



KubeCon



CloudNativeCon

North America 2022

BUILDING FOR THE ROAD AHEAD

**DETROIT 2022**

# SIG-Scheduling Deep Dive

*Wei Huang, Apple*  
*Kensei Nakada, Mercari*

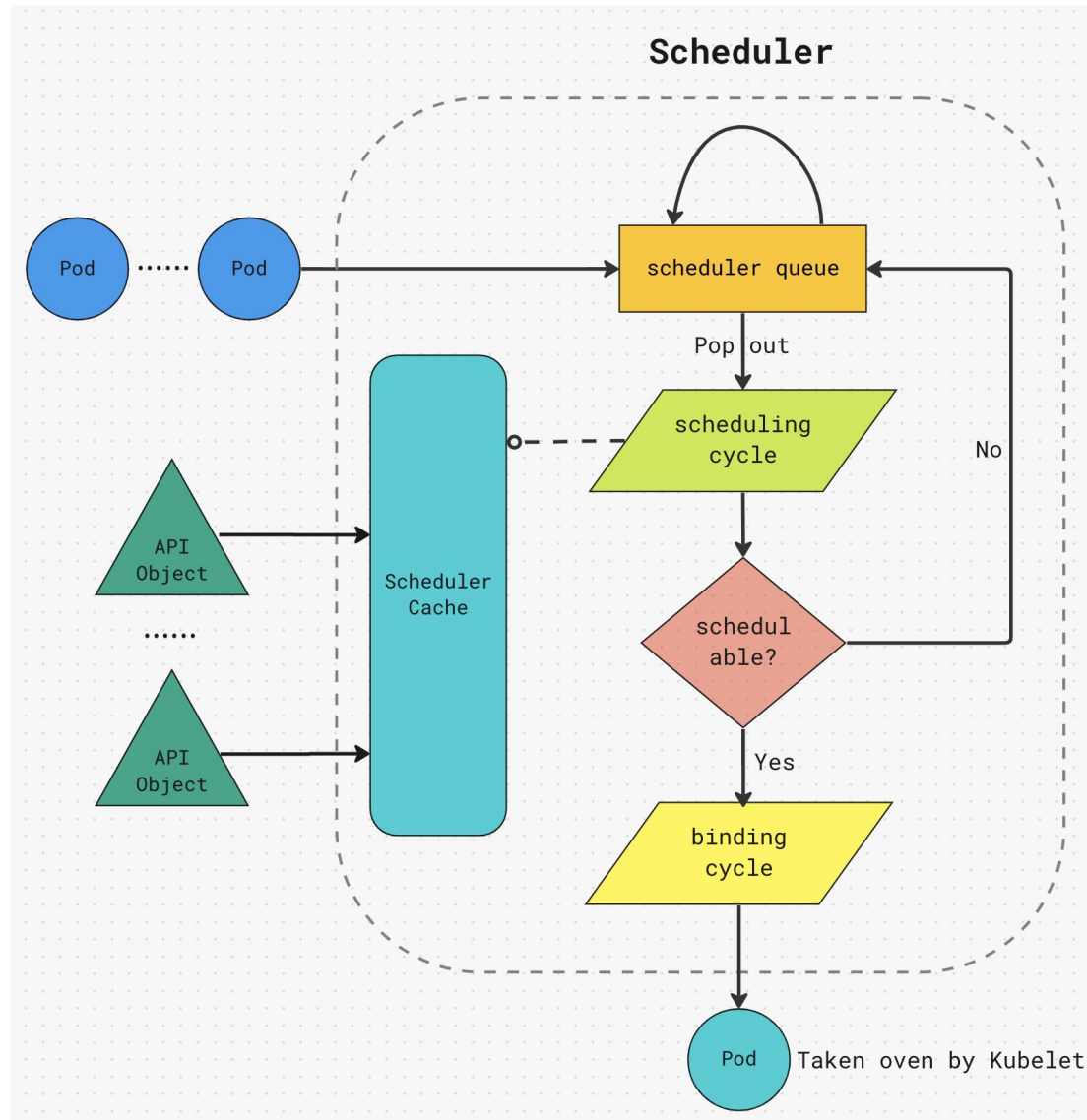
*Qingcan Wang, Alibaba*  
*Kante Yin, DaoCloud*

# Agenda

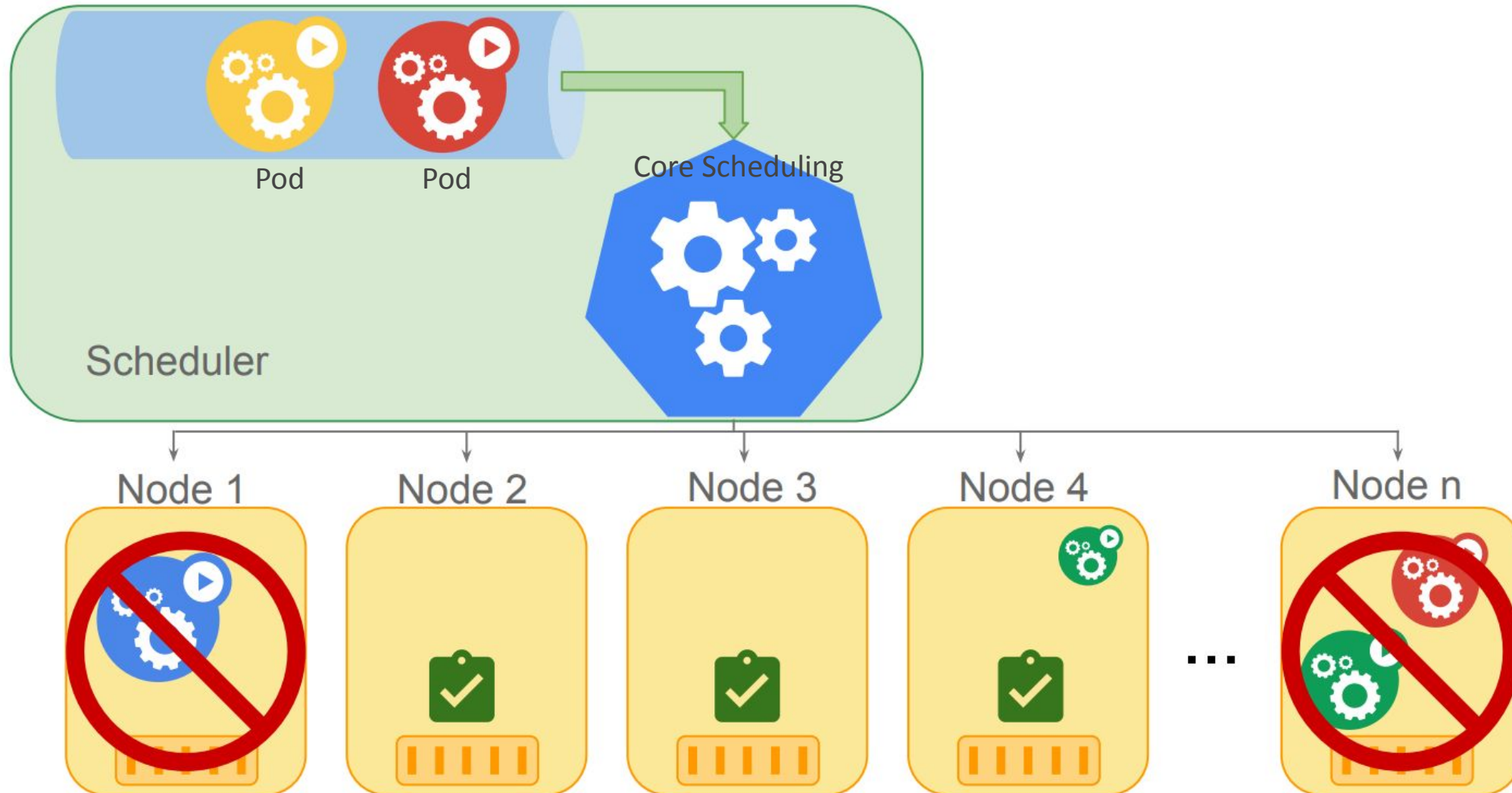
- Scheduler Overview
- Recent Developments
- Sub-project Updates
- Q & A

# Scheduler Overview

# Scheduler assigns Nodes To Pods

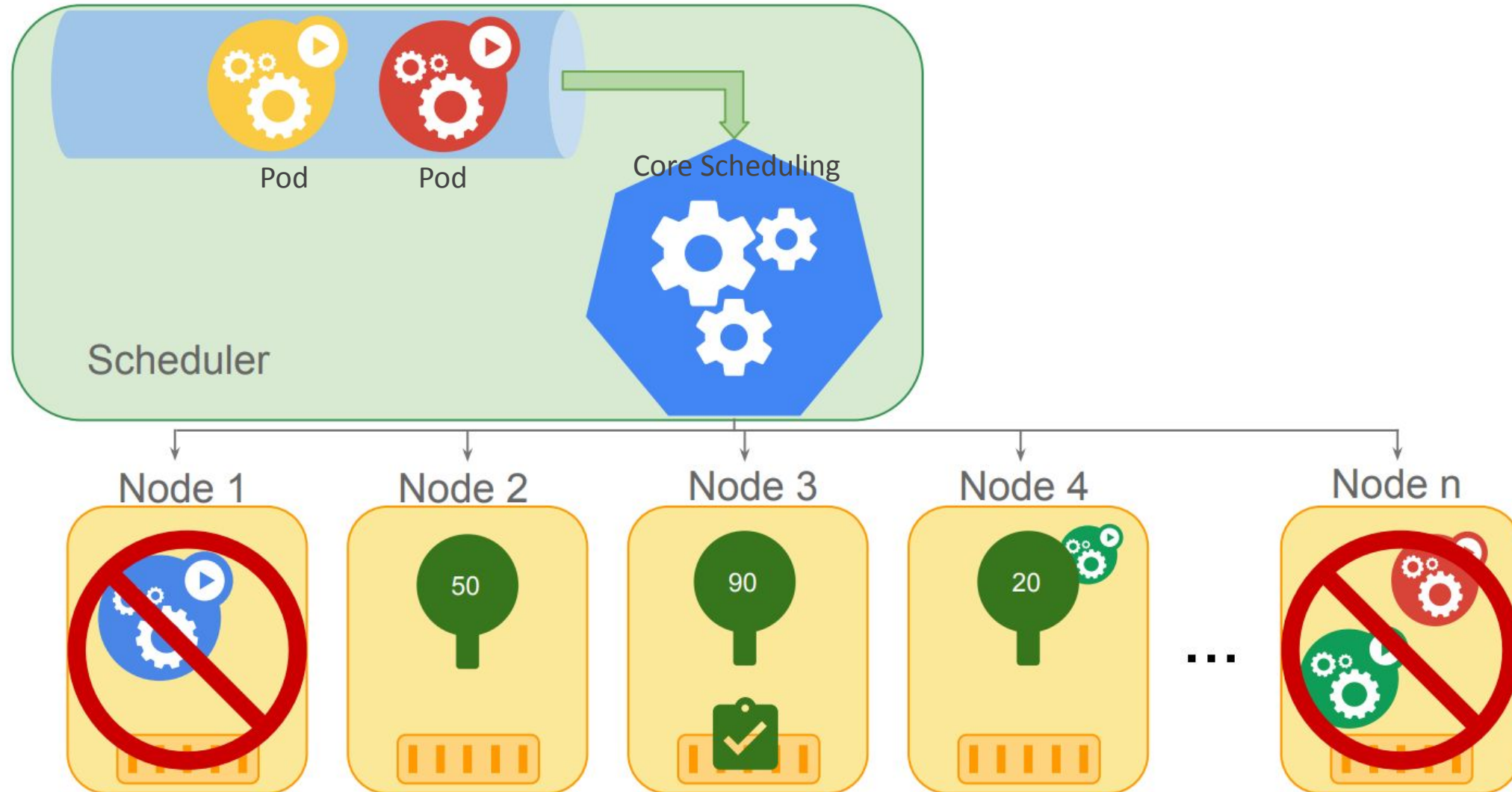


# Scheduler Cycle (Filter)

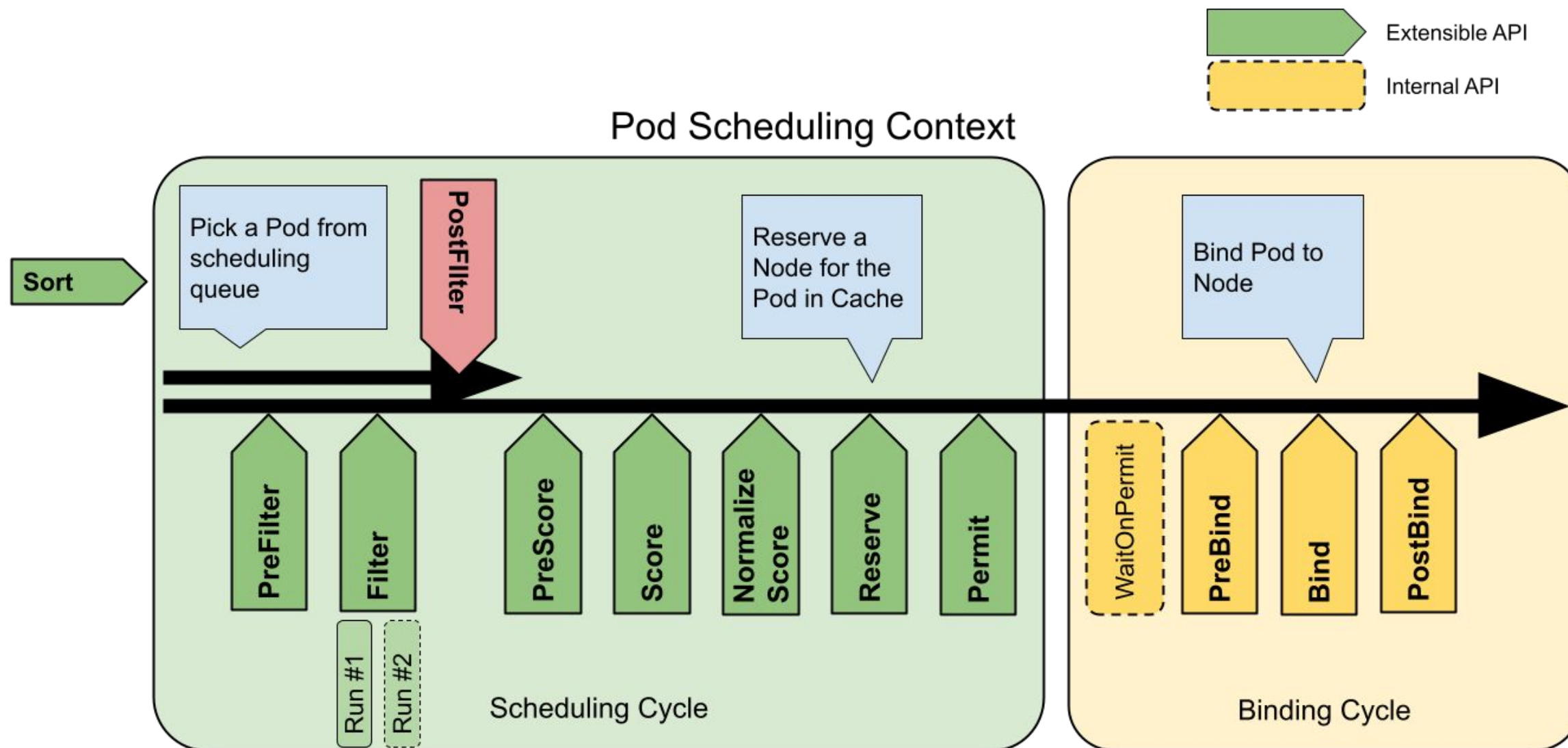




# Scheduling Cycle (Score)



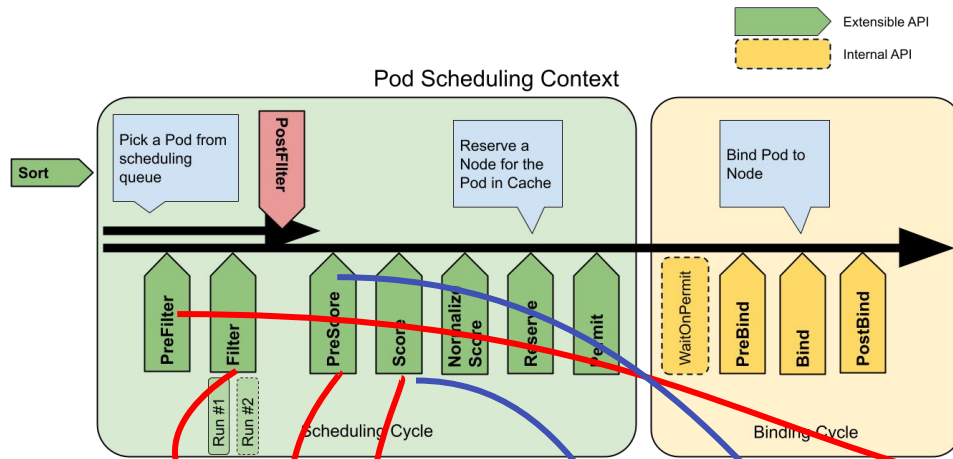
# Scheduling Framework





# Recent Developments

# KEP-2891: Simplified Scheduler Config



File: **regular-config.yaml**

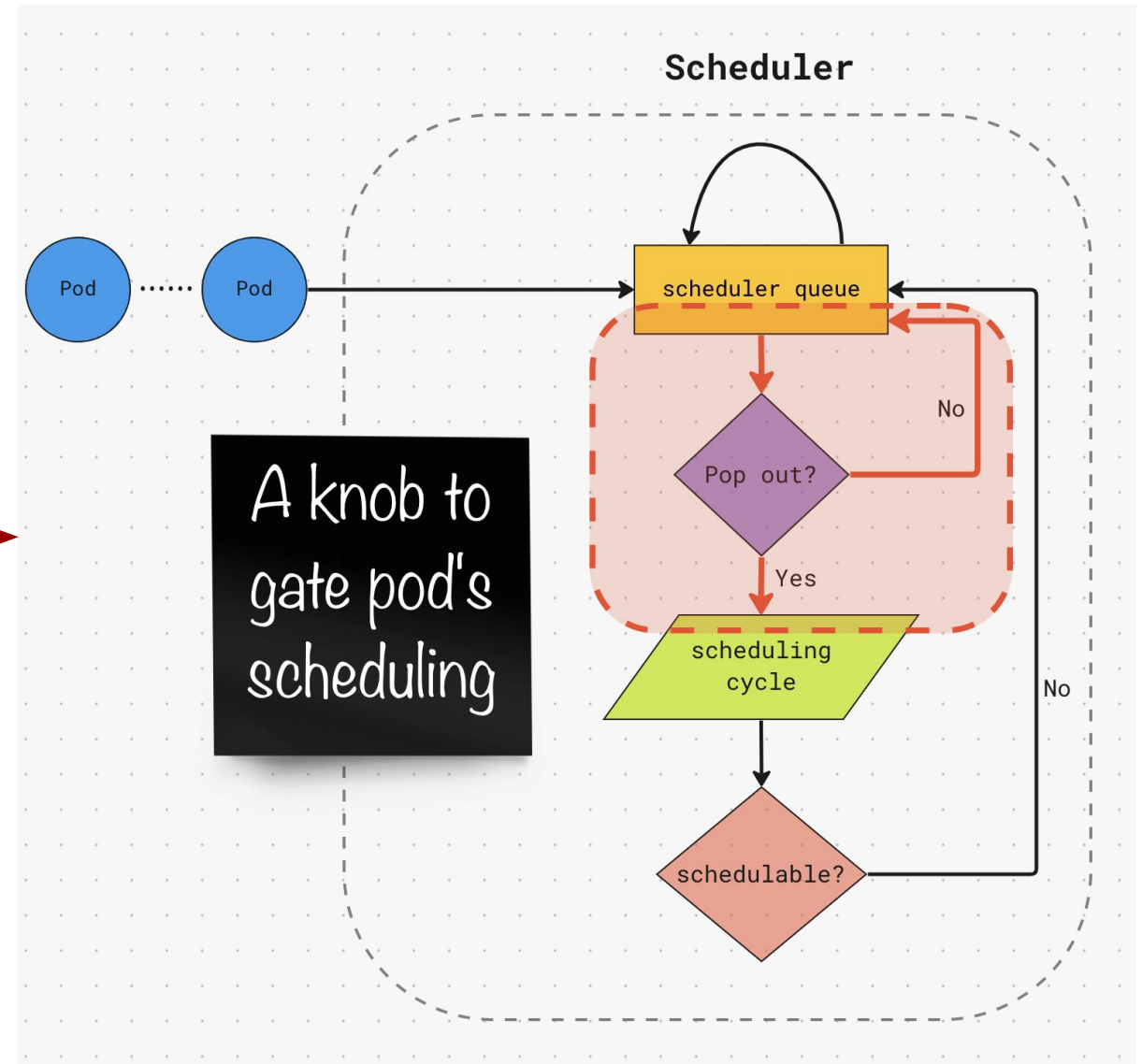
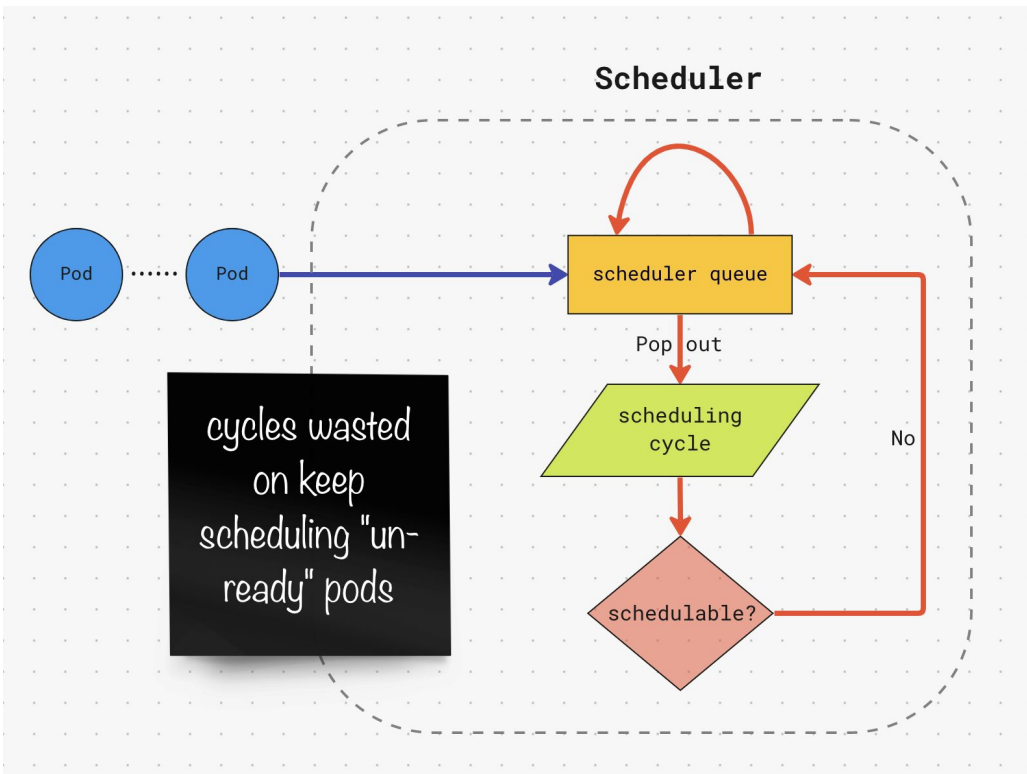
```
1  apiVersion: kubescheduler.config.k8s.io/v1beta3
2  kind: KubeSchedulerConfiguration
3  profiles:
4    - schedulerName: default-scheduler
5    plugins:
6      preFilter:
7        enabled:
8          - "foo"
9      filter:
10        enabled:
11          - "foo"
12      preScore:
13        enabled:
14          - "foo"
15          - "bar"
16      score:
17        enabled:
18          - "foo"
19          - "bar"
```

# KEP-2891: Simplified Scheduler Config (cont.)

```
File: multi-point-config.yaml
1  apiVersion: kubescheduler.config.k8s.io/v1beta3
2  kind: KubeSchedulerConfiguration
3  profiles:
4    - schedulerName: default-scheduler
5      plugins:
6        multiPoint:
7          enabled:
8            - name: "foo" # PreFilter / Filter / PreScore / Score
9            - name: "bar" # PreScore / Score
```

- Scheduler uses multiPoint to configure default plugins
- Co-used with traditional configuration-style
- Available in Kubernetes 1.23 (v1beta3 config) and 1.24+ (v1 config)

# (WIP) KEP-3521: Pod Scheduling Readiness

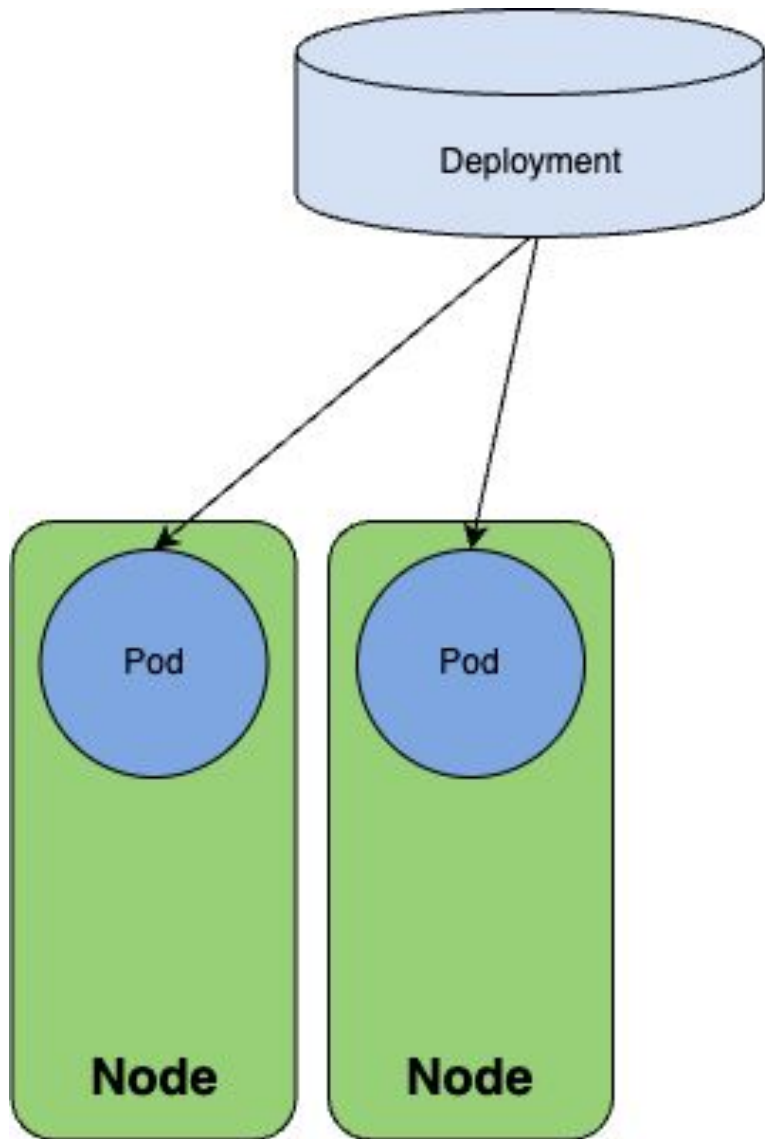


# KEP-3022: min domains in Pod Topology Spread

✨ Add `MinDomains` field to define the minimum number of topology domains

👤 By setting the minimum number of topology domains, you can expect the cluster autoscaler to add new topology domains

# KEP-3022: min domains in Pod Topology Spread

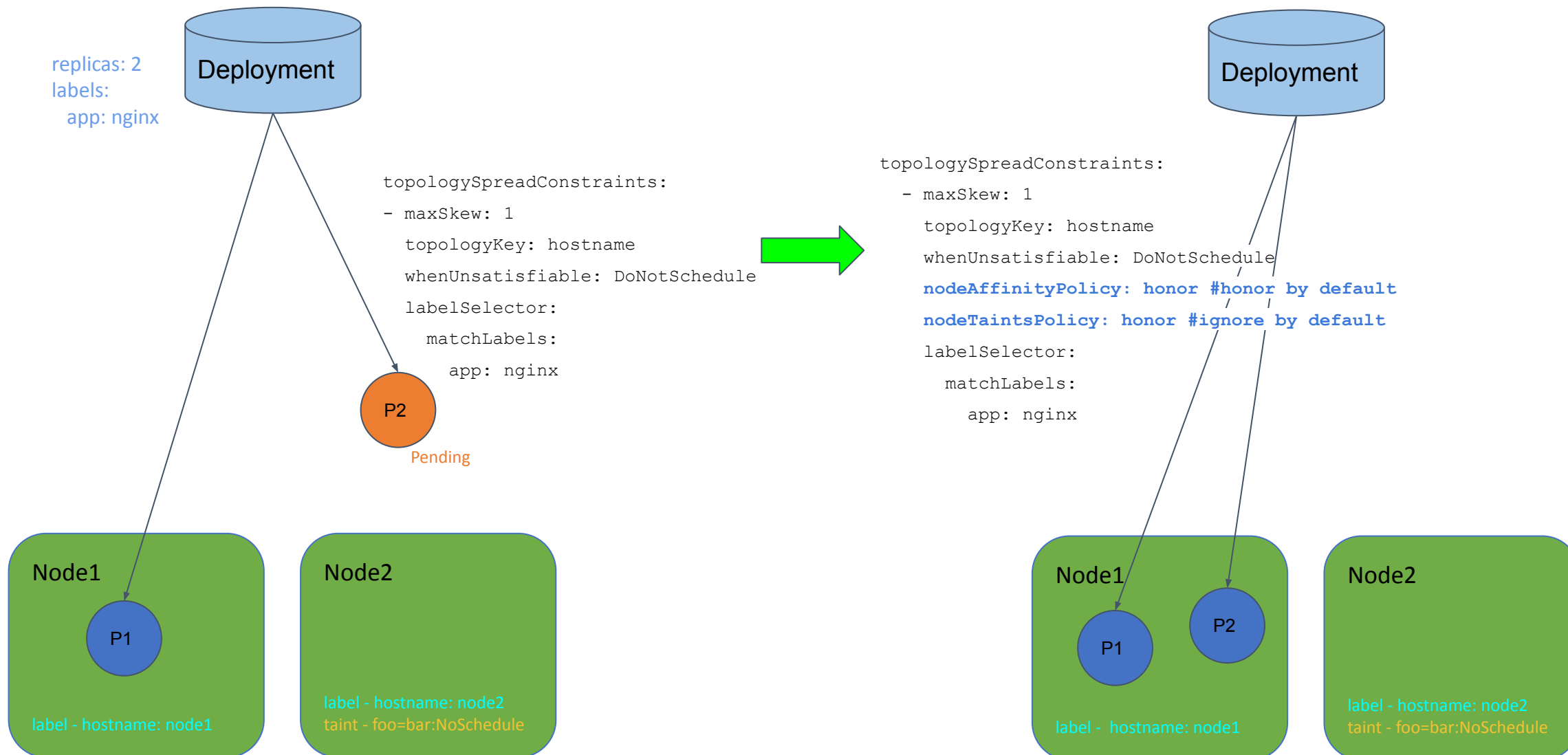


```
topologySpreadConstraints:  
  - maxSkew: 1  
    minDomains: 2  
    topologyKey: kubernetes.io/hostname  
    whenUnsatisfiable: DoNotSchedule  
    labelSelector:  
      matchLabels:  
        foo: bar
```

Cluster Autoscaler

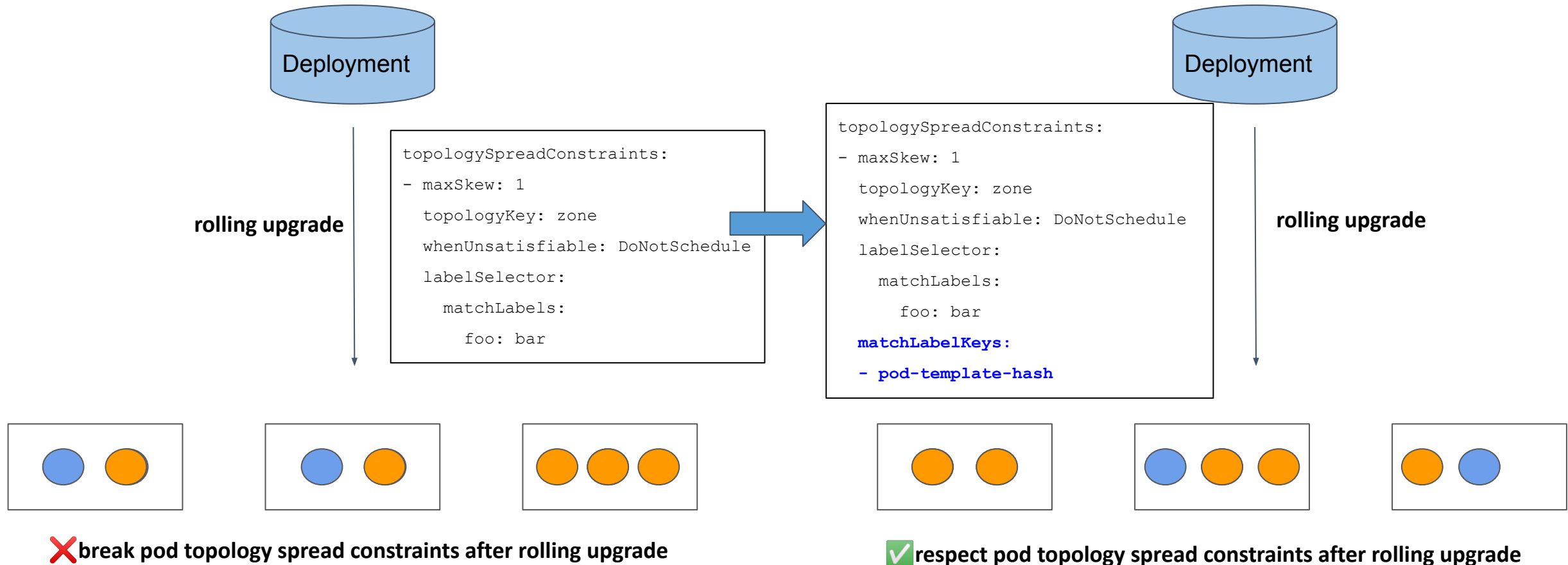


# KEP-3094: Take taints/tolerations into consideration when calculating PodTopologySpread skew



# KEP-3243: Respect PodTopologySpread after rolling upgrades

*PodTopologySpread* provides a new primitive **matchLabelKeys** to group Pods atop *labelSelector*. *Alpha version since v1.25*



# Worth noting features/fixes

- ✨ Performance improvement on DaemonSet Pod's scheduling latency [#108648](#)
- 🐛 Fix memory leak on kube-scheduler preemption [#111773](#)
- ⚠️ Flush internal unschedulablePods pool every 5m (from 60s) [#108761](#)
- ✨ Component Config in kube-scheduler is stable now [#110534](#)
- ⚠️ The legacy scheduler policy config is removed in v1.23 [#105424](#)

# Sub-project Updates

✨ **Simulate any Kubernetes scheduler without real clusters and see scheduling decisions in detail**

🚧 Development is ongoing for our first release! 🏃


👤 You can see each scheduler plugin's result in scheduling


- now only supports filter/score
- Other extension points and extenders will be also supported soon

💬 What's next?

- The scenario-based simulation and benchmark
- The simulator operator

# Kube-scheduler-simulator


 Kubernetes scheduler simulator




NEW STORAGECLASS

NEW NODE

Nodes

 node1

 pod1

Resource

☐ edit

APPLY

DELETE

Filter

Node	AzureDiskLimits	EBSLimits	GCEPDLimits	InterPodAffinity	NodeAffinity	NodeName	NodePorts	NodeResourcesFit	NodeTaints
node1	passed	passed	passed	passed	passed	passed	passed	passed	passed
node2	passed	passed	passed	passed	passed	passed	passed	passed	passed

Rows per page: 10 1-2 of 2 < >

Score

Node	ImageLocality	InterPodAffinity	NodeAffinity	NodeResourcesBalancedAllocation	NodeResourcesFit	PodTopologySpread	TaintToleration
node1	0	0	0	76	73	0	0
node2	0	0	0	76	73	0	0

Rows per page: 10 1-2 of 2 < >

Final Score (Normalized + Applied plugin weight)

Node	ImageLocality	InterPodAffinity	NodeAffinity	NodeResourcesBalancedAllocation	NodeResourcesFit	PodTopologySpread	TaintToleration
node1	0	0	0	76	73	0	0
node2	0	0	0	76	73	0	0

Rows per page: 10 1-2 of 2 < >

Resource Definition

▼ metadata

name: pod1

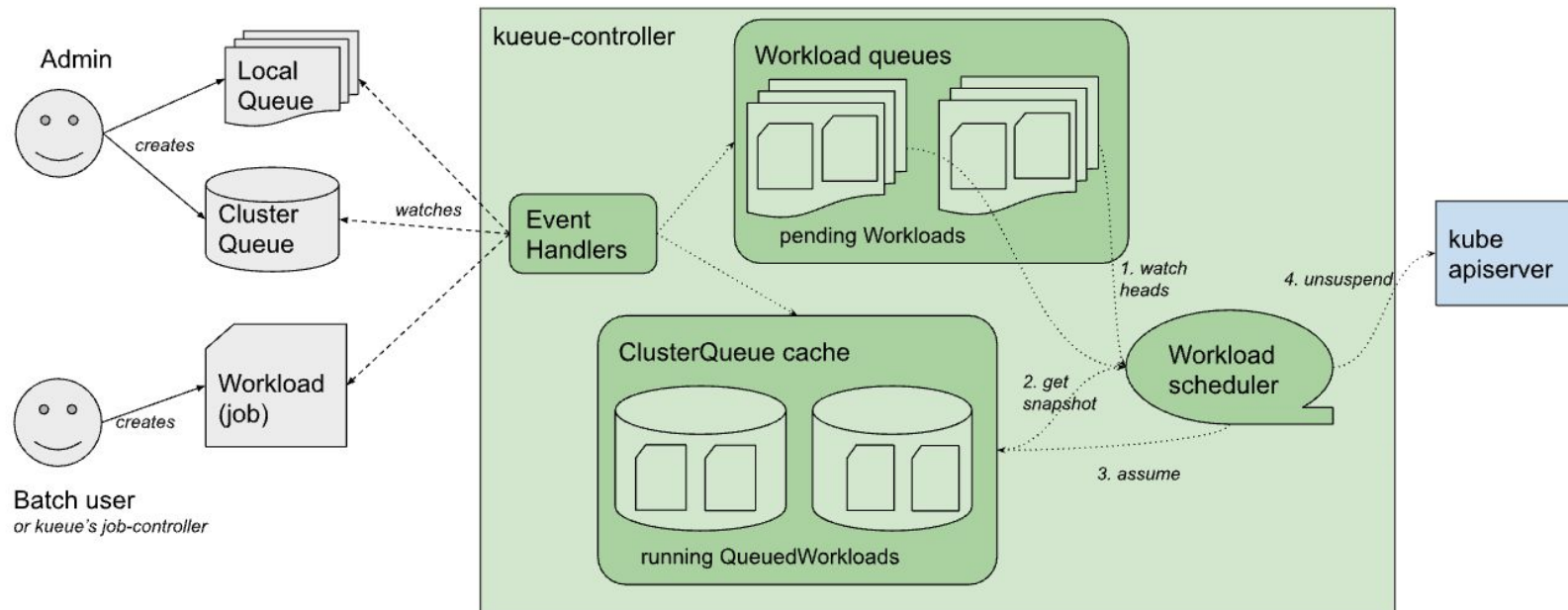
namespace: default



## Kueue: Kubernetes-native Job Queueing

Manage access to a limited pool of resources shared by multiple tenants, including these core features:

- Quota management
- Fair sharing of resources between tenants
- Flexible placement of jobs across different resource types based on availability
- Different queueing strategies: StrictFIFO or BestEffortFIFO



## APIs <https://github.com/kubernetes-sigs/kueue/tree/main/docs/concepts>

- [ResourceFlavor](#): A kind or type of resource in a cluster. It could distinguish among different characteristics of resources such as availability, pricing, architecture, models, etc.
- [ClusterQueue](#): A cluster-scoped resource that governs a pool of resources, defining usage limits and fair sharing rules
- [LocalQueue](#): A namespaced resource that groups closely related workloads belonging to a single tenant
- [Workload](#): An application that will run to completion. It is the unit of *admission* in Kueue. Sometimes referred to as *job*

## Ongoing Work

- Add enhancement for workload preemption <https://github.com/kubernetes-sigs/kueue/pull/410>
- Dynamically reclaiming resources <https://github.com/kubernetes-sigs/kueue/pull/331>
- Add flavorAssignmentStrategy to localQueue <https://github.com/kubernetes-sigs/kueue/pull/376>
- Integration with cluster autoscaler
- ...

## ✨ A Post-Scheduling Eviction Component

### **NEW** A Big Step Forward – Descheduler Framework([#753](#))

- What's the motivations?
  - Growing needs for building customized descheduler(similar to kube-scheduler)
  - Increasing new strategies raises the maintenance complexity
  - More flexible and extensible
- Where are we?
  - All strategies migrated to plugins with Descheduler && Balance extension point
  - Evictor Filter && preEvictionFilter as extension points
- What is next?
  - v1alpha2 configuration + conversion
  - Arguments defaulting
  - Sort && preEvictionSort extension points

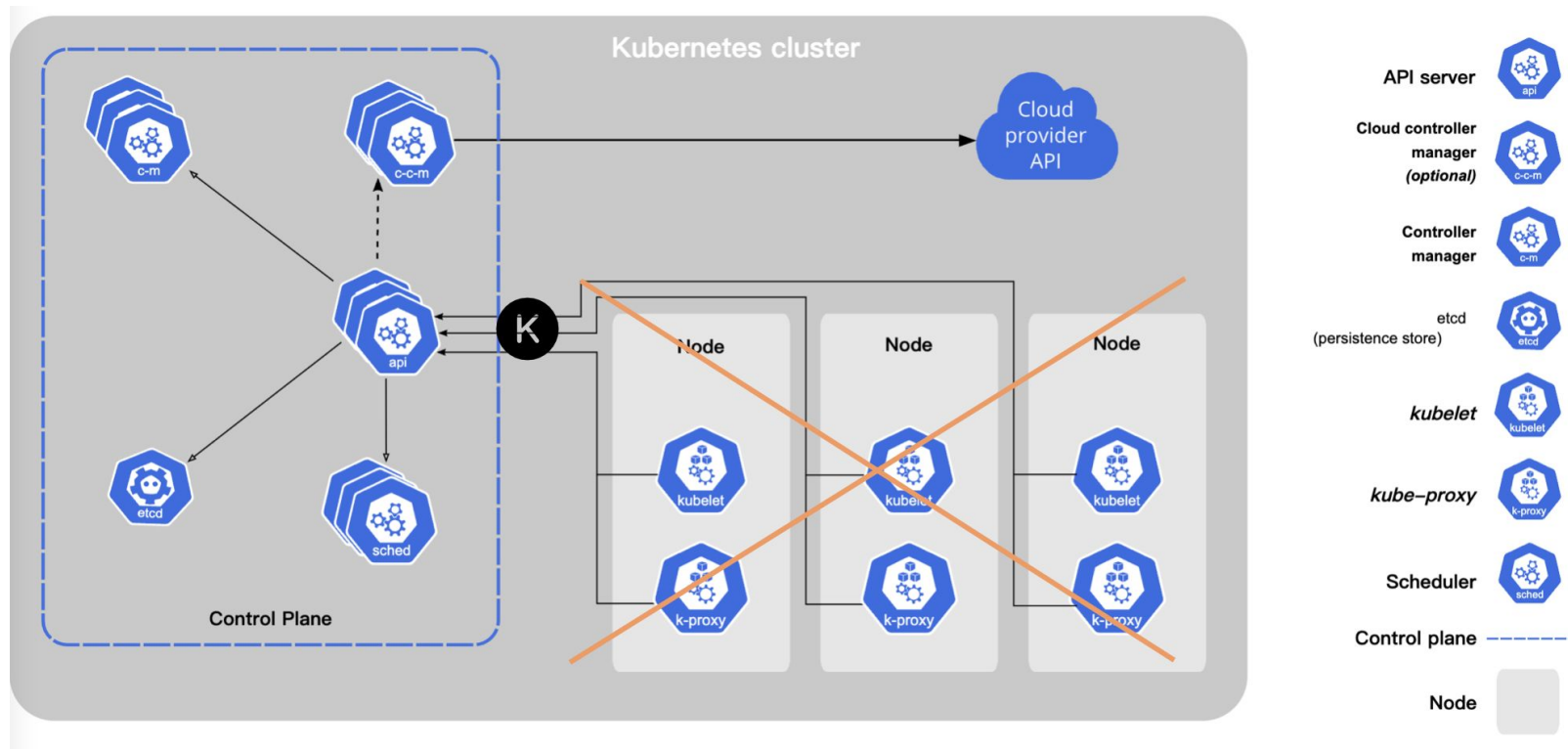
## ★ Other Improvements && Bug Fixes

- Leader election for HA Deployment [#722](#)
- New policy MaxNoOfPodsToEvictPerNamespace added [#658](#)
- A bunch of improvements in installation with helm chart
- ... (mainly focus on the descheduler framework for all codes are moving around)

## 🎉 Project management update:

👉 [Descheduler Community Meeting](#)

👉 [Descheduler Release Dashboard](#)

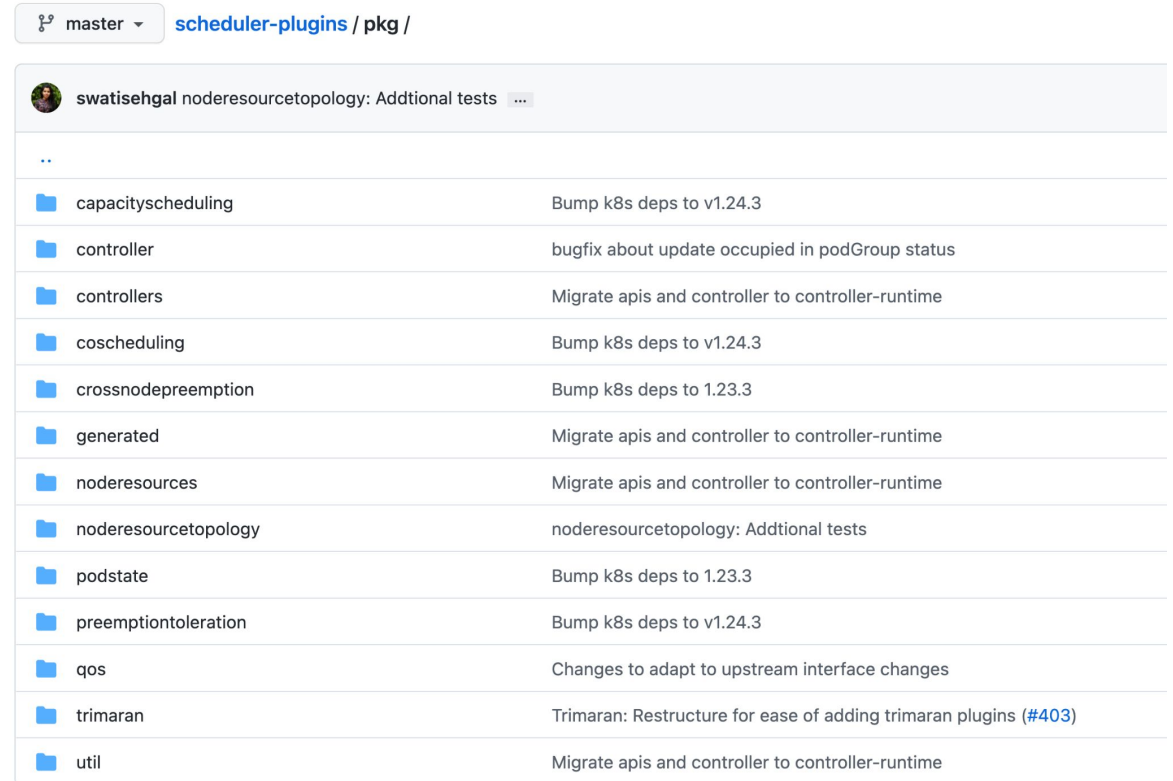


- Functionality tests
- Scalability tests
- CI pipeline

- Integration (ClusterAutoscaler, ClusterAPI, Karmada, ...)
- Scheduler Simulator

# Scheduler-plugins

- A repository hosting out-of-tree scheduler plugins
- Exercising innovative ideas
  - co-scheduling
  - elastic quota
  - topology aware scheduling
  - network cost aware scheduling
  - load aware scheduling
  - .....
- Patterns to develop a scheduler plugin



The screenshot shows the GitHub repository for `scheduler-plugins` at the `master` branch. The repository is owned by `swatisehgal`. The table below lists the subdirectories and their associated changes or descriptions.

Directory	Description
<code>capacityscheduling</code>	Bump k8s deps to v1.24.3
<code>controller</code>	bugfix about update occupied in podGroup status
<code>controllers</code>	Migrate apis and controller to controller-runtime
<code>coscheduling</code>	Bump k8s deps to v1.24.3
<code>crossnodepreemption</code>	Bump k8s deps to 1.23.3
<code>generated</code>	Migrate apis and controller to controller-runtime
<code>noderesources</code>	Migrate apis and controller to controller-runtime
<code>noderesourcetopology</code>	noderesourcetopology: Additional tests
<code>podstate</code>	Bump k8s deps to 1.23.3
<code>preemptiontoleration</code>	Bump k8s deps to v1.24.3
<code>qos</code>	Changes to adapt to upstream interface changes
<code>trimaran</code>	Trimaran: Restructure for ease of adding trimaran plugins (#403)
<code>util</code>	Migrate apis and controller to controller-runtime



# Q & A



Please scan the QR Code above to  
leave feedback on this session