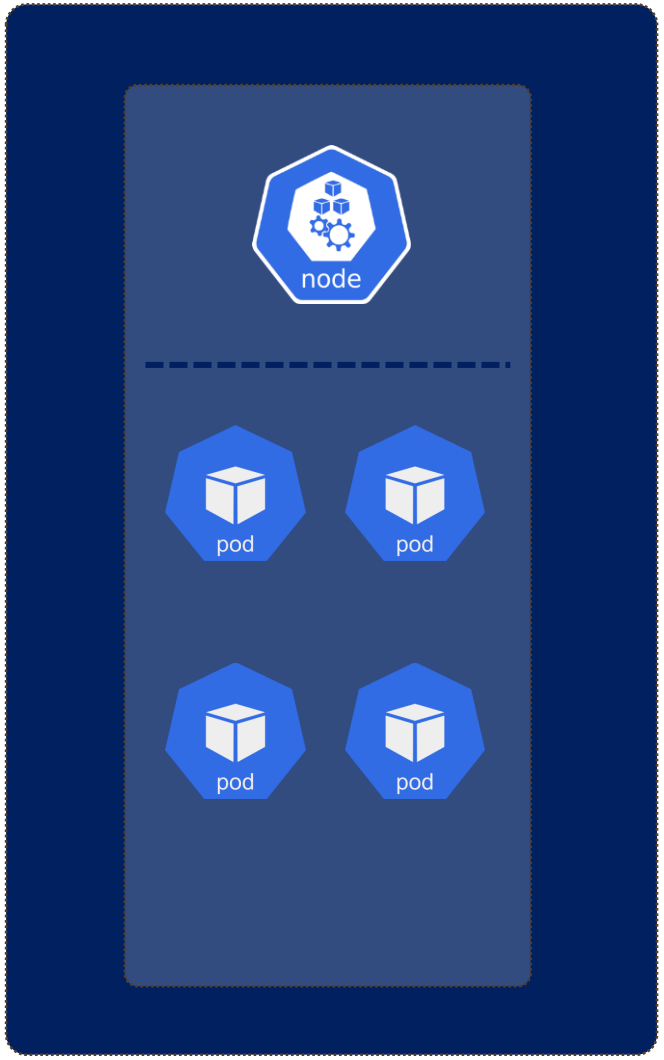
Cluster Autoscaler / Node Autoscaling

- ✓ Automatic Scaling
- ✓ Cost Efficiency
- ✓ Resource Optimization
- ✓ Integration with Cloud Providers

Pod Autoscaling

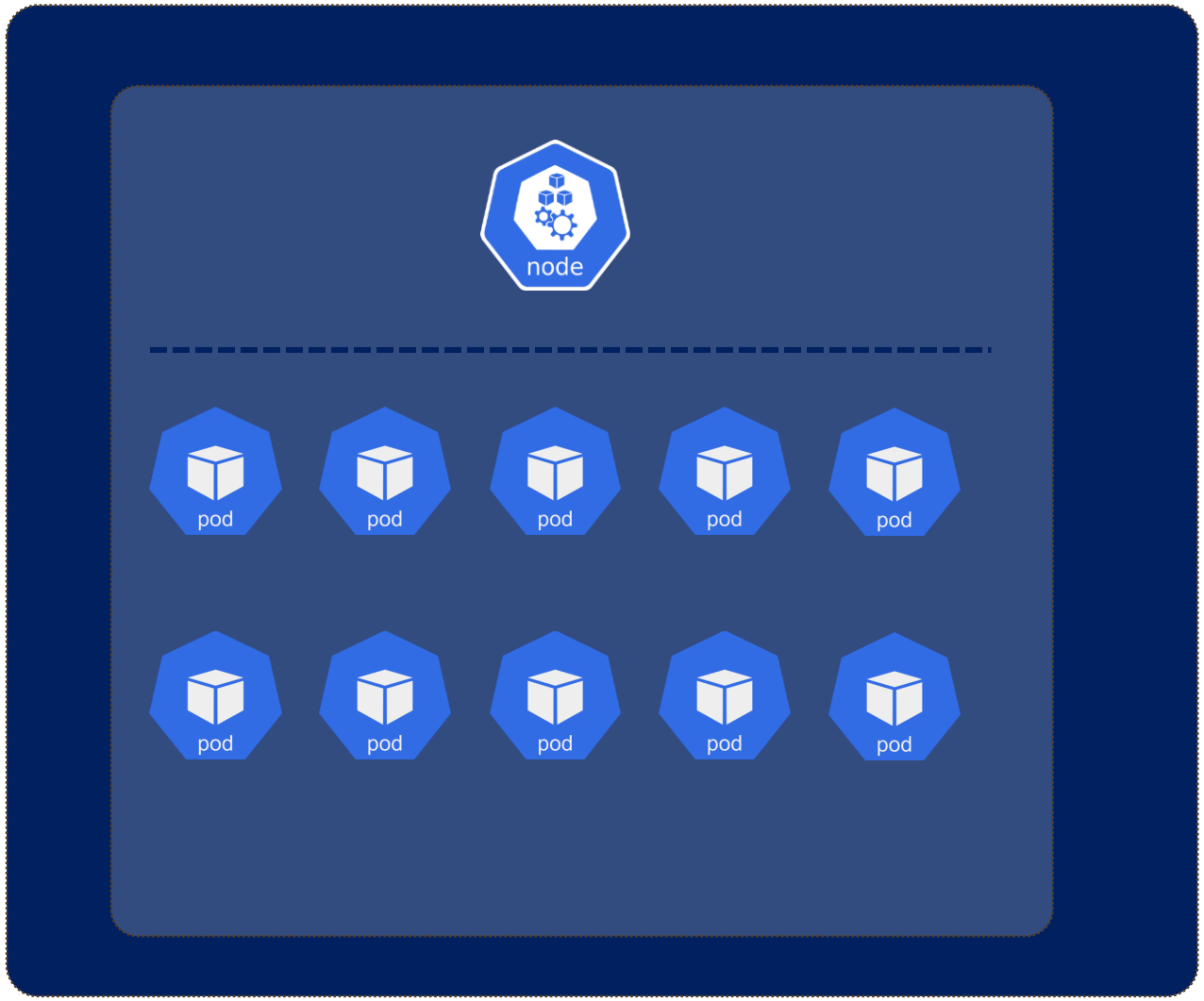
- ✓ Performance Management
- ✓ Response to Traffic
- ✓ Custom Metrics
- ✓ Configurable Parameters

POD autoscaling with HPA



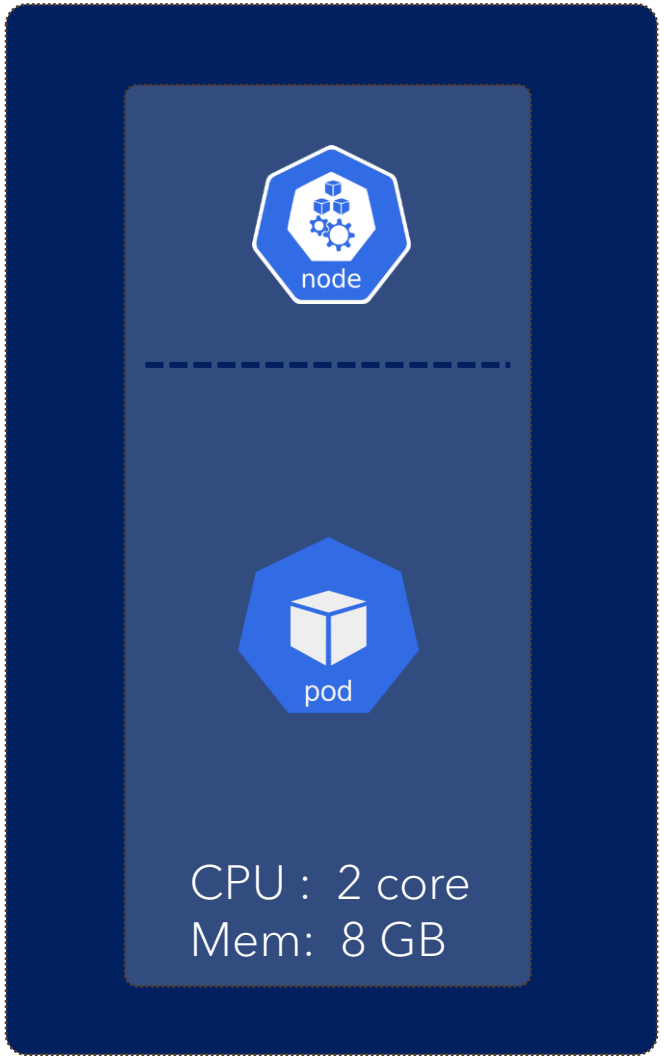
```yaml
apiVersion: autoscaling/v2
kind: HorizontalPodAutoscaler
metadata:
  name: sample-hpa
  namespace: default
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: sample-deployment
  minReplicas: 4
  maxReplicas: 10
  metrics:
    - type: Resource
      resource:
        name: cpu
        target:
          type: Utilization
          averageUtilization: 50
    - type: Resource
      resource:
        name: memory
        target:
          type: Utilization
          averageUtilization: 60
```

POD autoscaling with HPA



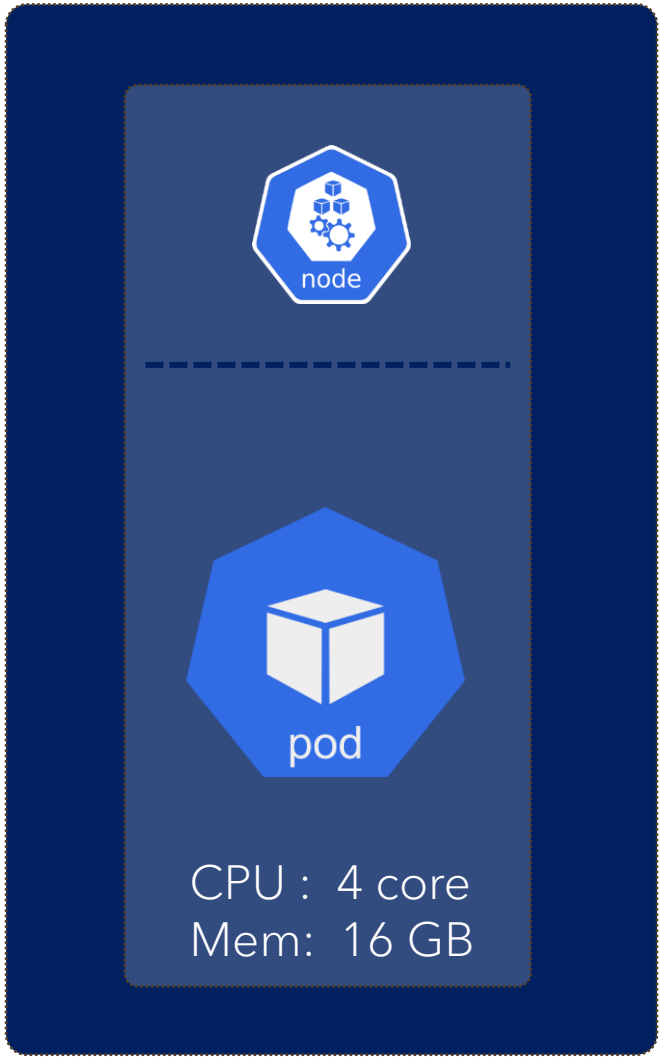

```yaml
apiVersion: autoscaling/v2
kind: HorizontalPodAutoscaler
metadata:
  name: sample-hpa
  namespace: default
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: sample-deployment
  minReplicas: 4
  maxReplicas: 10
  metrics:
    - type: Resource
      resource:
        name: cpu
        target:
          type: Utilization
          averageUtilization: 50
    - type: Resource
      resource:
        name: memory
        target:
          type: Utilization
          averageUtilization: 60
```

POD autoscaling with VPA



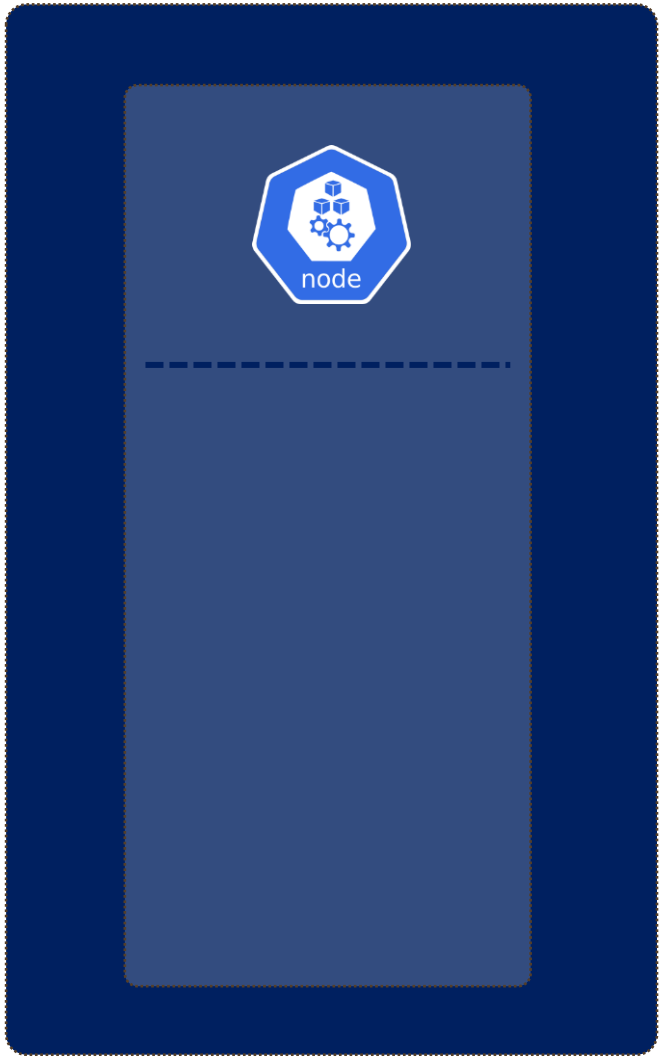
```yaml
apiVersion: autoscaling.k8s.io/v1
kind: VerticalPodAutoscaler
metadata:
  name: sample-vpa
  namespace: default
spec:
  targetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: sample-deployment
  resourcePolicy:
    containerPolicies:
      - containerName: '*'
        minAllowed:
          cpu: '2'
          memory: '8Gi'
        maxAllowed:
          cpu: '4'
          memory: '16Gi'
        controlledResources: ['cpu', 'memory']
  updatePolicy:
    updateMode: Auto
```

POD autoscaling with VPA




```yaml
apiVersion: autoscaling.k8s.io/v1
kind: VerticalPodAutoscaler
metadata:
  name: sample-vpa
  namespace: default
spec:
  targetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: sample-deployment
  resourcePolicy:
    containerPolicies:
      - containerName: '*'
        minAllowed:
          cpu: '2'
          memory: '8Gi'
        maxAllowed:
          cpu: '4'
          memory: '16Gi'
        controlledResources: ['cpu', 'memory']
  updatePolicy:
    updateMode: Auto
```

KEDA

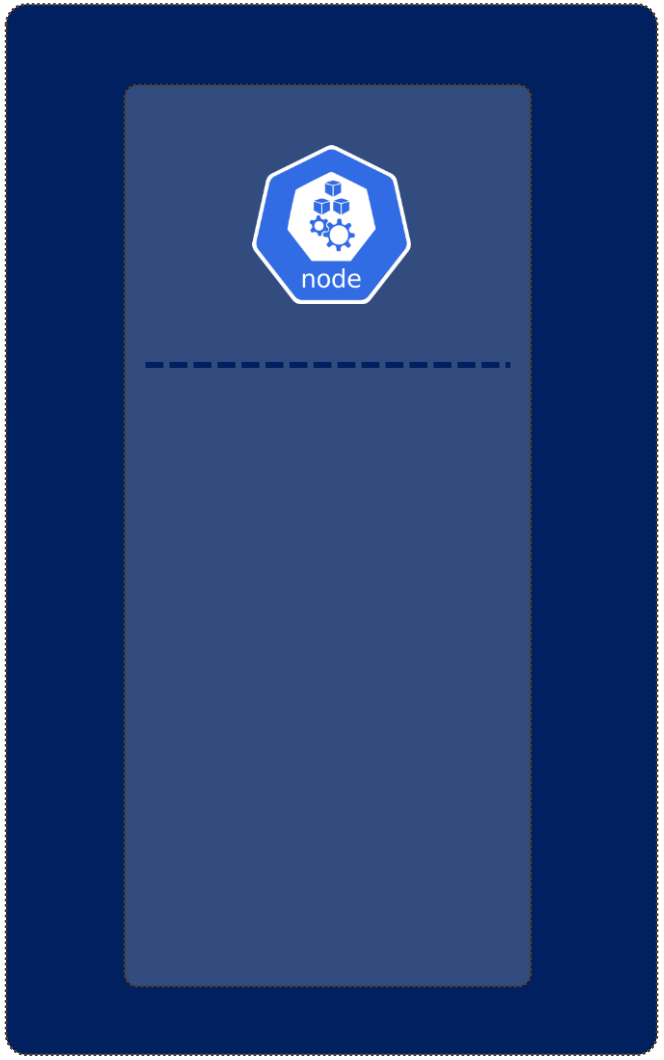


- ✓ Event-Driven Scaling
- ✓ Supports Multiple Triggers
- ✓ Seamless Integration
- ✓ Scale down to zero
- ✓ Cloud-Native Focus

```yaml
apiVersion: keda.sh/v1alpha1
kind: ScaledObject
metadata:
  name: http-scaled-object
  namespace: default
spec:
  scaleTargetRef:
    kind: Deployment
    name: http-web-app
  minReplicaCount: 0
  maxReplicaCount: 10
  pollingInterval: 5
  cooldownPeriod: 30
  triggers:
    metadata:
      targetValue: "100"
```

Ref : https://keda.sh/

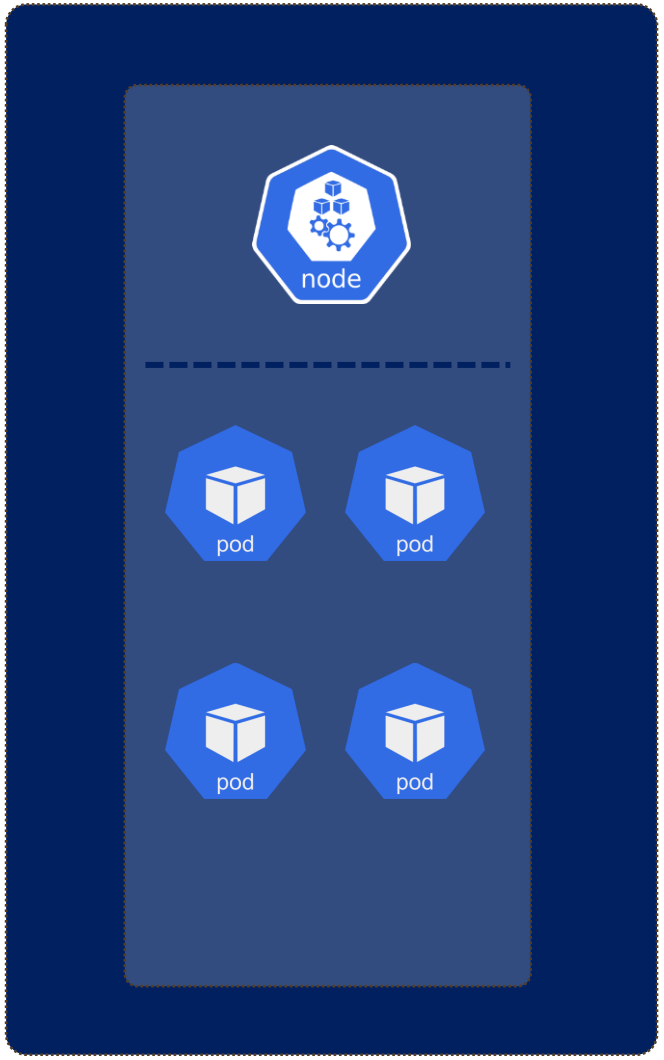
KEDA



```yaml
apiVersion: keda.sh/v1alpha1
kind: ScaledObject
metadata:
  name: http-scaled-object
  namespace: default
spec:
  scaleTargetRef:
    kind: Deployment
    name: http-web-app
  minReplicaCount: 0
  maxReplicaCount: 10
  pollingInterval: 5
  cooldownPeriod: 30
  triggers:
    metadata:
      targetValue: "100"
```

Ref : https://keda.sh/

KEDA



```yaml
apiVersion: keda.sh/v1alpha1
kind: ScaledObject
metadata:
  name: http-scaled-object
  namespace: default
spec:
  scaleTargetRef:
    kind: Deployment
    name: http-web-app
  minReplicaCount: 0
  maxReplicaCount: 10
  pollingInterval: 5
  cooldownPeriod: 30
  triggers:
    metadata:
      targetValue: "100"
```

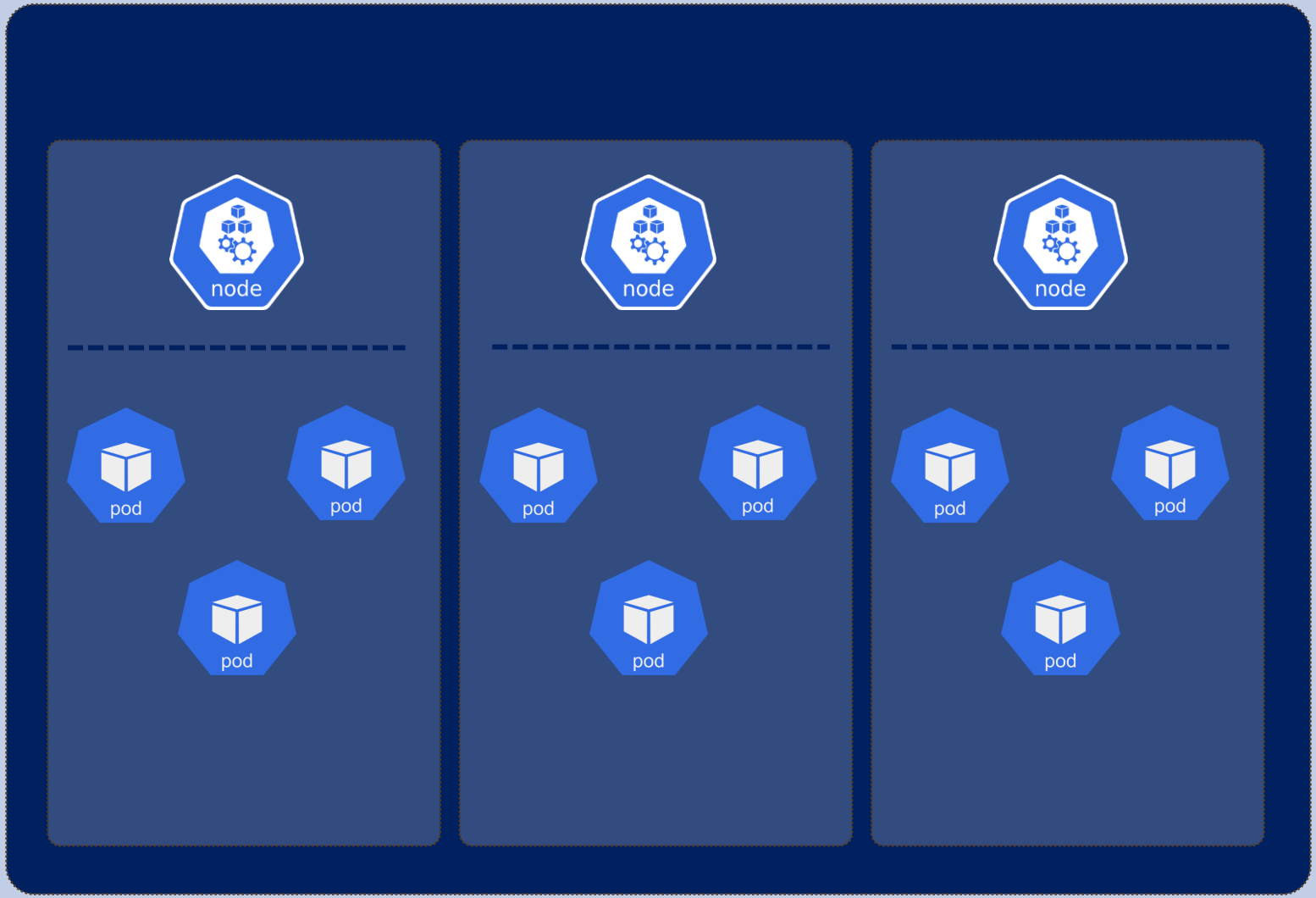
Ref : https://keda.sh/

Other autoscaling techniques

(At pod level)

- ✓ Custom Metrics Autoscaling
- ✓ Operator-Based Scaling
- ✓ Many more

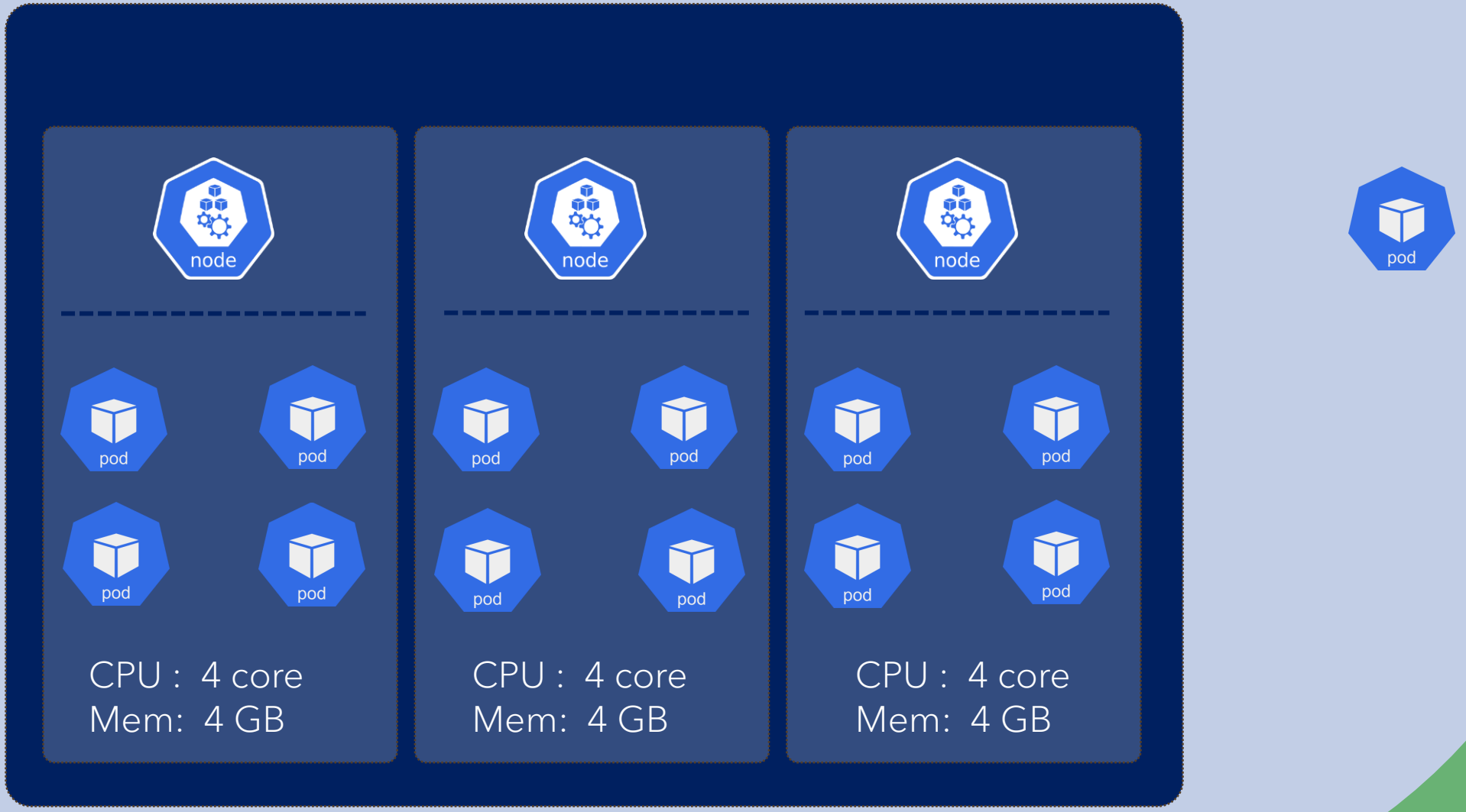
Node Pool



Node Scaling

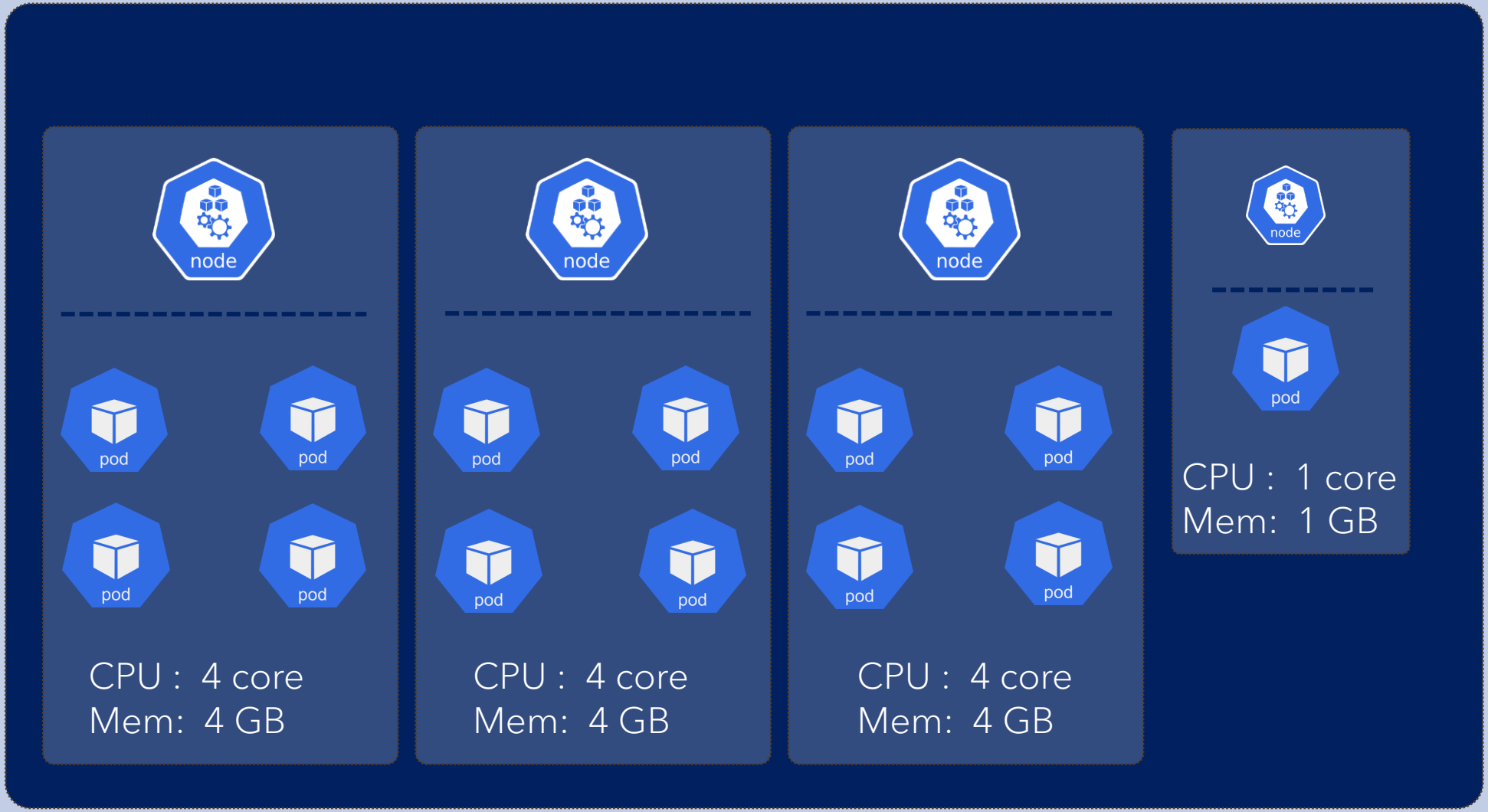
- ✓ Manual Node Scaling
- ✓ Cloud Provider Autoscalers

Node auto provisioning



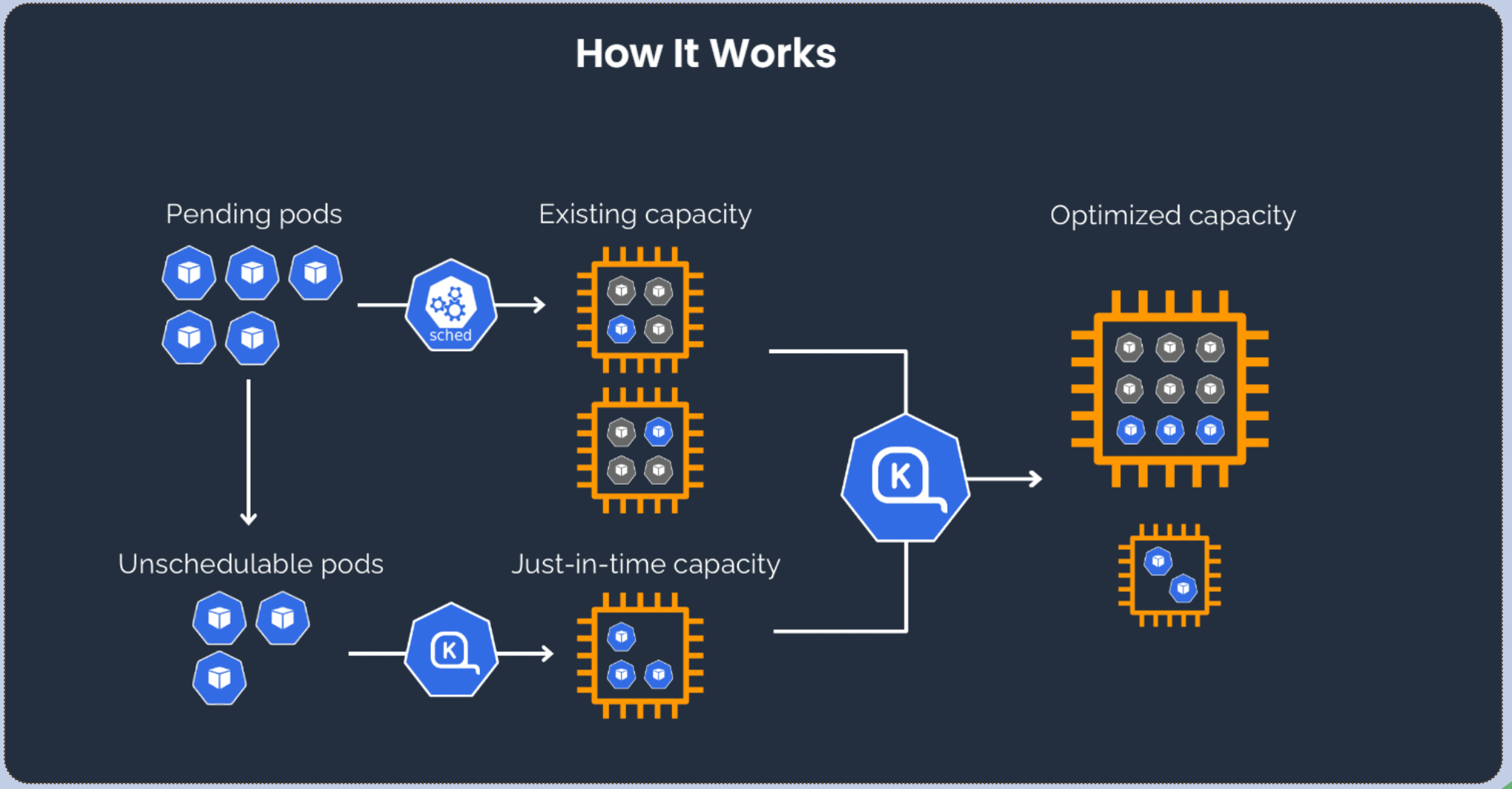
Ref : https://github.com/kubernetes/autoscaler/blob/master/cluster-autoscaler/proposals/node_autoprovisioning.md

Node auto provisioning



Ref : https://github.com/kubernetes/autoscaler/blob/master/cluster-autoscaler/proposals/node_autoprovisioning.md

Node auto provisioning



Ref : https://karpenter.sh/

Pod Scaling

- ✓ Horizontal Pod Autoscaler
- ✓ Vertical Pod Autoscaler
- ✓ Event Driven Autoscaler

Node Scaling

- ✓ Manual Node Scaling
- ✓ Cloud Provider Autoscalers
- ✓ Node Auto Provision
- ✓ Predictive Autoscaler

Newer Scaling Techniques

- ✓ Predictive Autoscaling
- ✓ Priority-Based Autoscaling

Ref :https://learn.microsoft.com/en-us/azure/azure-monitor/autoscale/autoscale-predictive

Ref :https://docs.aws.amazon.com/autoscaling/ec2/userguide/predictive-scaling-graphs.html

Feel free to reach out in case of any doubt!!

Thank you