



KubeCon



CloudNativeCon

Europe 2020

sig-autoscaling deep dive

Virtual

josephburnett@google.com
maciekpytel@google.com

Agenda



1. Workload autoscaling
 - a. HPA lifecycle
 - b. Custom metrics
 - c. HPA scale controls **new**
2. Cluster autoscaling
 - a. Architecture
 - b. How scale-up works - example

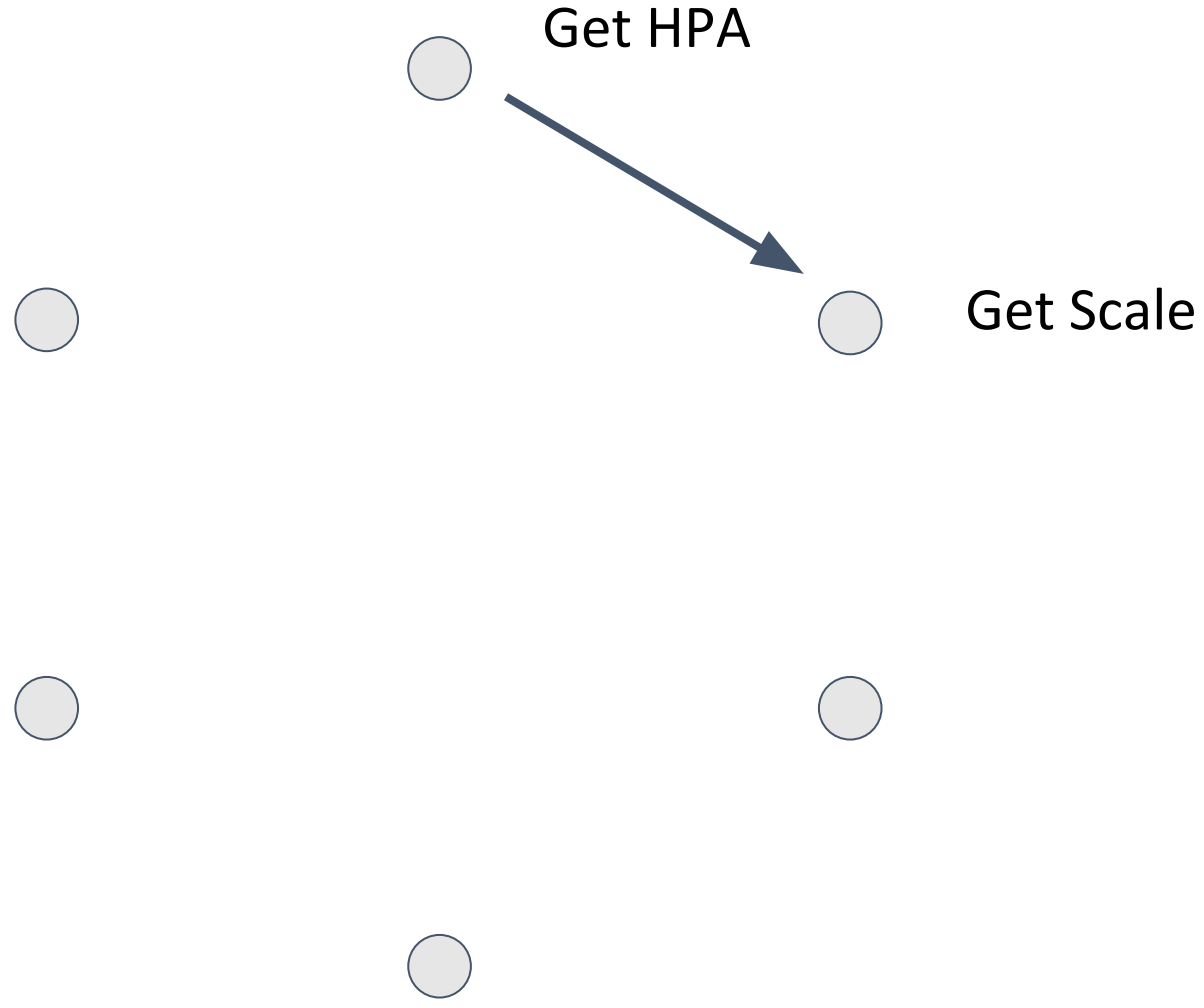
Recommendation Lifecycle



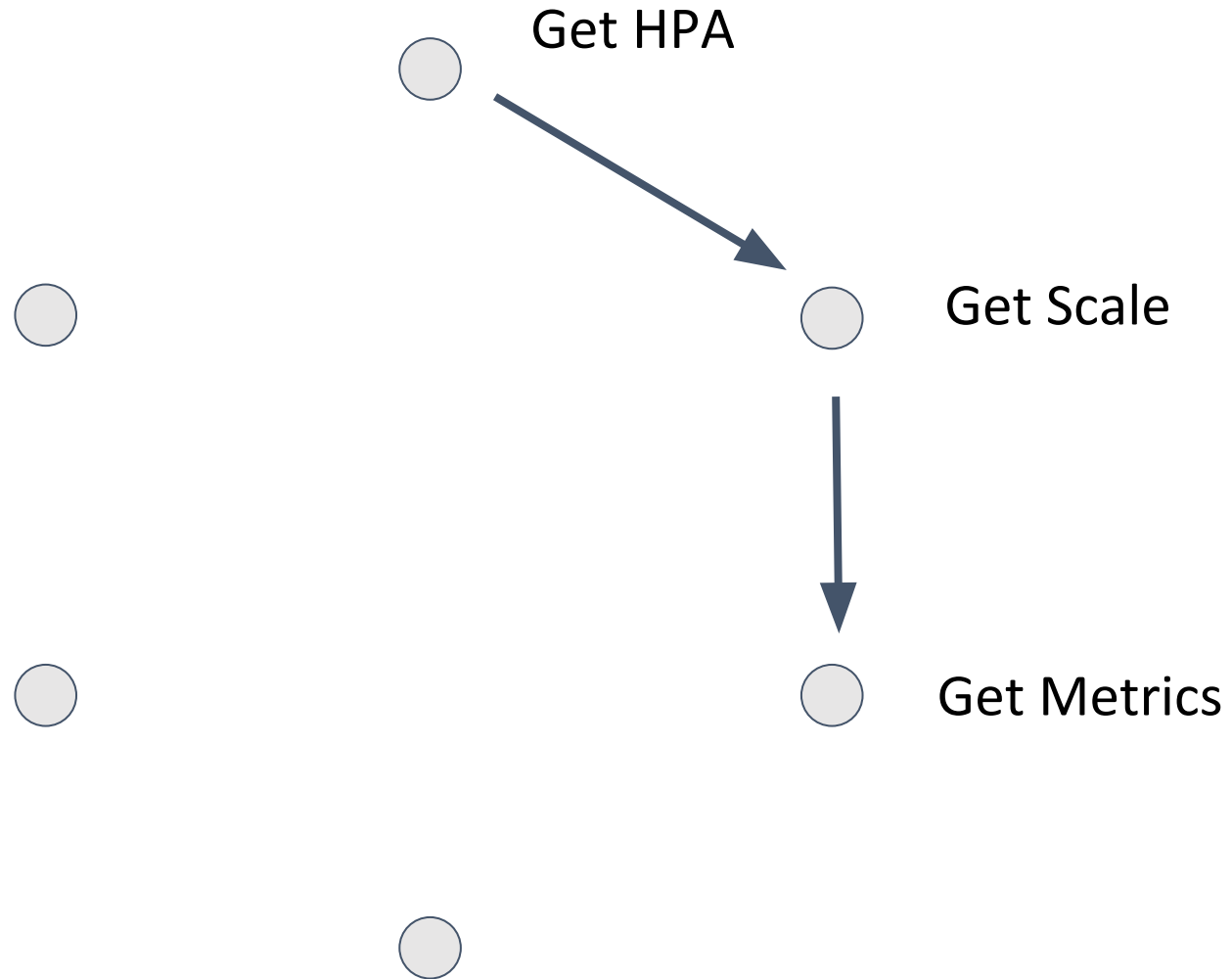
Get HPA



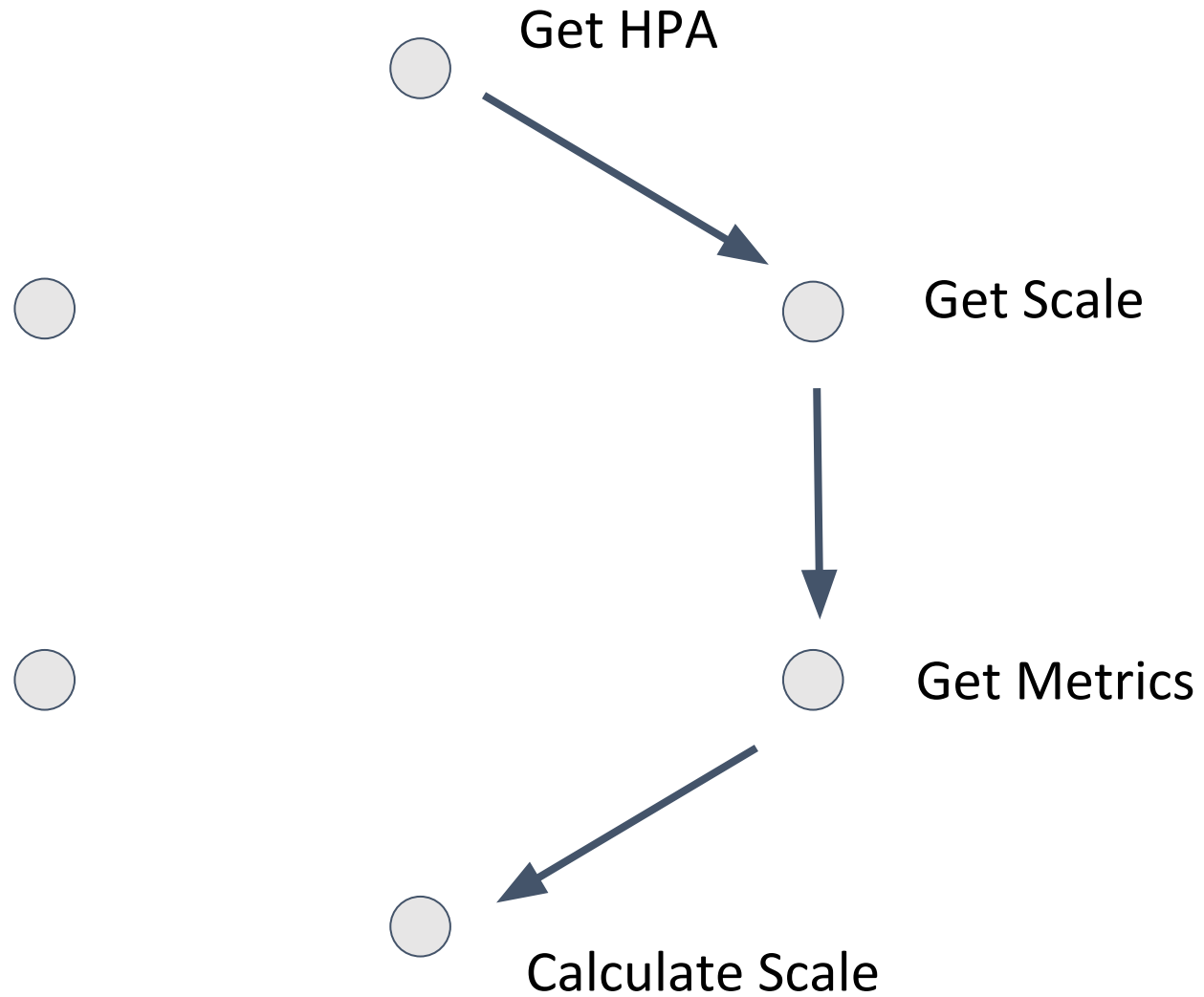
Recommendation Lifecycle



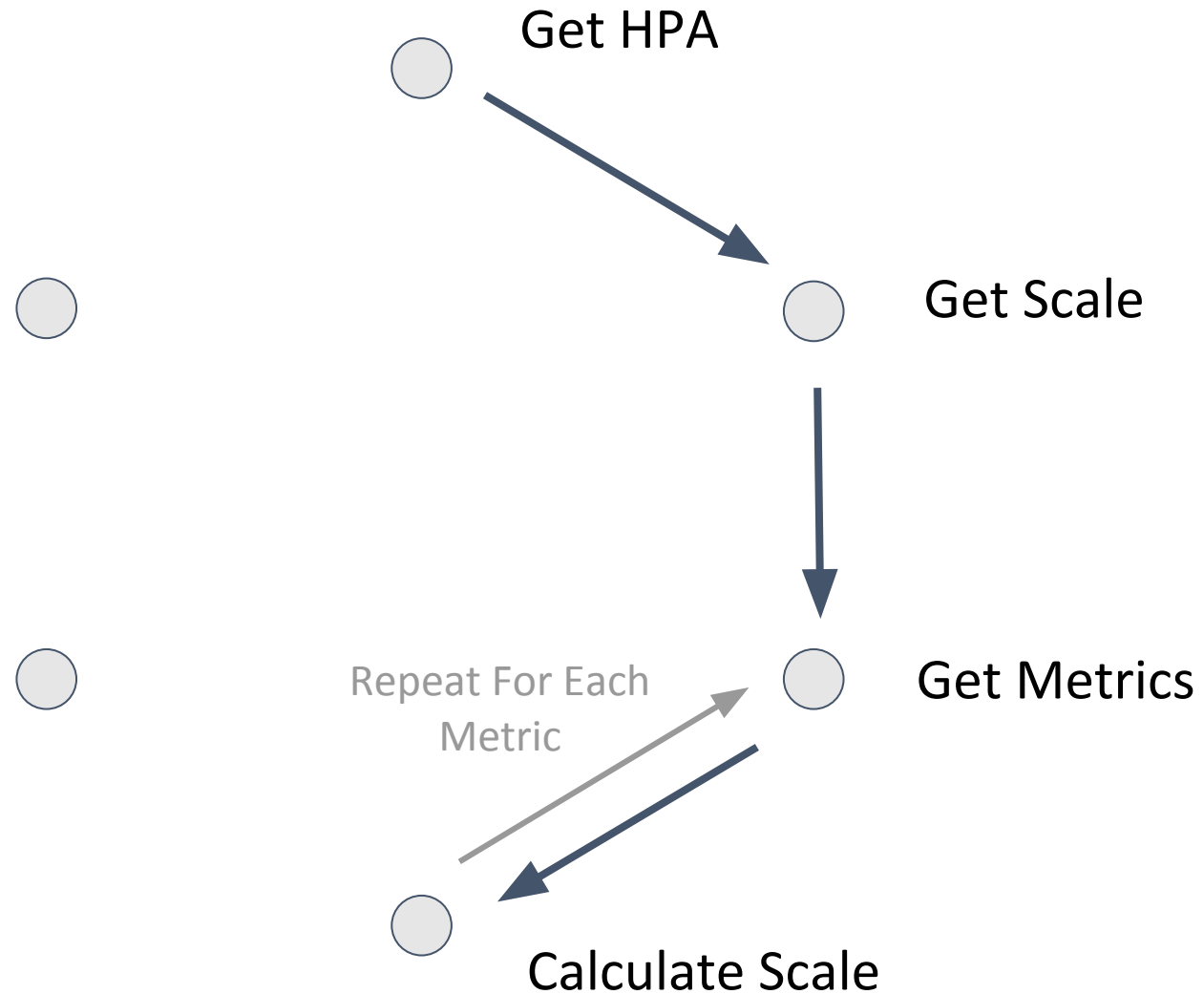
Recommendation Lifecycle



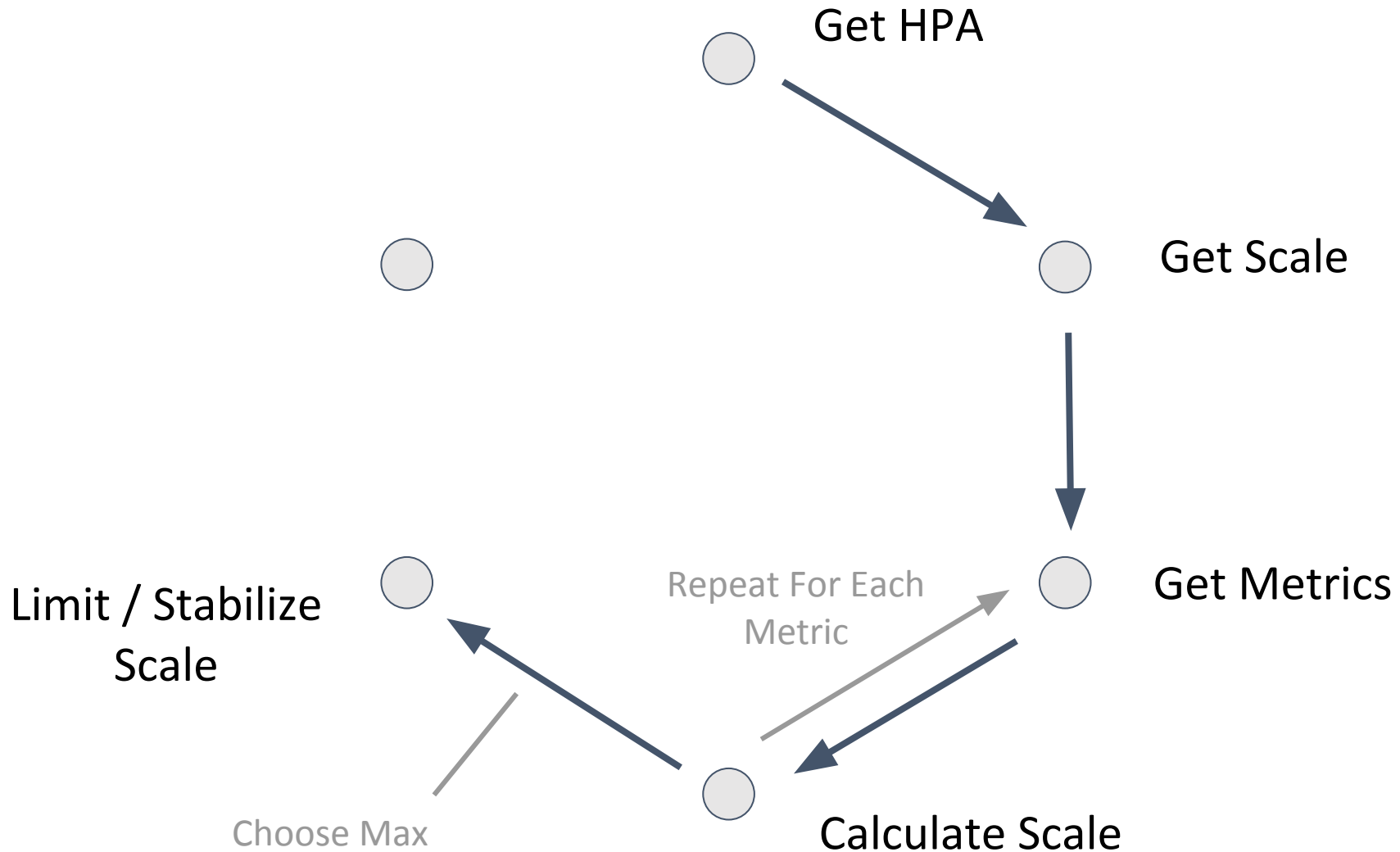
Recommendation Lifecycle



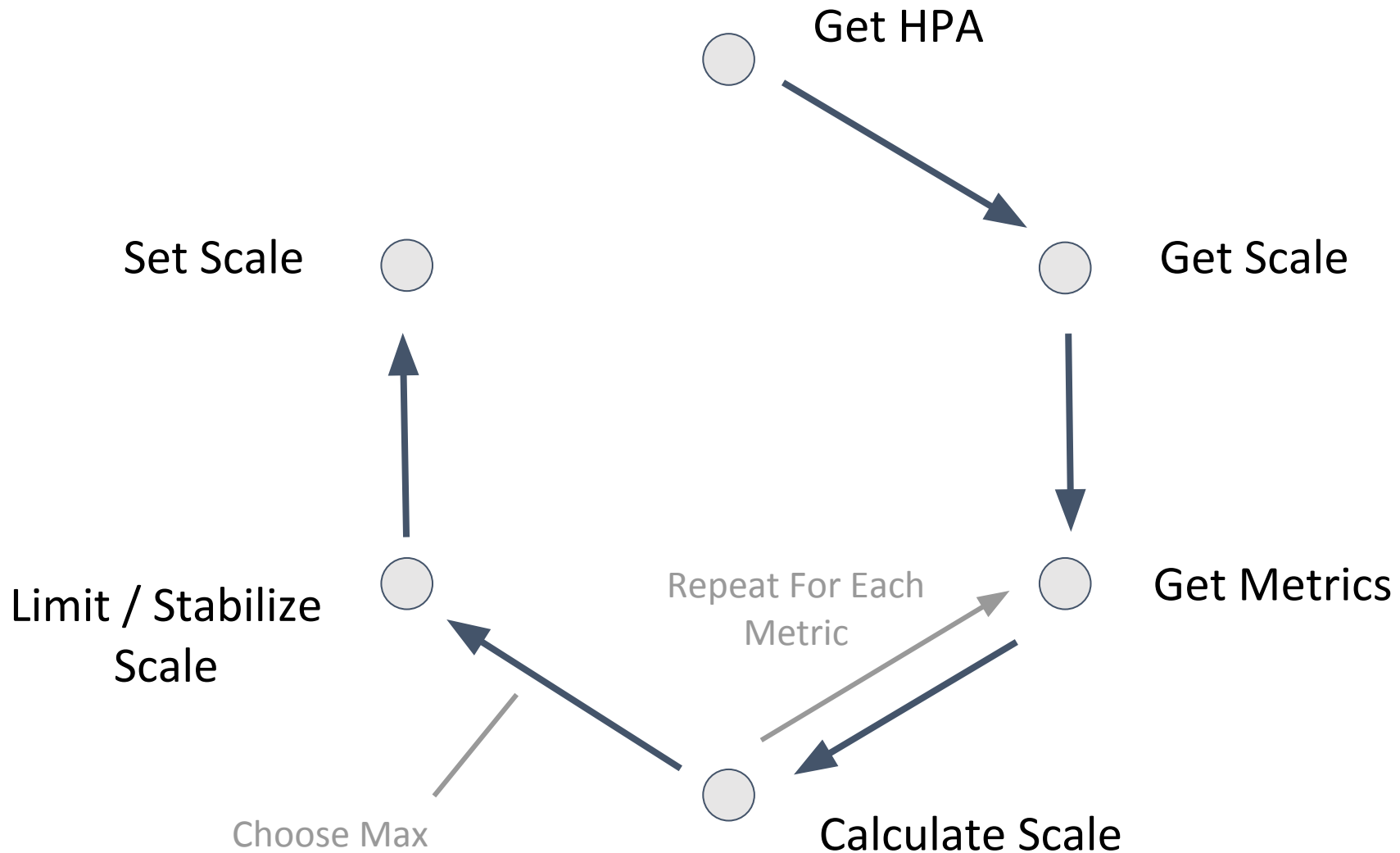
Recommendation Lifecycle



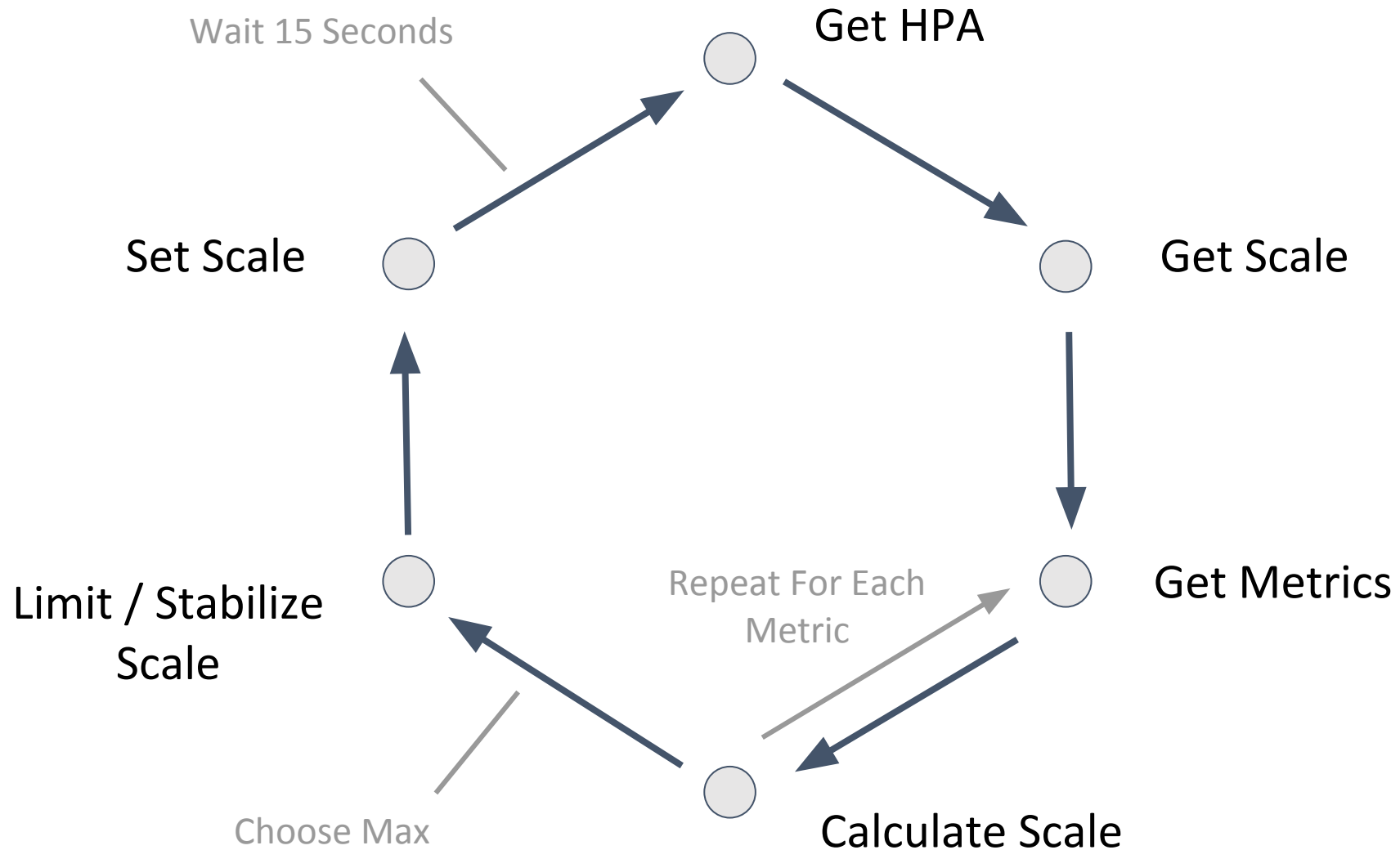
Recommendation Lifecycle



Recommendation Lifecycle



Recommendation Lifecycle



HPA v2



KubeCon



CloudNativeCon

Europe 2020



```
apiVersion: autoscaling/v1
kind: HorizontalPodAutoscaler
metadata:
  annotations:
    autoscaling.alpha.kubernetes.io/conditions:
' [{"type": "AbleToScale", "status": "True", "lastTransitionTime": "2020-07-16T07:38:51Z", "reason": "ScaleDownStabilized", "message": "recent recommendations were higher than current one, applying the highest recent recommendation"}, {"type": "ScalingActive", "status": "True", "lastTransitionTime": "2020-07-16T07:39:51Z", "reason": "ValidMetricFound", "message": "the HPA was able to successfully calculate a replica count from cpu resource utilization (percentage of request)"} ]'
  autoscaling.alpha.kubernetes.io/current-metrics:
' [{"type": "Resource", "resource": {"name": "cpu", "currentAverageUtilization": 0, "currentAverageValue": "1m"}} ]'
creationTimestamp: "2020-07-16T07:38:30Z"
name: php-apache
namespace: default
resourceVersion: "1320"
selfLink: /apis/autoscaling/v1/namespaces/default/horizontalpodautoscalers/php-apache
uid: 8073cc59-9d0a-4a5c-9304-be95311bf95d
spec:
  maxReplicas: 10
  minReplicas: 1
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: php-apache
  targetCPUUtilizationPercentage: 50
status:
  currentCPUUtilizationPercentage: 0
  currentReplicas: 1
  desiredReplicas: 1
```

v1

```
kubectl get hpa
```

```
-oyaml
```

```
apiVersion: autoscaling/v2beta2
kind: HorizontalPodAutoscaler
metadata:
  creationTimestamp: "2020-07-16T07:38:30Z"
  name: php-apache
  namespace: default
  resourceVersion: "1320"
  selfLink: /apis/autoscaling/v2beta2/namespaces/default/horizontalpodautoscalers/php-apache
  uid: 8073cc59-9d0a-4a5c-9304-be95311bf95d
spec:
  maxReplicas: 10
  metrics:
  - resource:
    name: cpu
    target:
      averageUtilization: 50
      type: Utilization
    type: Resource
  minReplicas: 1
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: php-apache
status:
  conditions:
  - lastTransitionTime: "2020-07-16T07:38:51Z"
    message: recent recommendations were higher than current one, applying the highest recent recommendation
    reason: ScaleDownStabilized
    status: "True"
    type: AbleToScale
  - lastTransitionTime: "2020-07-16T07:39:51Z"
    message: the HPA was able to successfully calculate a replica count from cpu resource utilization (percentage of request)
    reason: ValidMetricFound
```

v2

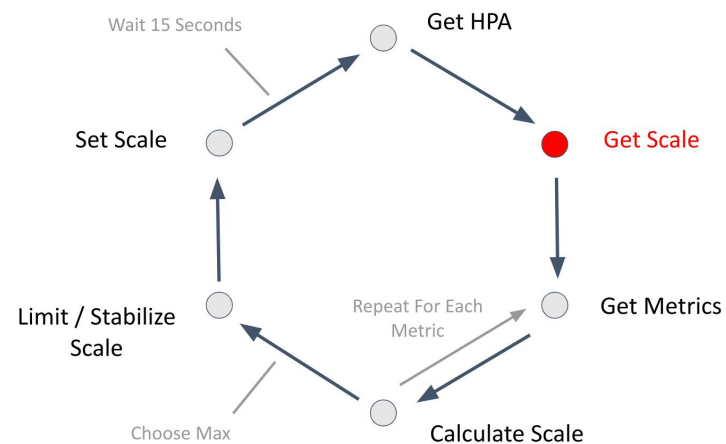
```
kubectl get hpa.v2beta2.autoscaling
```

```
-oyaml
```

```
DesiredWithinRange
status: "False"
type: ScalingLimited
currentMetrics:
- resource:
  current:
    averageUtilization: 0
    averageValue: 1m
    name: cpu
    type: Resource
  currentReplicas: 1
  desiredReplicas: 1
```

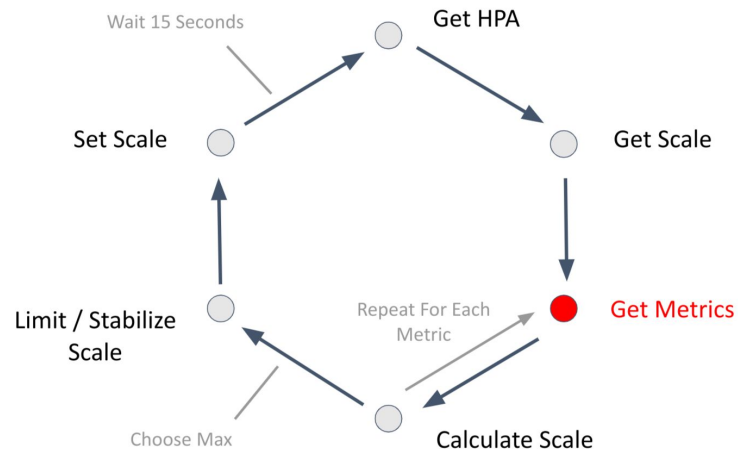
```
apiVersion: autoscaling/v1
kind: HorizontalPodAutoscaler
metadata:
  name: php-apache
  namespace: default
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: php-apache
  ...
```

```
apiVersion: autoscaling/v2beta2
kind: HorizontalPodAutoscaler
metadata:
  name: php-apache
  namespace: default
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: php-apache
  ...
```



```
apiVersion: autoscaling/v1
kind: HorizontalPodAutoscaler
metadata:
  name: php-apache
  namespace: default
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: php-apache
  targetCPUUtilizationPercentage: 50
```

```
apiVersion: autoscaling/v2beta2
kind: HorizontalPodAutoscaler
metadata:
  name: php-apache
  namespace: default
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: php-apache
  metrics:
  - type: Resource
    resource:
      name: cpu
      target:
        averageUtilization: 50
        type: Utilization
```



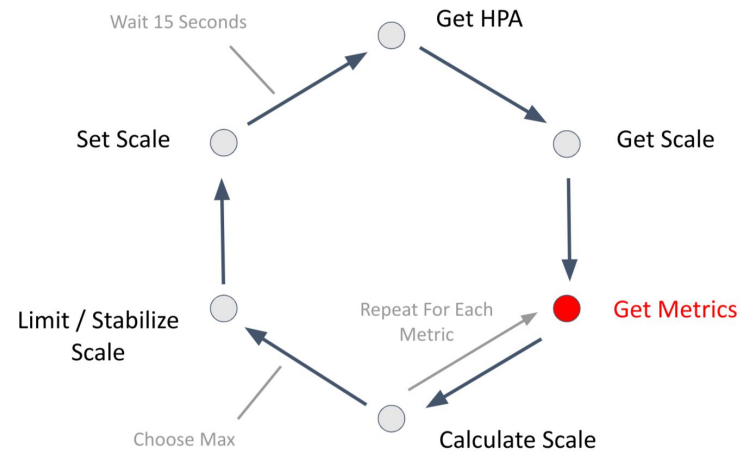
Status

status:

```
currentCPUUtilizationPercentage: 0  
currentReplicas: 1  
desiredReplicas: 1
```

status:

```
currentMetrics:  
- type: Resource  
  resource:  
    current:  
      averageUtilization: 0  
      averageValue: 1m  
    name: cpu  
currentReplicas: 1  
desiredReplicas: 1  
...
```



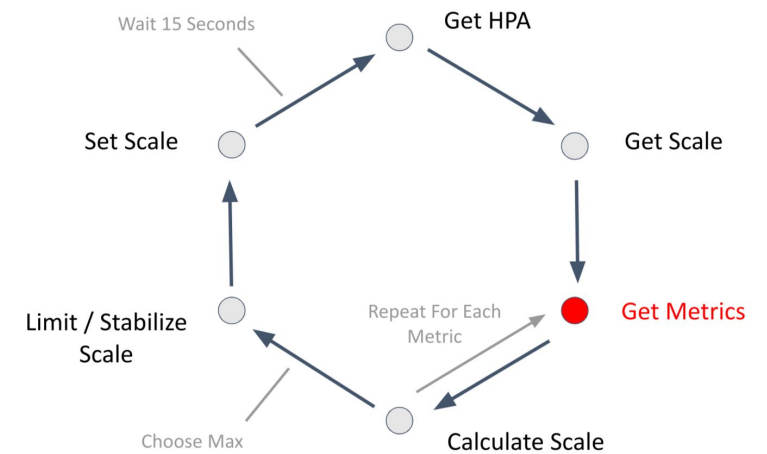
Custom Metrics

custom.metrics.kubernetes.io

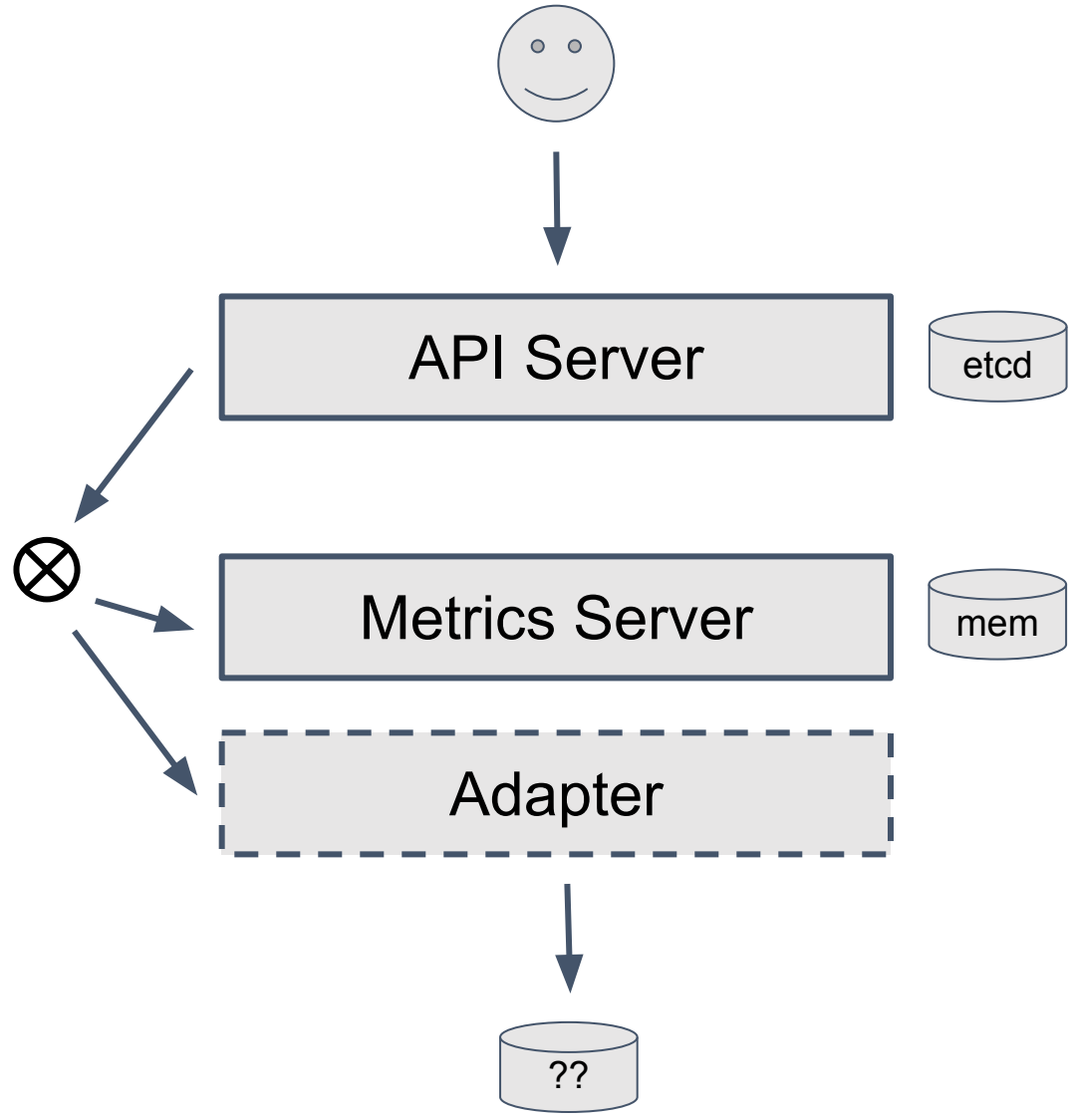
```
...
metrics:
- type: Pods
  pods:
    metricName: custom-metric
    targetAverageValue: 20
- type: Object
  object:
    metric:
      name: requests-per-second
    describedObject:
      apiVersion: networking.k8s.io/v1beta1
      kind: Ingress
      name: main-route
    target:
      type: Value
      value: 2k
...
```

external.metrics.kubernetes.io

```
...
metrics:
- type: External
  external:
    metricName: num_undelivered_messages
    metricSelector:
      matchLabels:
        subscription_id: echo-read
    targetAverageValue: 2
...
```



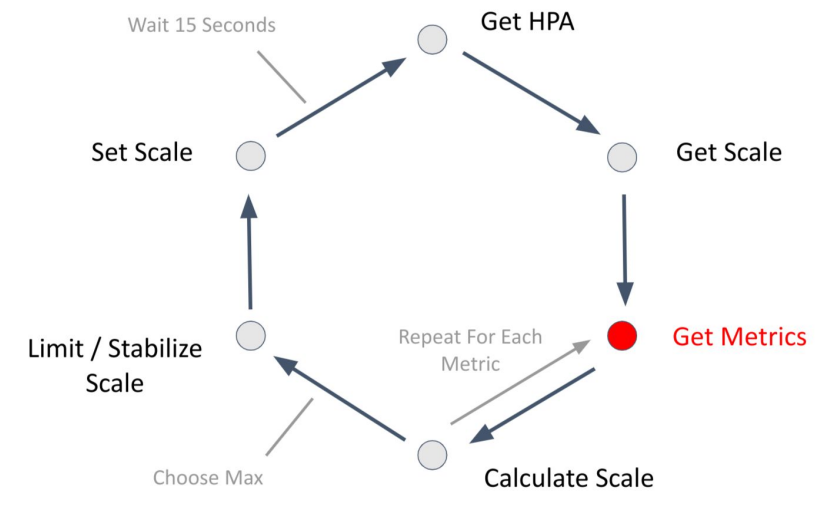
Custom Metrics



`hpa.v2beta2.autoscaling`

`metrics.kubernetes.io`

`custom.metrics.kubernetes.io`
`external.metrics.kubernetes.io`



Scale Controls

...

behavior:

scaleUp:

policies:

- type: **Percent**

value: 200

periodSeconds: 15

- type: **Pods**

value: 4

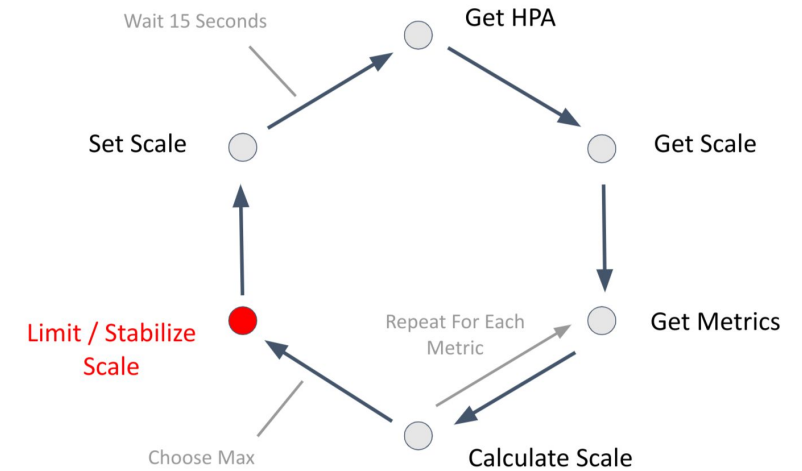
periodSeconds: 15

scaleDown:

stabilizationWindowSeconds: 300

...

defaults



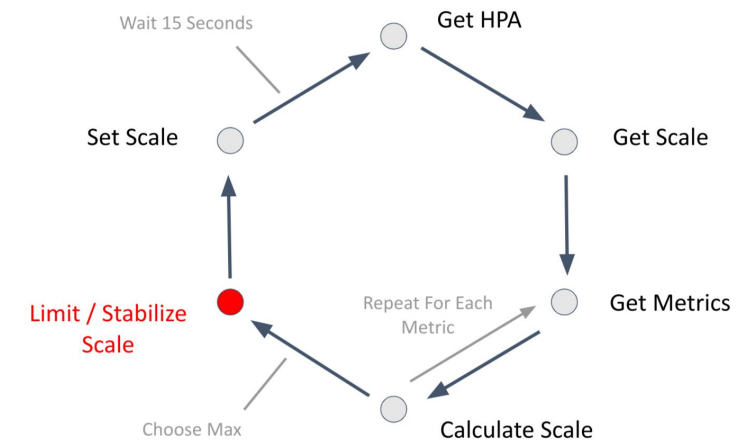
--horizontal-pod-autoscaler-downscale-stabilization=5m

Scale Controls

Same structure for scaleUp and scaleDown controls:

```
type HPAScalingRules struct {  
    // StabilizationWindowSeconds is the number of seconds for which past recommendations should be  
    // considered while scaling up or scaling down.  
    // StabilizationWindowSeconds must be greater than or equal to zero and less than or equal to 3600 (one hour).  
    // If not set, use the default values:  
    // - For scale up: 0 (i.e. no stabilization is done).  
    // - For scale down: 300 (i.e. the stabilization window is 300 seconds long).  
    // +optional  
    StabilizationWindowSeconds *int32 `json:"stabilizationWindowSeconds" protobuf:"varint,3,opt,name=stabilizationWindowSeconds"`  
    // selectPolicy is used to specify which policy should be used.  
    // If not set, the default value MaxPolicySelect is used.  
    // +optional  
    SelectPolicy *ScalingPolicySelect `json:"selectPolicy,omitEmpty" protobuf:"bytes,1,opt,name=selectPolicy"`  
    // policies is a list of potential scaling policies which can be used during scaling.  
    // At least one policy must be specified, otherwise the HPAScalingRules will be discarded as invalid  
    // +optional  
    Policies []HPAScalingPolicy `json:"policies,omitEmpty" protobuf:"bytes,2,rep,name=policies"`  
}
```

```
type HPAScalingPolicy struct {  
    // Type is used to specify the scaling policy.  
    Type HPAScalingPolicyType `json:"type" protobuf:"bytes,1,opt,name=type,casttype=HPAScalingPolicyType"`  
    // Value contains the amount of change which is permitted by the policy.  
    // It must be greater than zero  
    Value int32 `json:"value" protobuf:"varint,2,opt,name=value"`  
    // PeriodSeconds specifies the window of time for which the policy should hold true.  
    // PeriodSeconds must be greater than zero and less than or equal to 1800 (30 min).  
    PeriodSeconds int32 `json:"periodSeconds" protobuf:"varint,3,opt,name=periodSeconds"`  
}
```





A big thanks to Ivan Glushkov and Arjun Naik
for the KEP and implementation.

Status Continued



```
status:  
  ...
```

```
status:  
  ...  
  conditions:  
  - lastTransitionTime: ...  
    message: ...  
      recent recommendation  
    reason: ScaleDownStabilized  
    status: "True"  
    type: AbleToScale  
  - lastTransitionTime: ...  
    message: ...  
    reason: ValidMetricFound  
    status: "True"  
    type: ScalingActive  
  - lastTransitionTime: ...  
    message: ...  
    reason: DesiredWithinRange  
    status: "False"  
    type: ScalingLimited
```

Conditions



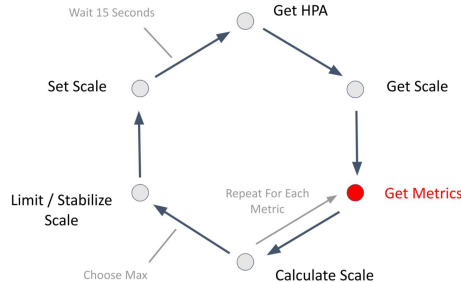
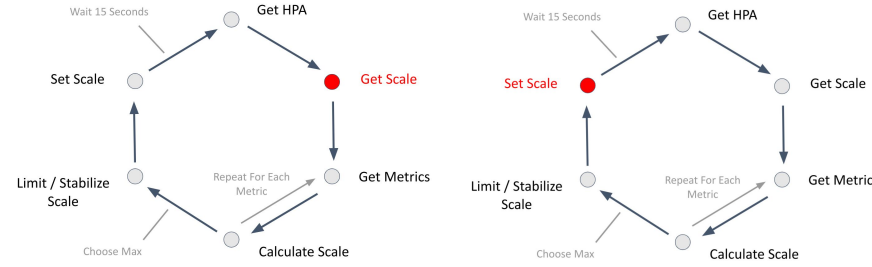
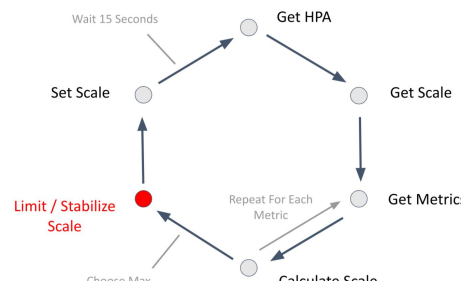
KubeCon



CloudNativeCon

Europe 2020

Virtual

Type	Meaning	Phase
ScalingActive	Target metrics are valid. Metrics can be retrieved.	 <p style="text-align: right; color: red;">Get Metrics</p>
AbleToScale	The scale target ref is valid. Scale be retrieved and updated.	 <p style="text-align: right; color: red;">Get Scale Set Scale</p>
ScalingLimited	The desired replica count is being limited. This could be min / max, downscale stabilization, or scale controls behavior.	 <p style="text-align: right; color: red;">Stabilize / Limit Scale</p>

V1-V2 Conversion



KubeCon



CloudNativeCon

Europe 2020



```
apiVersion: autoscaling/v1
kind: HorizontalPodAutoscaler
metadata:
  annotations:
    autoscaling.alpha.kubernetes.io/conditions:
      '[{"type":"AbleToScale","status":"True","lastTransitionTime":"2020-07-16T07:38:51Z","reason":"ScaleDownStabilized",
"message":"recent
recommendations were higher than current one, applying the highest recent
recommendation"}, {"type":"ScalingActive","status":"True","lastTransitionTime":"2020-07-16T07:39:51Z","reason":"ValidMetricFound",
"message":"the
HPA was able to successfully calculate a replica count from cpu resource utilization
(percentage of
request)"}, {"type":"ScalingLimited","status":"False","lastTransitionTime":"2020-07-16T07:39:51Z","reason":"DesiredWithinRange",
"message":"the
desired count is within the acceptable range"}]'
    autoscaling.alpha.kubernetes.io/current-metrics:
      '[{"type":"Resource","resource":{"name":"cpu","currentAverageUtilization":0,"currentAverageValue":"1m"}}]'
  creationTimestamp: "2020-07-16T07:38:30Z"
  name: php-apache
  namespace: default
  resourceVersion: "1320"
  selfLink: /apis/autoscaling/v1/namespaces/default/horizontalpodautoscalers/php-apache
  uid: 8073cc59-9d0a-4a5c-9304-be95311bf95d
spec:
  maxReplicas: 10
  minReplicas: 1
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: php-apache
  targetCPUUtilizationPercentage: 50
status:
  currentCPUUtilizationPercentage: 0
  currentReplicas: 1
  desiredReplicas: 1
```

```
apiVersion: autoscaling/v2beta2
kind: HorizontalPodAutoscaler
metadata:
  creationTimestamp: "2020-07-16T07:38:30Z"
  name: php-apache
  namespace: default
  resourceVersion: "1320"
  selfLink: /apis/autoscaling/v2beta2/namespaces/default/horizontalpodautoscalers/php-apache
  uid: 8073cc59-9d0a-4a5c-9304-be95311bf95d
spec:
  maxReplicas: 10
  metrics:
  - resource:
    name: cpu
    target:
      averageUtilization: 50
      type: Utilization
    type: Resource
  minReplicas: 1
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: php-apache
status:
  conditions:
  - lastTransitionTime: "2020-07-16T07:38:51Z"
    message: recent recommendations were higher than current one, applying the highest
    recent recommendation
    reason: ScaleDownStabilized
    status: "True"
    type: AbleToScale
  - lastTransitionTime: "2020-07-16T07:39:51Z"
    message: the HPA was able to successfully calculate a replica count from cpu resource
    utilization (percentage of request)
    reason: ValidMetricFound
    status: "True"
    type: ScalingActive
  - lastTransitionTime: "2020-07-16T07:39:51Z"
    message: the desired count is within the acceptable range
    reason: DesiredWithinRange
    status: "False"
    type: ScalingLimited
  currentMetrics:
  - resource:
    current:
      averageUtilization: 0
      averageValue: 1m
      name: cpu
    type: Resource
  currentReplicas: 1
  desiredReplicas: 1
```



Cluster Autoscaler



KubeCon



CloudNativeCon

Europe 2020

Virtual

- Provides nodes so that every pod in cluster can schedule.
- Compacts and removes underutilized nodes.
- Based on scheduling simulations and declared pod requests, not on metrics.
- CA operates on NodeGroups - resizable sets of identical nodes.
 - NodeGroup is implemented differently by each provider (ex. ASG in AWS, MIG in GCE, MachineSet or MachineDeployment in Cluster API).

Architecture



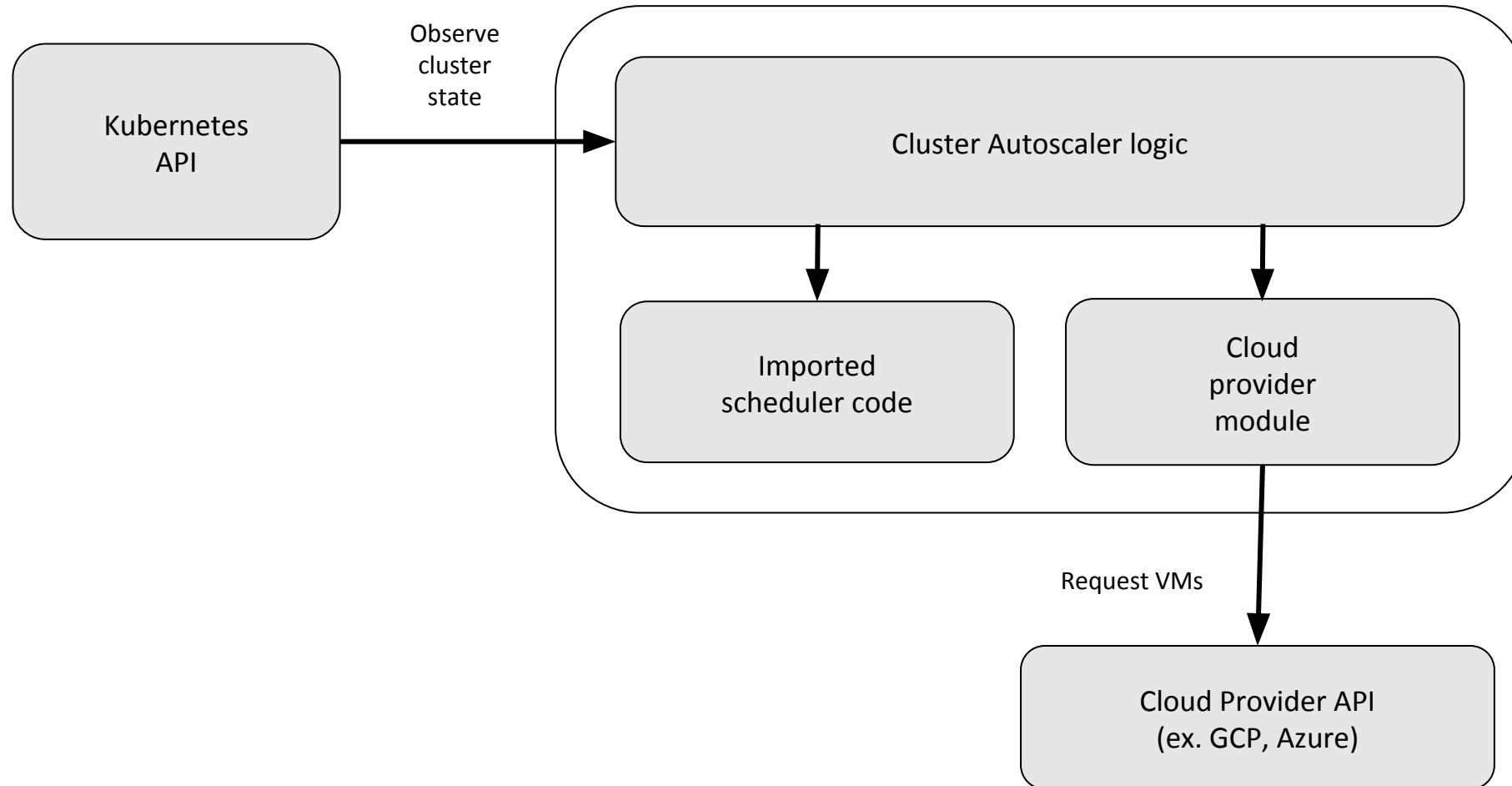
KubeCon



CloudNativeCon

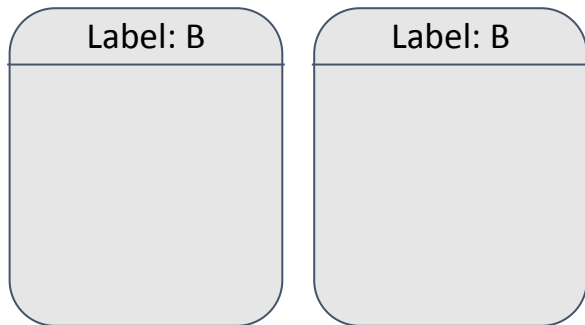
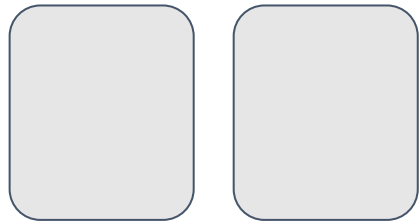
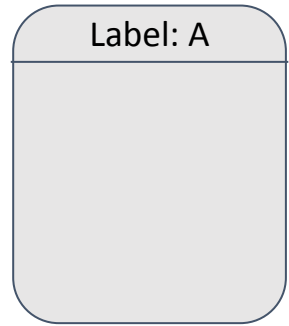
Europe 2020

Virtual

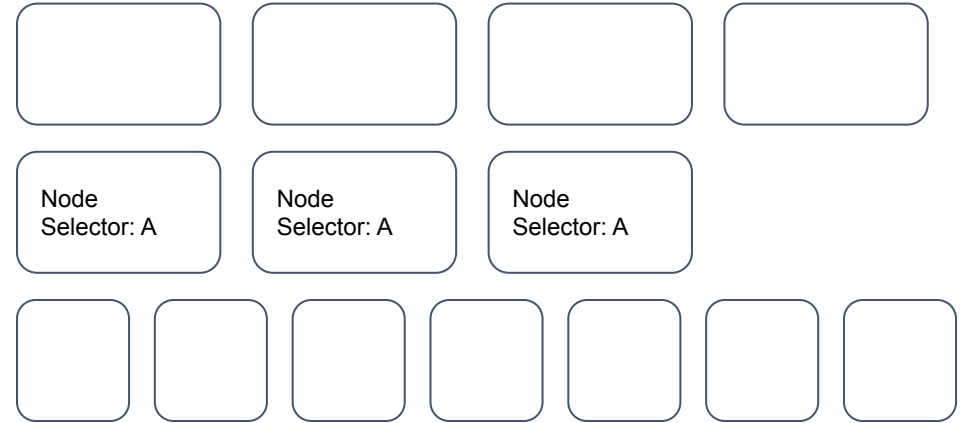


The story of a scale-up

Nodes:

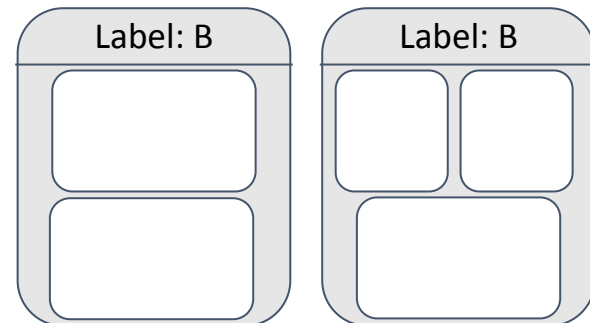
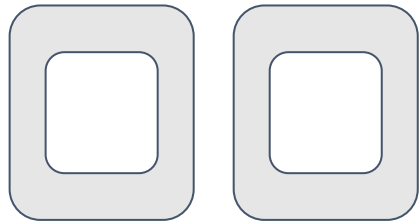
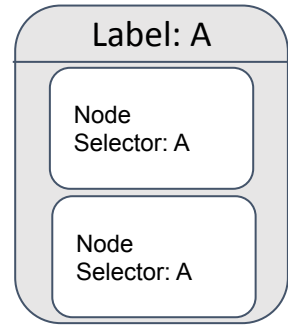


Pods:

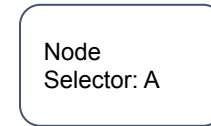


Scheduling

Nodes:

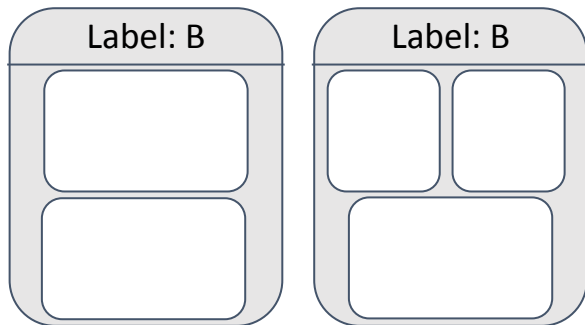
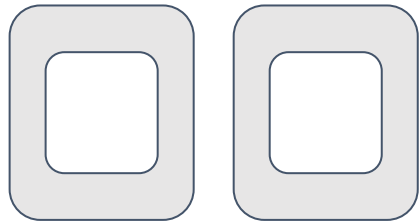
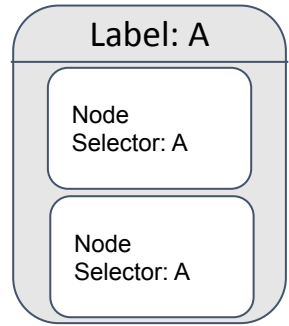


Pods:

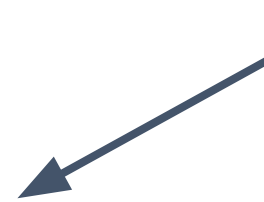
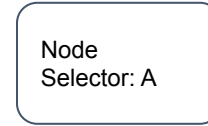


Scheduling

Nodes:



Pods:



Unschedulable!

Simulations



KubeCon

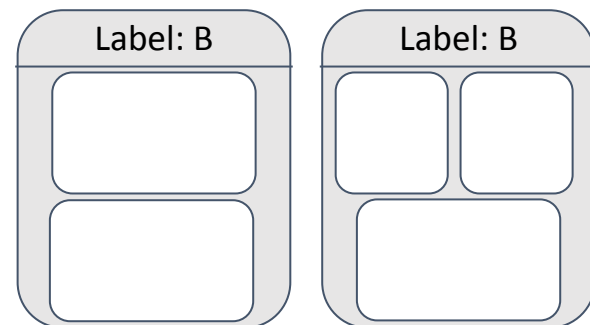
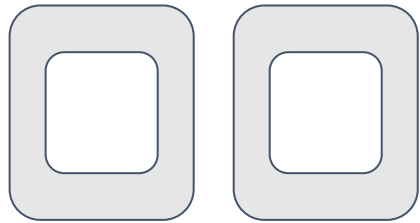
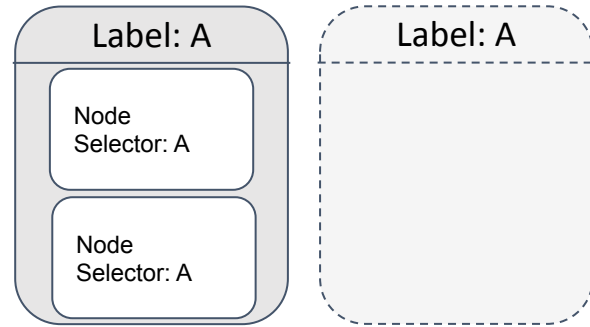


CloudNativeCon

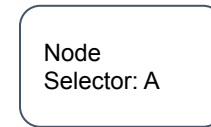
Europe 2020

Virtual

Nodes:

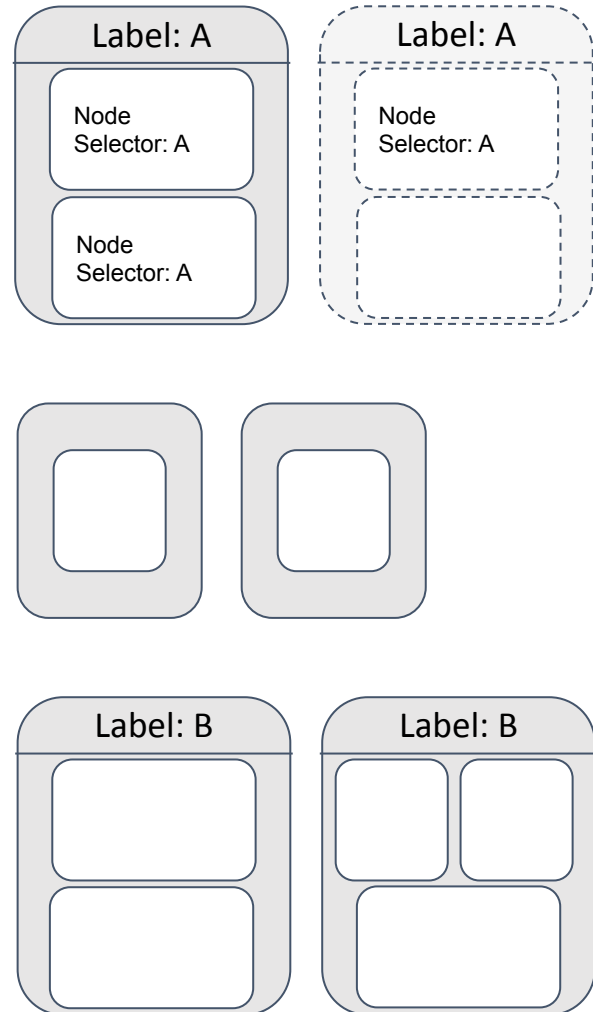


Pods:

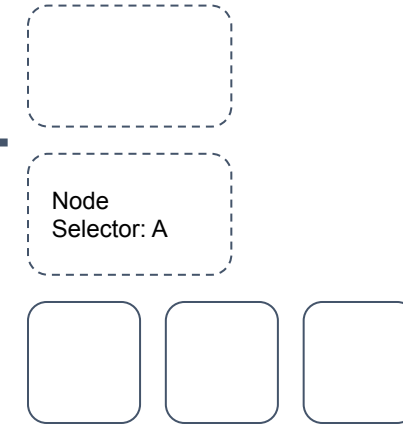


Simulations

Nodes:

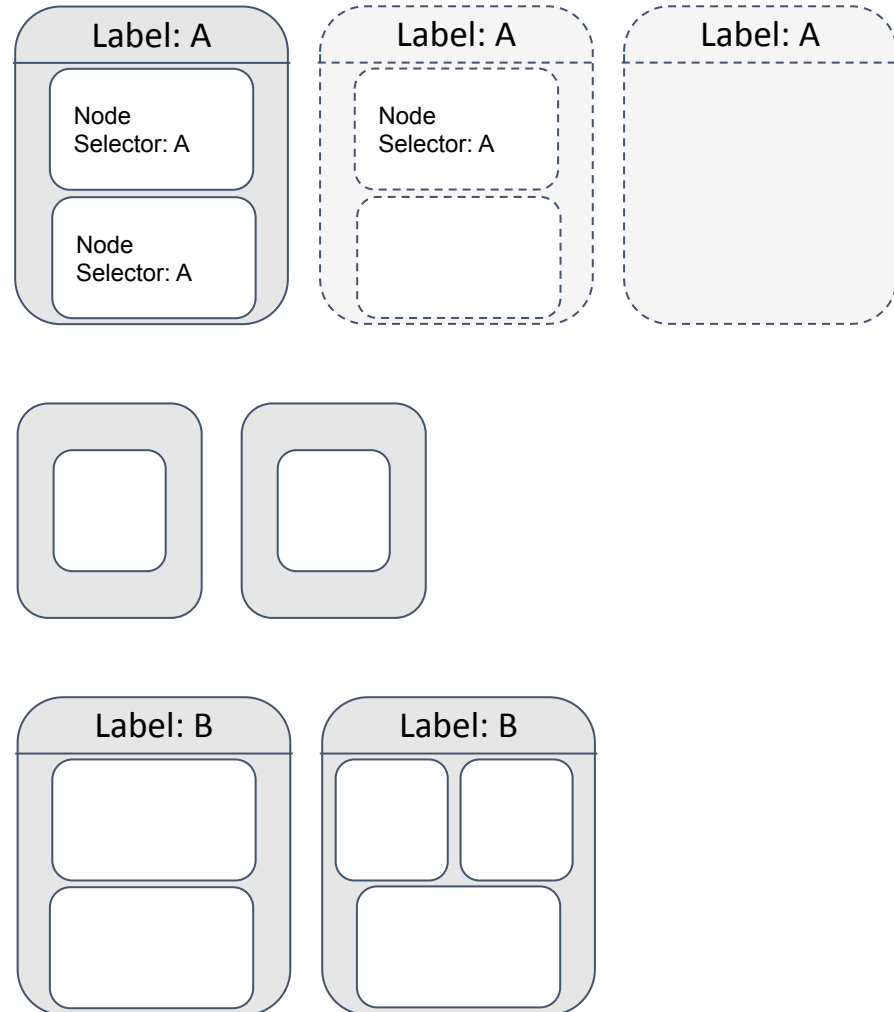


Pods:

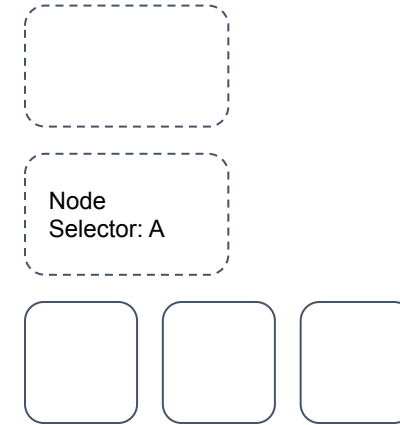


Simulations

Nodes:



Pods:



Simulations



KubeCon

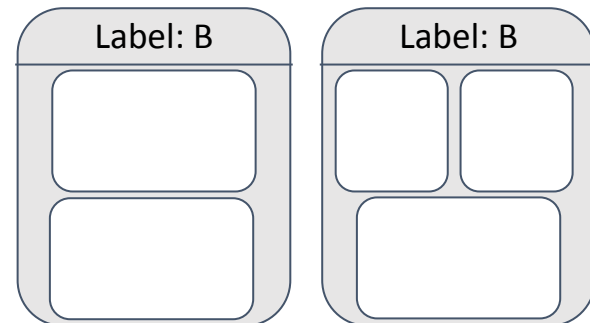
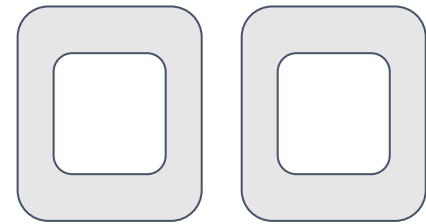
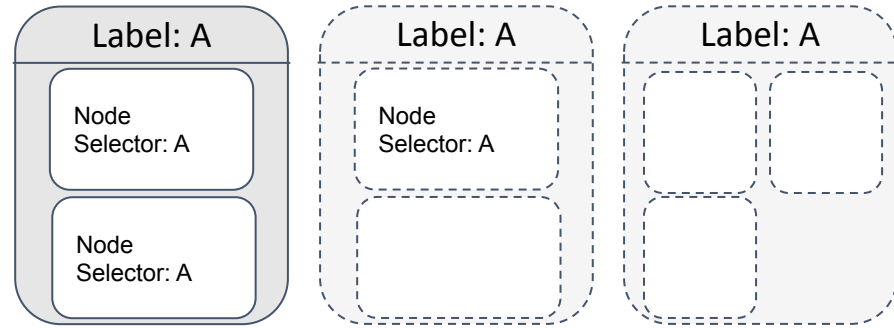


CloudNativeCon

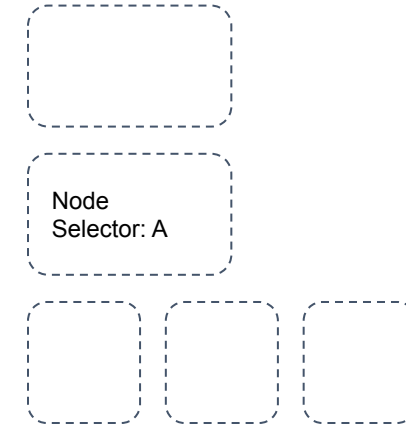
Europe 2020

Virtual

Nodes:

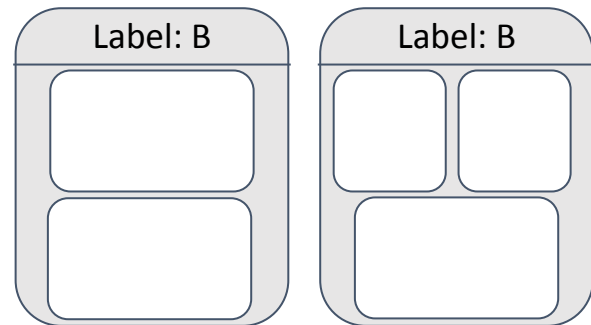
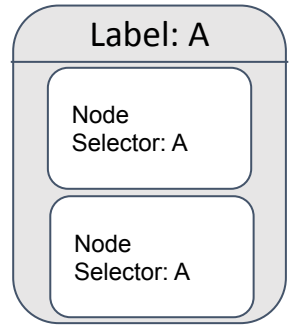


Pods:

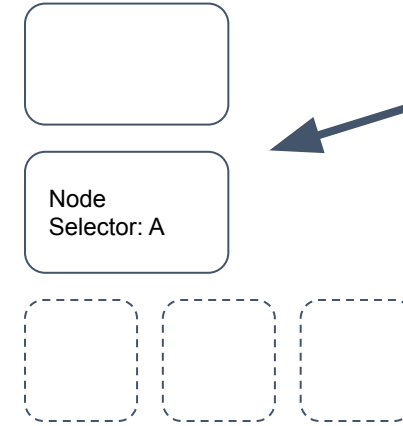


Simulations

Nodes:



Pods:

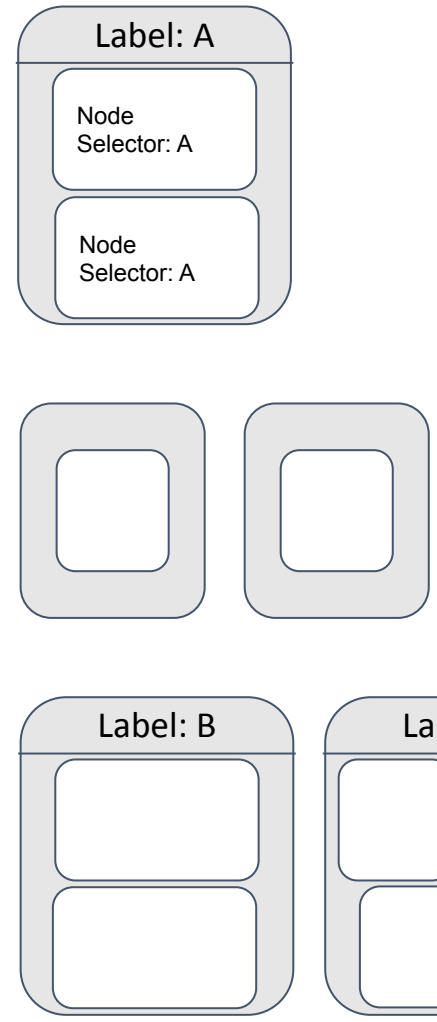


Those pods would remain unschedulable

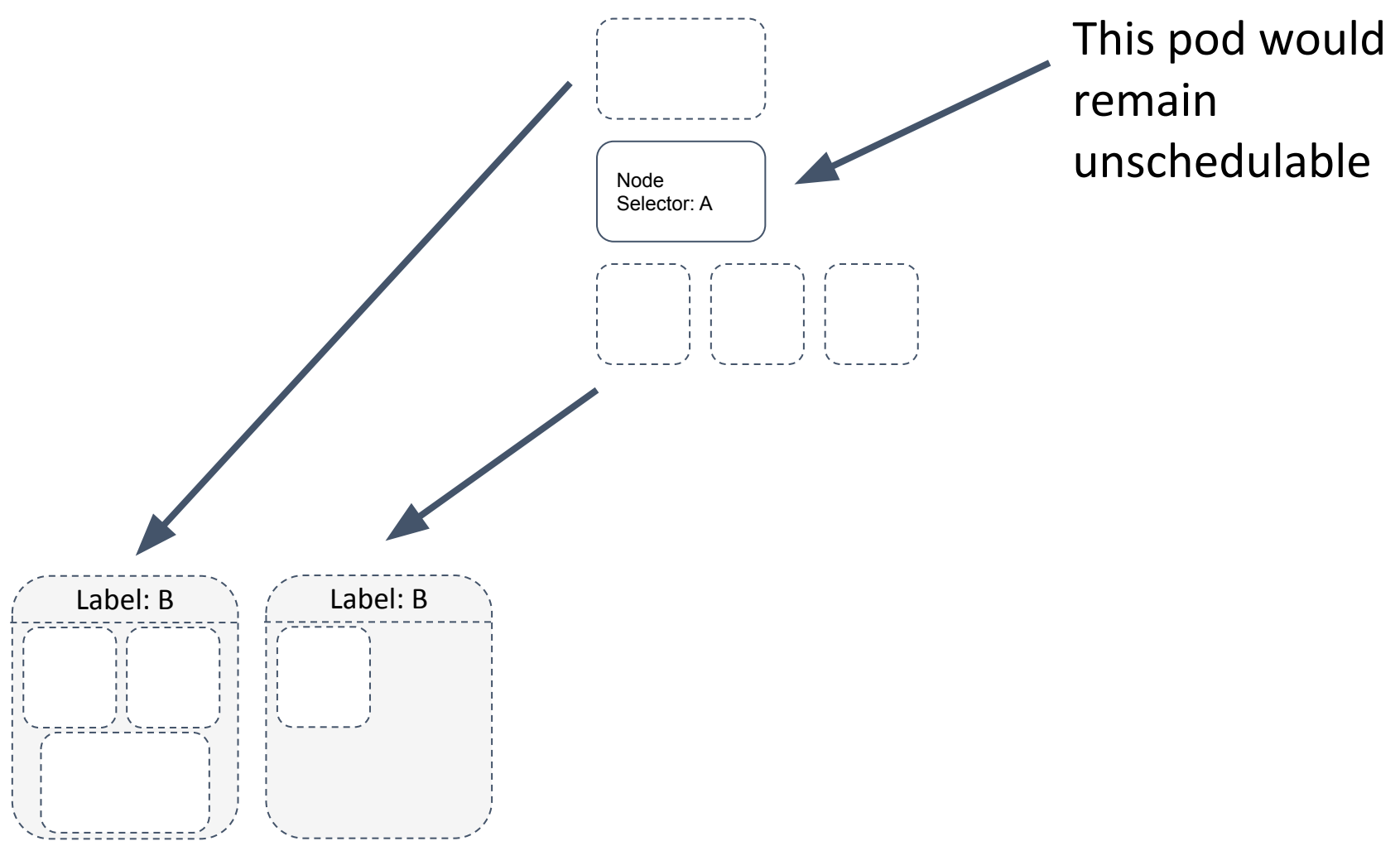


Simulations

Nodes:



Pods:



This pod would remain unschedulable

What now?



- CA doesn't consider mixed scale-ups.
- We have 3 options:
 - Add 2 nodes of first type to help all pods
 - Add 3 nodes of second type to help some pods
 - Add 2 nodes of third type to help some pods
- How to choose?
 - Different strategies ("expanders").
- What if some pods remain pending?
 - CA will try to scale-up again in next iteration.



- Meetings every Monday at 7:00 PST / 16:00 CET
 - <https://zoom.us/j/944410904>
- We have our own repo: <https://github.com/kubernetes/autoscaler>
- #sig-autoscaling at kubernetes.slack.com

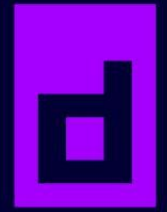


KubeCon



CloudNativeCon

Europe 2020



Virtual



KEEP CLOUD NATIVE

CONNECTED

