
Proposal: allow the kubelet to trigger the rescheduling of the pods

Francesco Romani • 2024-01-XX
github.com: ffromani | mail: fromani@redhat.com

Elevator pitch

If the kubelet determines it cannot run a pod, it can trigger pod unbinding and enable rescheduling of the same pod.

Unbinding: deassociate a pod from a node

([incorrect] oversimplification: clear `pod.Spec.NodeName`)

How come the kubelet can't run a pod?

- Failure to allocate resources within the required constraints (easiest example: NUMA alignment)
 - Runtime errors
 - More and more responsibility being shift in the runtime
 - Device plugins (possibly)
 - CSI? (currently being explored)
-

Context

- Idea emerged during the [Kubecon NA 2023 conversations](#) about DRA
 - [Not a DRA beta graduation blocker](#)
 - Still useful feature for quite a few usecases:
 - Completely fix [runaway pod creation](#)
 - with and without NUMA-aware scheduling
 - DRA optimization/enhancement
 - CSI improvement?
 - ...
-

Alternatives

- Descheduler
 - Can't prevent the scheduler to pick again a "bad" node -> runaway pod creation again
 - Race with controllers
 - Good fit for non-yet-running pods?
 - Core issue solved by non-core component?
 - What else?
 - ...
-

Current status / Next steps

- [First design draft ready](#)
 - Folks already commenting and giving feedback (thanks!)
 - Reaching out **wg-batch (jan 4 2024) sig-node (jan 9 2024), sig-scheduling (jan 11 2024), sig-arch (TBD)** for go/no-go about the idea
 - Reaching out the community
 - KEP draft for 1.30/1.31 depending on feedback
 - Code PoC
-

Early Feedback

- Rescheduling seems to hint scheduler(/cluster) need more data
 - Flow should cover exceptions (limited scope)
 - Unbinding endpoint must be properly secured
 - `pod.Spec.NodeName` becomes mutable: need API and arch review
 - Avoid moving towards “live migration” (significant pod lifecycle changes)
-

Questions? comment?

[design document](#)

K8S slack: fromani

github.com: ffromani

mail: fromani@redhat.com
