
KubeCon
CloudNativeCon
Europe 2024

SIG-Scheduling Intro & Deep Dive

Wei Huang
Apple

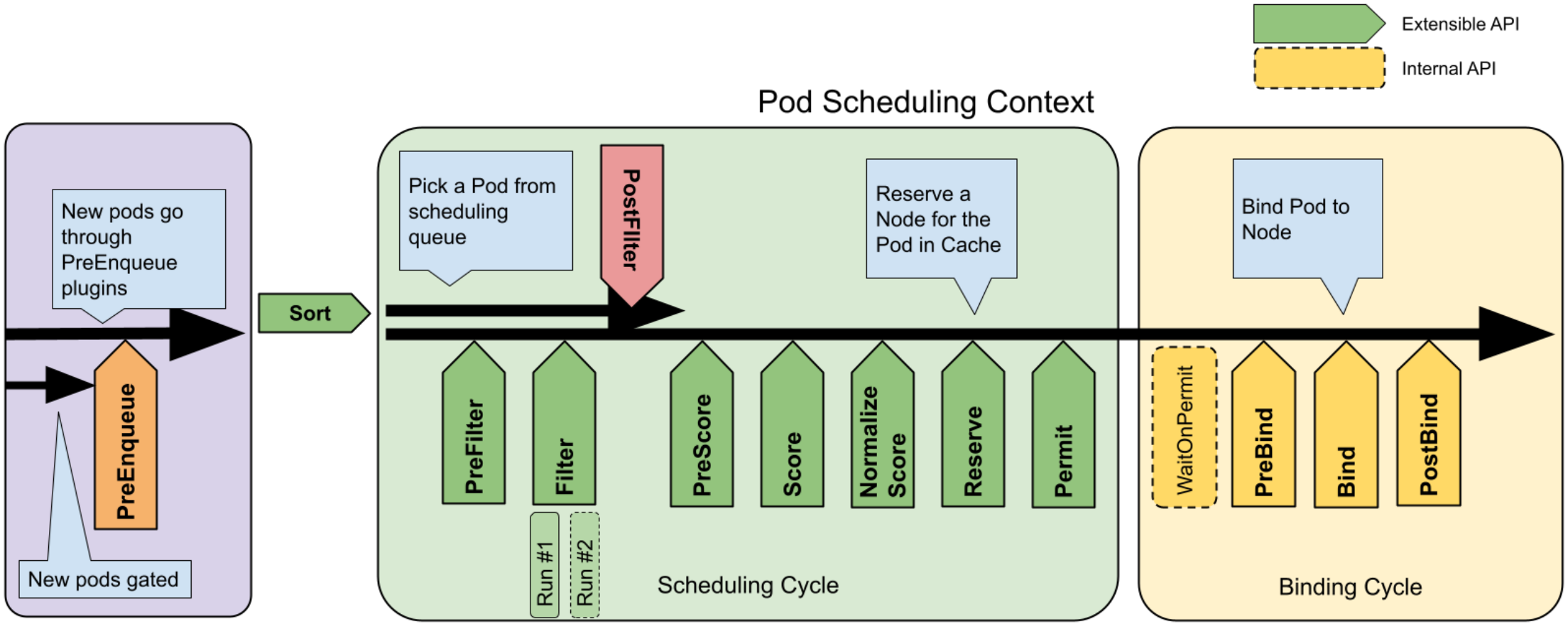
Kante Yin
DaoCloud

Agenda

- **Scheduler Overview**
- **Recent Updates**
  - Major Themes in v1.29 & v1.30
  - Other notable updates
- **Sub-project Updates**
  - KWOK
  - Kueue
  - Kube-scheduler-wasm-extension
  - Scheduler Plugins
  - Descheduler
- **Join us**
- **Q & A**

Scheduler Overview

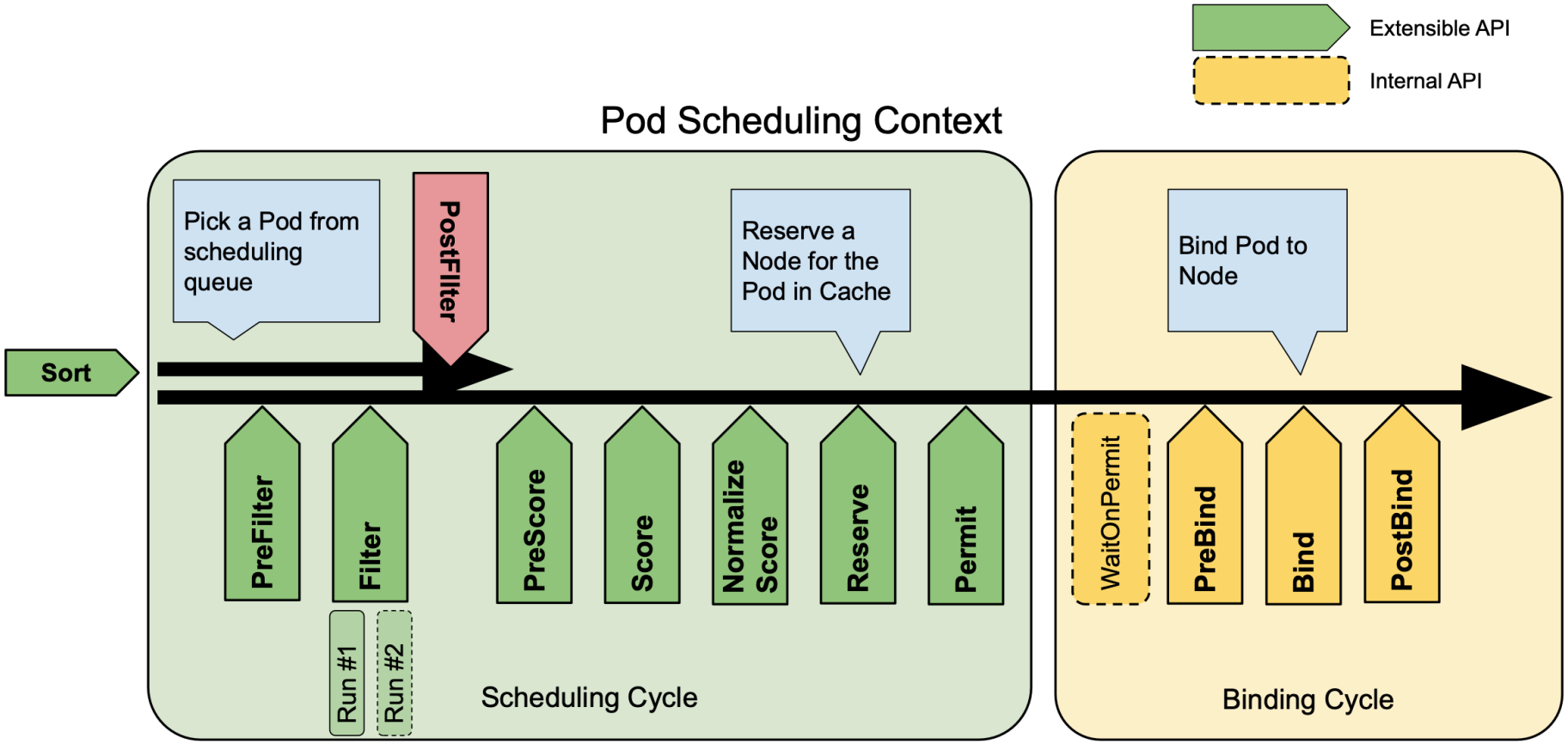
Schedule Framework



Recent Updates

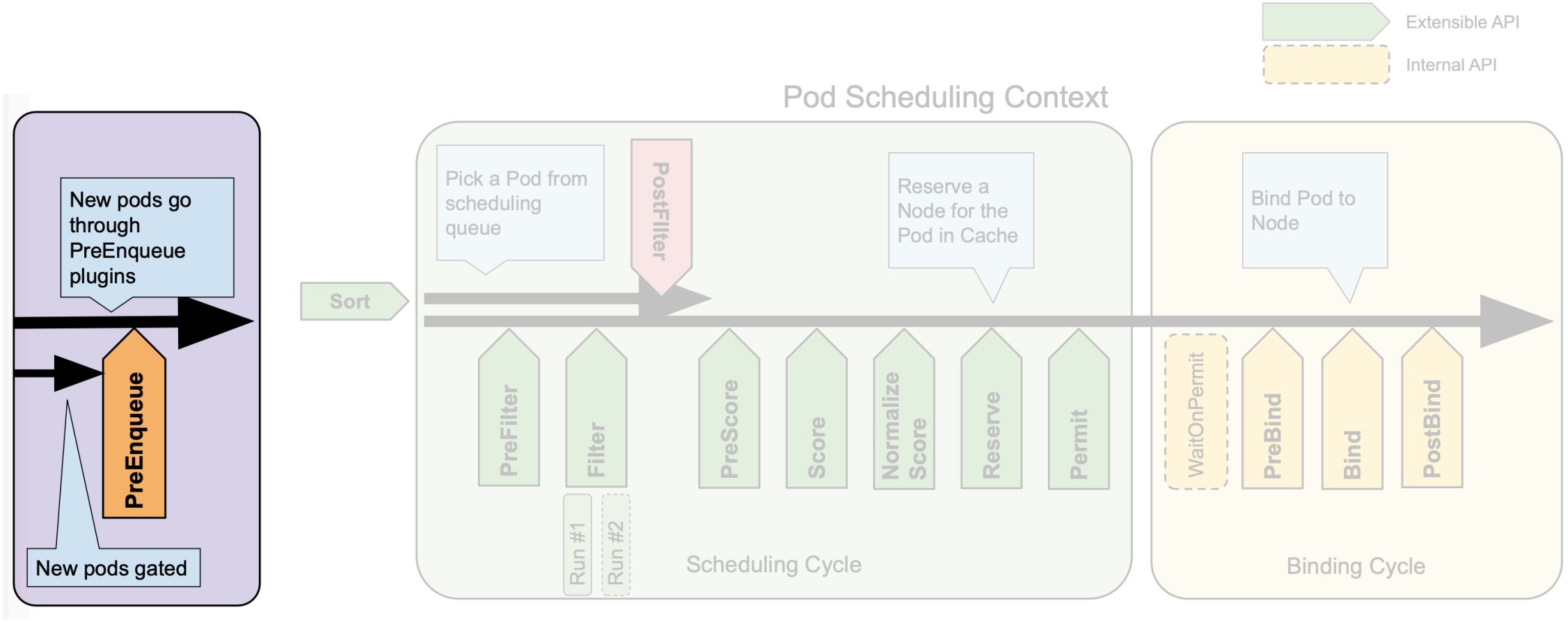
KEP-3521: Pod Scheduling Readiness (GA)

For a long time...



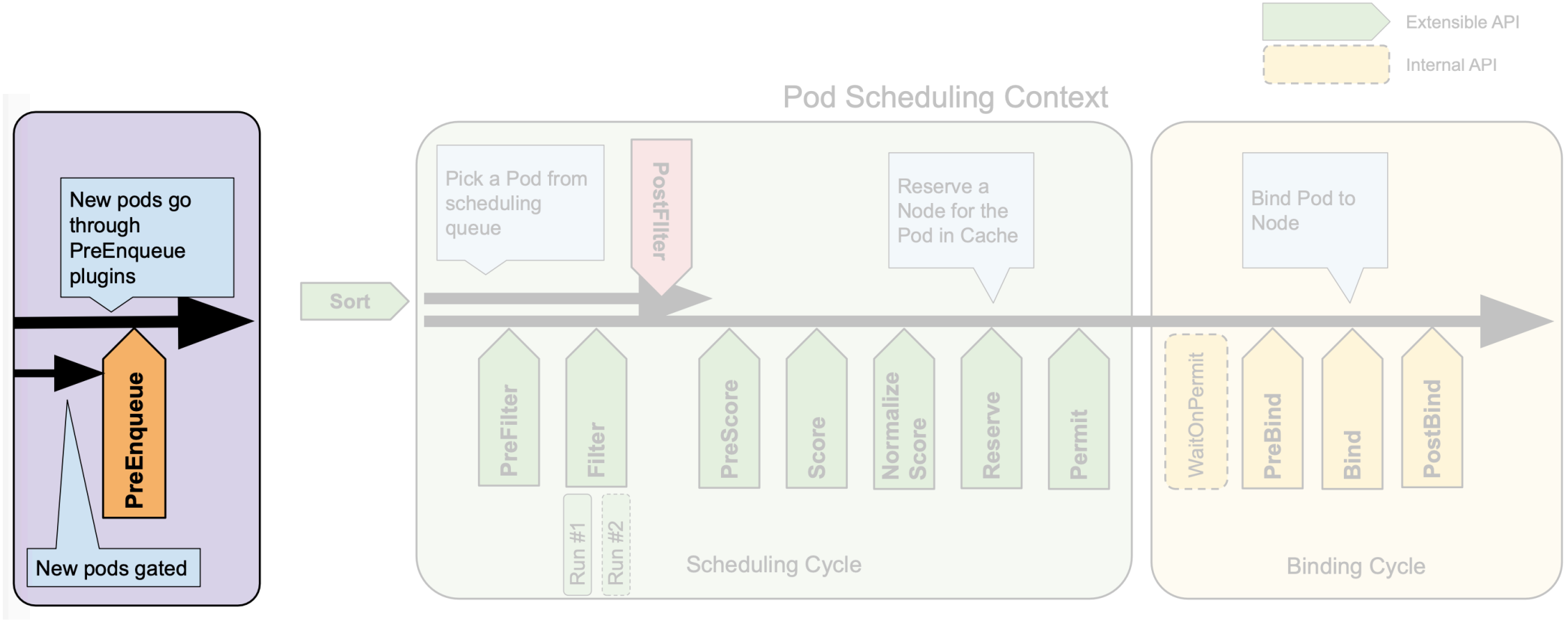
KEP-3521: Pod Scheduling Readiness (GA)

Until Kubernetes 1.26 (Alpha) ...



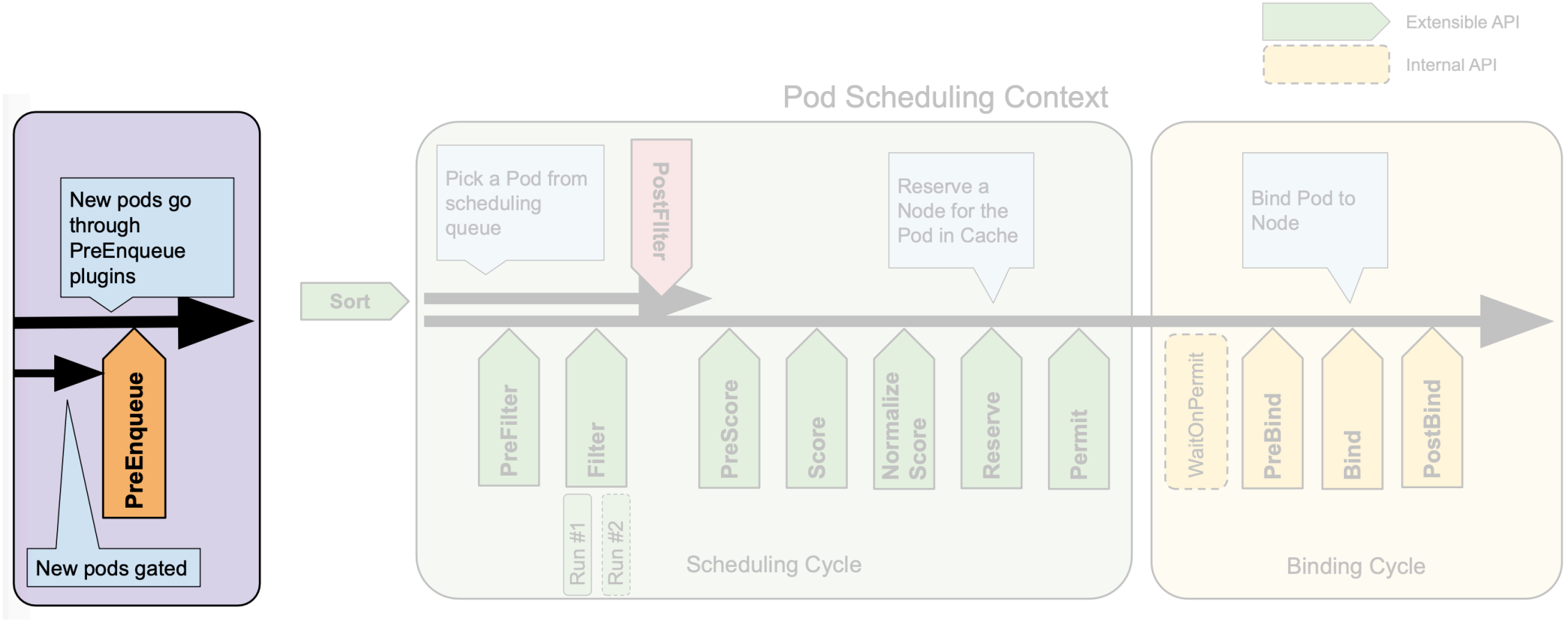
KEP-3521: Pod Scheduling Readiness (GA)

Until Kubernetes 1.27 (Beta) ...



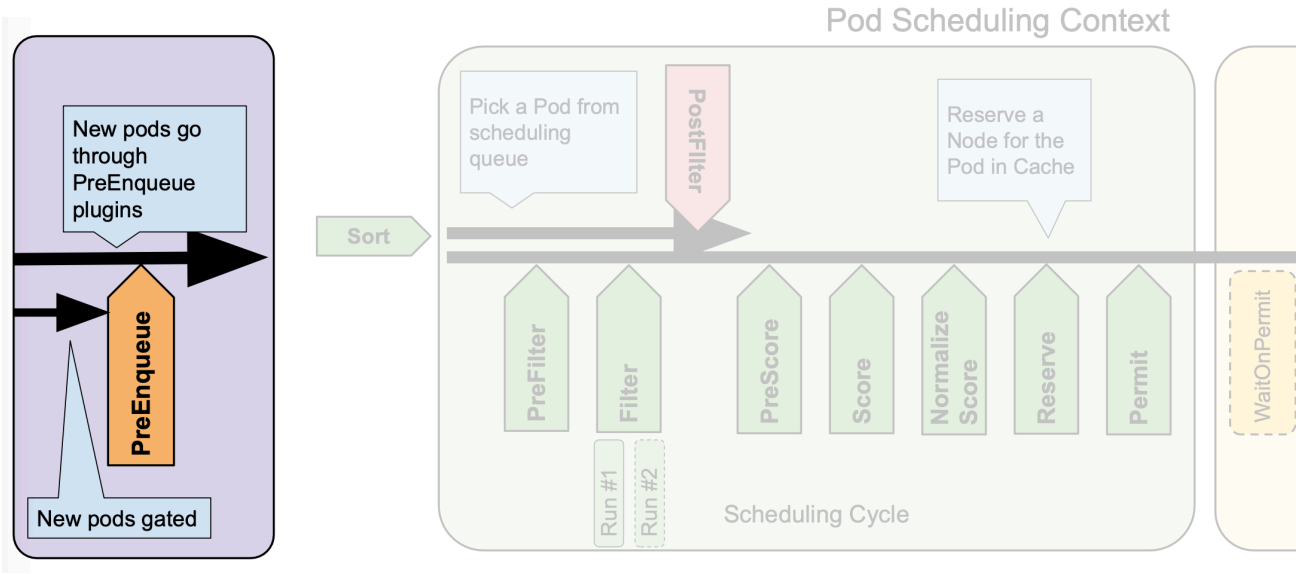
KEP-3521: Pod Scheduling Readiness (GA)

Until Kubernetes 1.30 (GA) ...



KEP-3521: Pod Scheduling Readiness (GA)

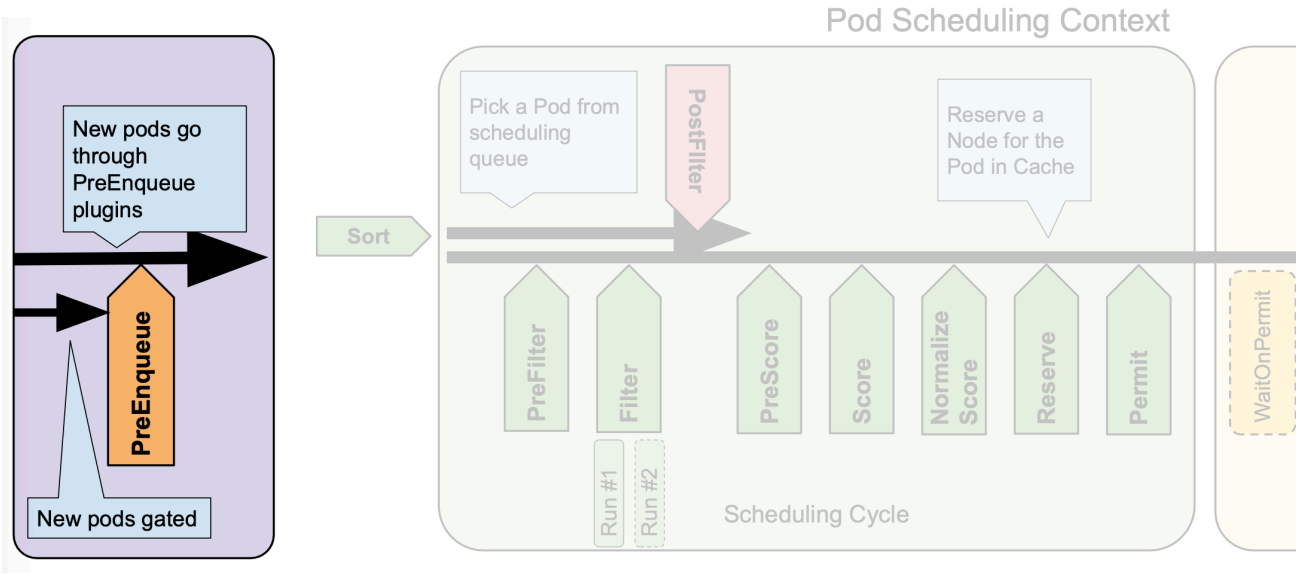

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: example-pod
spec:
  schedulingGates:
  - name: example.com/foo
  - name: example.com/bar
```




```
NAME       READY   STATUS
test-pod   0/1     SchedulingGated
```

KEP-3521: Pod Scheduling Readiness (GA)

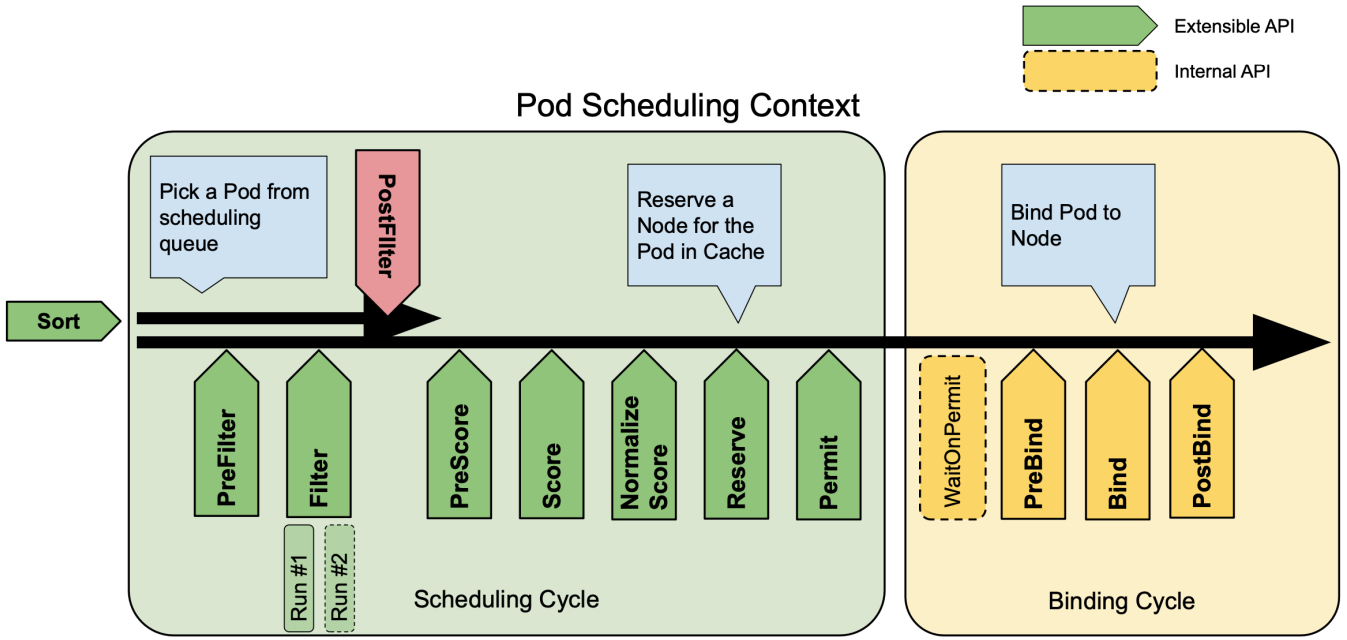

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: example-pod
spec:
  schedulingGates:
  - name: example.com/foo
  - name: example.com/bar
```




```
NAME       READY   STATUS
test-pod   0/1     SchedulingGated
```

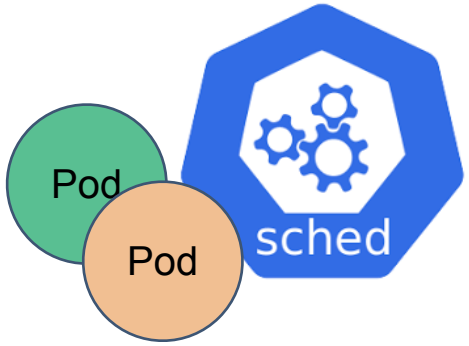
KEP-3521: Pod Scheduling Readiness (GA)


```yaml
apiVersion: v1
kind: Pod
metadata:
  name: example-pod
spec:
  schedulingGates:
  - name: example.com/foo
  - name: example.com/bar
```




```
NAME        READY   STATUS
test-pod    0/1     Running
```

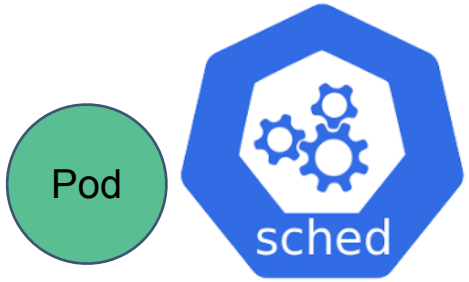
KEP-3022: MinDomains in PodTopologySpread (GA)



Topology Spread Scheduling

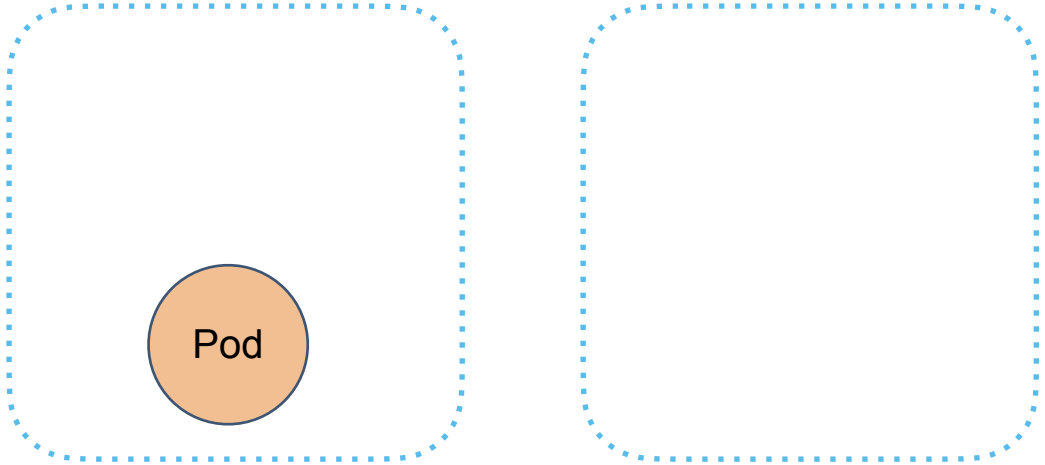
Domain-1

KEP-3022: MinDomains in PodTopologySpread (GA)



Topology Spread Scheduling **with** **minDomains=2**

Pending



Domain-1 **Domain-2**

Karpenter

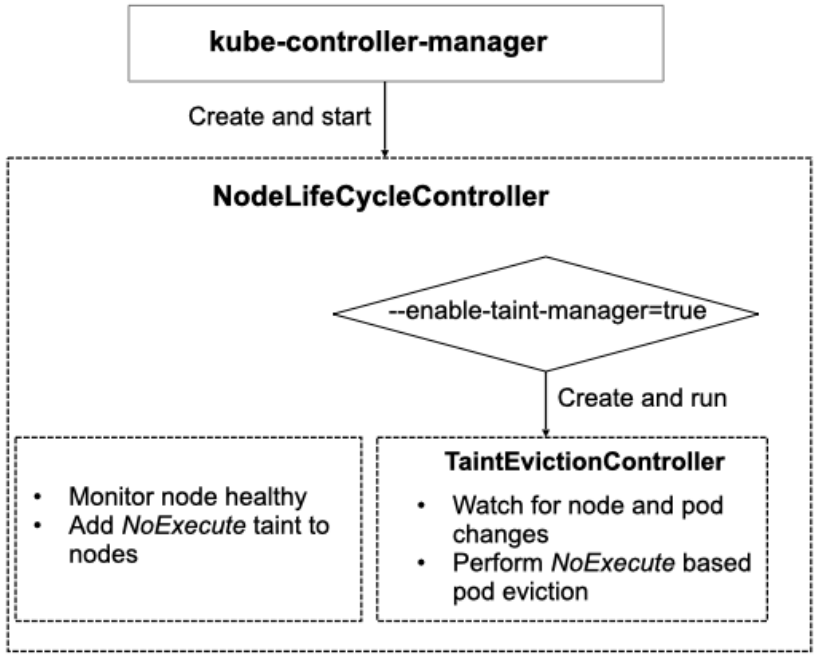
ClusterAutoscaler

KEP-3022: MinDomains in PodTopologySpread (GA)

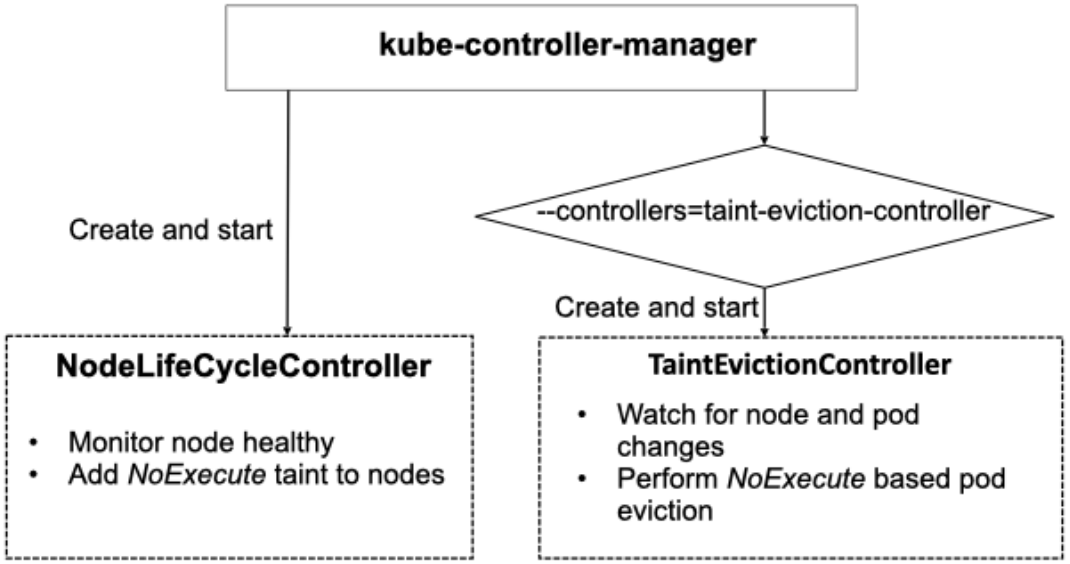
How to configure:

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: example-pod
spec:
  # Configure a topology spread constraint
  topologySpreadConstraints:
    - maxSkew: <integer>
      minDomains: <integer> # optional;
      topologyKey: <string>
      whenUnsatisfiable: <string>
      labelSelector: <object>
      matchLabelKeys: <list> # optional
      nodeAffinityPolicy: [Honor|Ignore] # optional;
      nodeTaintsPolicy: [Honor|Ignore] # optional;
  ### other Pod fields go here
```

KEP-3902: TaintManager decoupled from NLC (Beta)



Before



After

KEP-4247: QueueingHint for wise requeueing (Beta/OFF)

Before 1.22

An **aggressive** approach by putting **almost all** Pods back to scheduling queue when new event happens, this over-queues Pods and hence may lead to Head-of-Line (HOL) blocking

Before 1.28

A **coarse-grained** approach by allowing plugins to **register related events** which may cause failed Pods schedulable (e.g., a NodeAdd event may make a Pod failed by NodeResourceFit plugin schedulable)

What's new

A **fine-gained** approach by allowing plugins to **register callback functions** for wise decisions whether a Pod is schedulable or not.

KEP-4247: QueueingHint for wise requeueing (Beta/OFF)


```
type EnqueueExtensions interface {
    Plugin
    EventsToRegister() []ClusterEventWithHint
}
type ClusterEventWithHint struct {
    Event          ClusterEvent
    QueueingHintFn QueueingHintFn
}

type QueueingHint
                                    Pod, oldObj, newObj interface{})

    QueueSkip QueueingHint = iota // still unschedulable, skip the scheduling cycle
    Queue                         // maybe schedulable
)
```

⚠️ This is a beta feature but disabled by default because the unexpected memory increase.

KEP-3633: MatchLabelKeys in Pod(Anti)Affinity (Alpha)

MatchLabelKeys

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: example-pod
spec:
 affinity:
    podAffinity:
      requiredDuringSchedulingIgnoredDuringExecution:
      - labelSelector:
          matchExpressions:
          - key: app
            operator: In
            values:
            - database
        topologyKey: topology.kubernetes.io/zone
        matchLabelKeys:
        - pod-template-hash
```

Before Create

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: example-pod
spec:
 affinity:
    podAffinity:
      requiredDuringSchedulingIgnoredDuringExecution:
      - labelSelector:
          matchExpressions:
          - key: app
            operator: In
            values:
            - database
+         - key: pod-template-hash
+           operator: In
+           values:
+           - 765f54c55
        topologyKey: topology.kubernetes.io/zone
```

After Create

KEP-3633: MatchLabelKeys in Pod(Anti)Affinity (Alpha)

MisMatchLabelKeys

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: example-pod
spec:
 affinity:
    podAntiAffinity:
      requiredDuringSchedulingIgnoredDuringExecution:
      - labelSelector:
          matchExpressions:
          - key: tenant
            operator: Exists
        topologyKey: node-pool
        mismatchLabelKeys:
        - tenant
```

Before Create

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: example-pod
  labels:
    tenant: tenant-a
spec:
 affinity:
    podAntiAffinity:
      requiredDuringSchedulingIgnoredDuringExecution:
      - labelSelector:
          matchExpressions:
          - key: tenant
            operator: Exists
+         - key: tenant
+           operator: NotIn
+           values:
+           - tenant-a
        topologyKey: node-pool
```

After Create

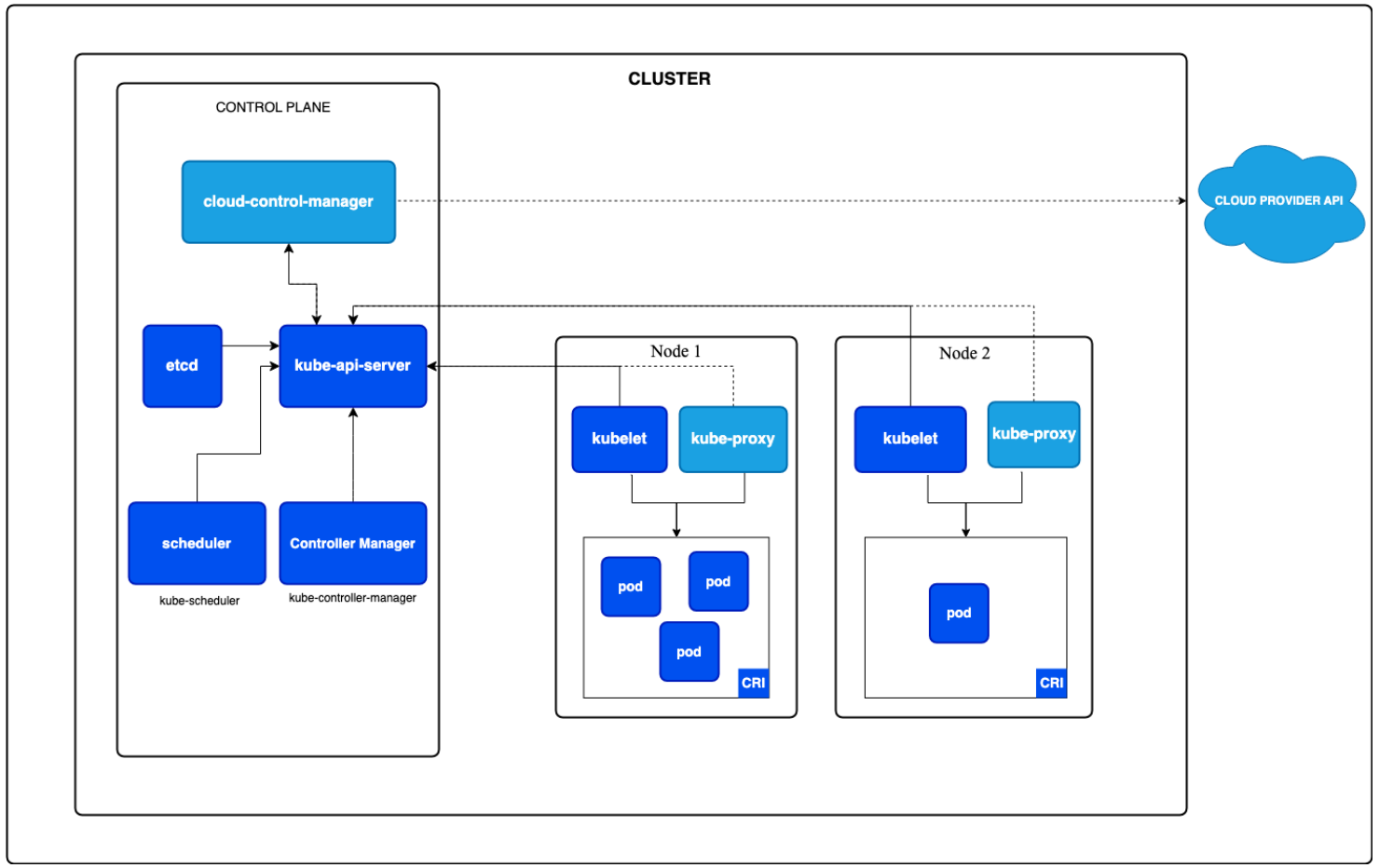
Other notable updates

- KubeSchedulerConfiguration v1beta3 **removed** in v1.29, use v1 instead
- **OrderedScoreFuncs()** introduced in preemption Interface which returns a list of ordered score functions help to customize your algorithms to select which node for eviction
- **Pending status** introduced in scheduler framework which will squeeze the unschedulable Pod into the activeQ immediately when met the certain conditions, otherwise it will wait in a backoff time

Sub-projects Updates

KWOK

KWOK is a toolkit that enables setting up a cluster of thousands of Nodes in seconds.



KWOK - Updates

✨ New Feature

- Support simulation for CPU and memory usage
- Implement Stage API support for all resources
- For `kwokctl`
  - Add `hack` subcommand to modify resources directly in etcd, bypassing the apiserver.
  - Add `record` and `replay` subcommands for resources
  - Integrate `metrics-server` to collect CPU and memory usage

🎯 Roadmap

- Extend simulation capabilities to include GPU usage and other custom metrics.
- Simulation for Volume Provisioner.
- Implement a simulation for faulty Pod and Node scenarios.

KWOK - Widely Used By

We got 2.2k stars 🚀

Clusterpedia

HEADLAMP

MCAD

MULTI CLUSTER APPLICATION DISPATCHER

Provided by KWOK Team

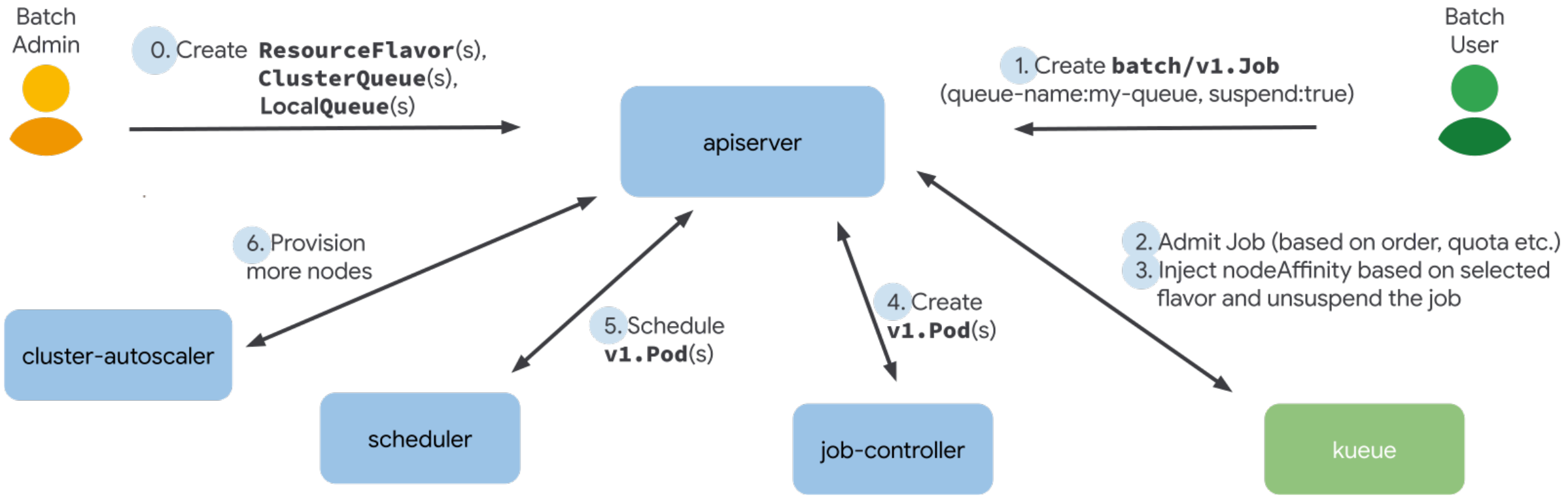
Kueue

Kueue is a Kubernetes-native job queueing system, offering:

- Priority based Job ordering and preemption
- Resource quota management
  - Resource fair sharing and borrowing between tenants
  - Fungibility: burst to on-demand, spot, other models
  - Multi-cluster support
- A variety of Jobs native support, BatchJob, Kubeflow Jobs, RayJob, plain Pods/PodGroups
- Work smoothly with Kubernetes components, like scheduler, autoscaler …

We got 1k stars 🚀

Kueue - Overview



Kueue - Updates

✨ New Release v0.6.0

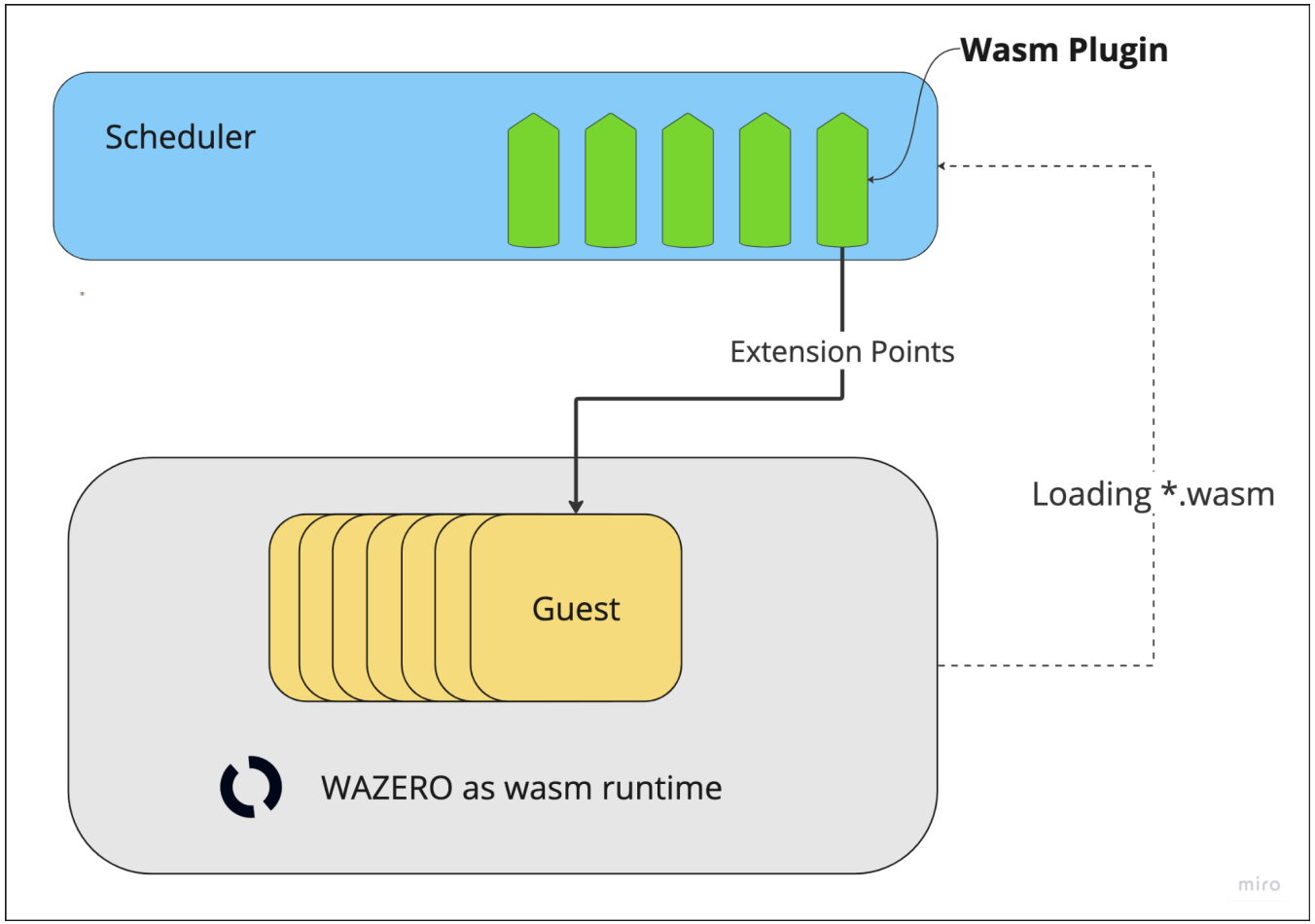
- MultiKueue: multi-cluster job dispatching
- Introduce Visibility API for the insights of pending workloads
- Introduce lendingLimit for guaranteed resource usage
- Support PodGroup for all-or-nothing queueing
- Support RayCluster
- Support for preemption while borrowing

🎯 Roadmap

- KEP-1714: DRF-like fair sharing
- KEP-79: Hierarchical Cohorts
- Integration with Kserve & Kubeflow Pipeline

Kube-scheduler-wasm-extension

Kube-scheduler-wasm-extension, a whole new way to extend the scheduler dynamically.



- No need to recompile the scheduler
- Faster than scheduler http extenders
- Similar experience vs developing traditional scheduler plugins
- Safe sandbox

- Still slower than in-memory function call

Kube-scheduler-wasm-extension - How To Use

Step1: Recompile scheduler with host WasmPlugin

Step2: Enable the guest plugins

```yaml
apiVersion: kubescheduler.config.k8s.io/v1
kind: KubeSchedulerConfiguration
profiles:
  - plugins:
      multipoint:
        enabled:
          - name: wasmplugin1
          - name: wasmplugin2
    pluginConfig:
      - name: wasmplugin1
        args:
          guestPath: "/path/to/wasm-plugin1"
      - name: wasmplugin2
        args:
          guestPath: "/path/to/wasm-plugin2"
```

👀 **Close the to first release !**

./examples

Scheduler Plugins

🧪 New incubating Plugins

- System call based scheduling (KEP 399) - collaborate with security-profiles-operator
- Disk IO Aware Scheduling (KEP 624)

✨ Stable Plugins

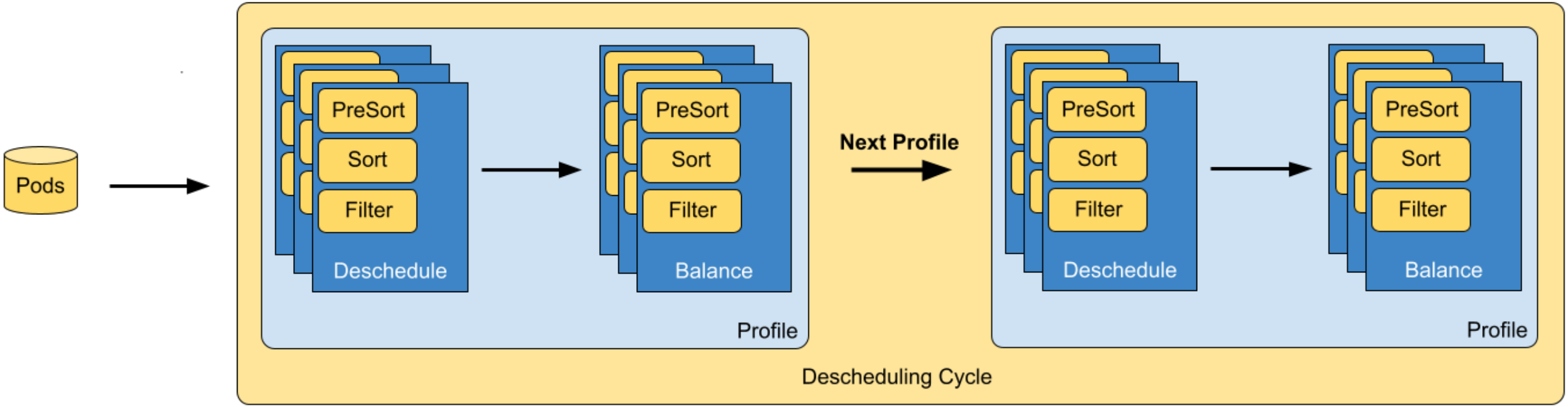
- Co-scheduling
- Capacity Scheduling (a.k.a ElasticQuota)
- Node Resource Topology
- Trimaran
  - TargetLoadPacking
  - LoadVariationRiskBalancing
  - LowRiskOverCommitment
- Network aware scheduling

We got 1k stars 🚀

Descheduler

DESCHEDULER

Rebalance clusters by evicting pods for better allocation



Descheduler Framework

Descheduler - Updates

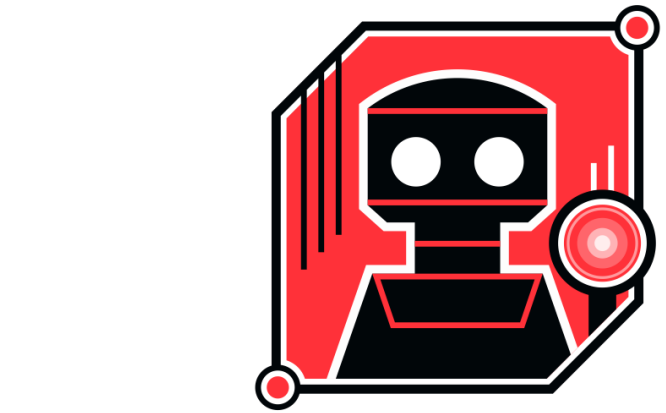
✨ New Release v0.29

- PodTopologySpread: handle nodeAffinityPolicy and nodeTaintsPolicy constraints
- PodTopologySpread: support matchLabelKeys
- PodLifeTime plugin considers container status ImagePullBackOff

🚀 Others

- UX improvements (e.g. docs, helms, logs)
- CVE fixes

🎉 We got 5 new contributors for this release.



DESCHEDULER

Join us

- Slack#SIG-Scheduling
- Mailing list
- SIG-Scheduling bi-weekly meeting (Thursdays 10:00 PST / 17:00 UTC)
- Descheduler bi-weekly meeting (Tuesdays 17:00 UTC)

🎉 **Thanks to all the contributions, the community couldn't be better without you !!!**

Q & A

Rate and leave your feedbacks!