KubeCon CloudNativeCon
North America 2026

CRI-O: Plug It, Tune It, Share It — Extending and Accelerating Your Runtime

Salt Lake City, Utah, United States





Sohan Kunkerkar Ayato Tokubi

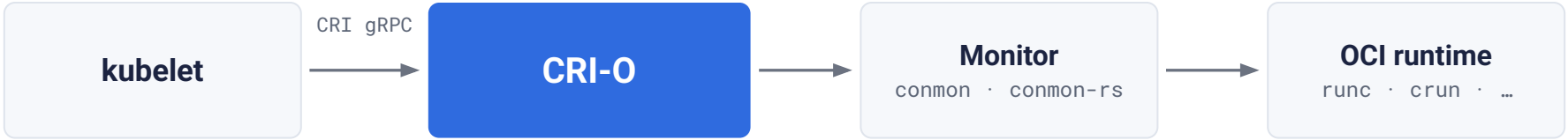
Overview

- What is CRI-O?
- CRI-O updates
- NRI: plugins, container settings, devices, networking, hooks
- Ulimits, stop signals, exec identity and writable cgroups
- MemoryQoS: node memory reservation policies
- VPA: reacting to memory pressure
- CRI list: cutting the kubelet's relist cost
- P2P image pulls with Spegel
- Future work

What is CRI-O?

INTRODUCTION

The kubelet's container runtime: the **CRI**, built on **OCI runtimes**, and nothing beyond the CRI.



Images
`container-libs/image`

Storage
`container-libs/storage`

Networking
`CNI`

Plugins
`NRI`

Versioning

One CRI-O minor per Kubernetes minor

CRI-O 1.37 runs with Kubernetes 1.37

CRI-O updates

UPDATES

Nov 2025 → Nov 2026, people only (no bots): **1.4×** more PR authors and **three** minor releases.

PRs merged into main

232 +3%

from 225

PR authors

52 1.4×

from 38

Releases

48 +14%

from 42 · 1.35, 1.36, 1.37

v1.36 · May 2026

- CRI list streaming RPCs
- CNI plugin health checks (STATUS)
- Additional read-only artifact stores

v1.37 · September 2026

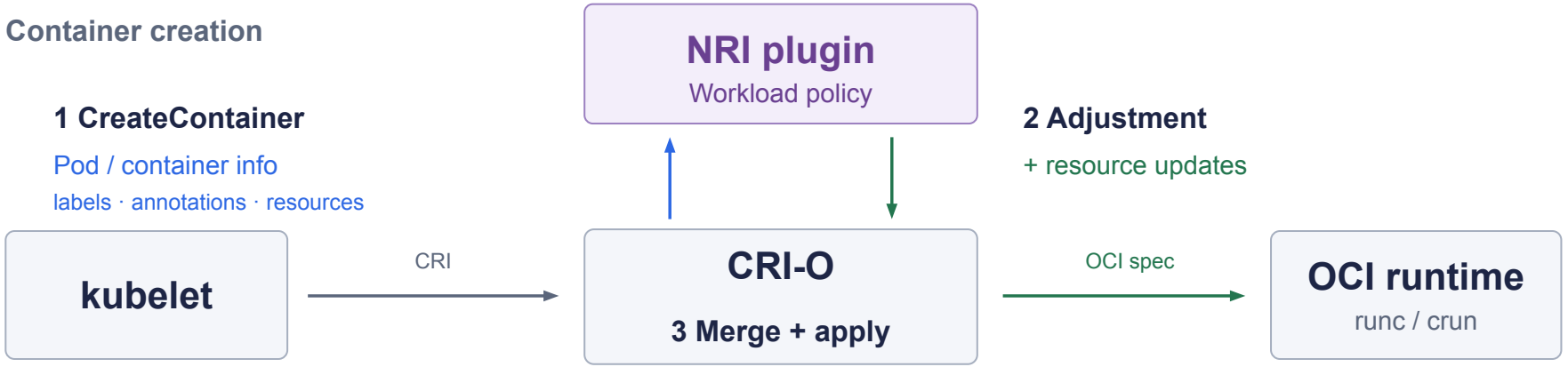
- Encrypted images for Confidential Containers
- gVisor shim (runsc)
- Checkpoint only by default, restore opt-in

Merge commits on cri-o/cri-o main, dependabot excluded. Previous period: Nov 2024 → Nov 2025.

NRI Overview

KubeCon CloudNativeCon North America 2026

Node Resource Interface — a common plugin interface for CRI-O and containerd.



Creation flow shown. Available responses depend on the event.

Adjust at creation

env · mounts · ulimits

Update resources

Existing containers · CPU / memory

React to lifecycle events

e.g. start / stop notifications

Sources: NRI overview · NRI API · CRI-O integration

Container Settings

Apply workload-specific settings when the container is created.

Examples of creation-time adjustments

Environment · Mounts · Rlimits · Arguments · Devices / CDI · OCI hooks · CPU / memory · …

Example: set nofile for a file server

Select the workload

Pod label

`app=file-server`

Select its target container

Request an adjustment

```
ContainerAdjustment

rlimits: RLIMIT_NOFILE

soft = hard = 65536
```

Apply at creation

CRI-O → OCI spec

```
{ "process": {
    "rlimits": [{
      "type": "RLIMIT_NOFILE",
      "soft": 65536, "hard": 65536
    }]
} }
```

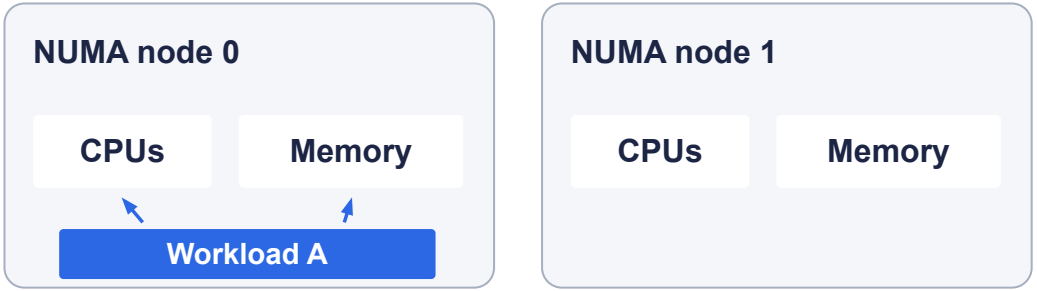
Sources: NRI API v0.12.2 · CRI-O v1.37.3 · OCI rlimits

CPU and Memory

Choose CPU and memory placement with workload-aware policies.

Example: NUMA-local placement

Illustrative placement within one node



Topology Aware

Select a topology-aware resource pool

NRI adjustment / update

CPU set + memory-node set

Balloons

Group workloads into CPU pools

Memtierd

Swap idle data out of RAM

Koordinator

Share resources: services + batch jobs

Sources: Topology Aware · Balloons · Memtierd · Koordinator · NRI integration · NRI API

Devices and CDI

A GPU needs more than a device node.

CDI = Container Device Interface

NRI plugin

Select a device

Return its CDI name

→

CRI-O

Resolve the CDI name

Read the local CDI spec

→

OCI spec

device nodes · mounts

env · hooks

NVIDIA GPU Operator

Expose GPUs to management containers

dra-driver-cpu

CDI passes CPU assignments; NRI sets cpusets

Requires matching CDI specs on the node.

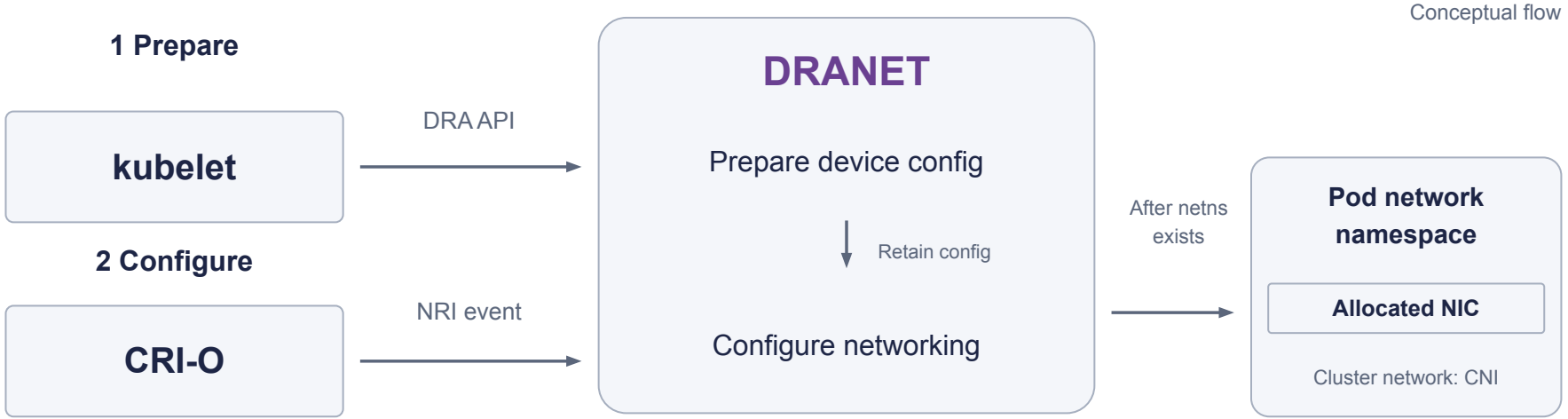
Sources: CDI spec · GPU Operator NRI · dra-driver-cpu · CRI-O CDI

Networking

KubeCon CloudNativeCon North America 2026

An allocated NIC still needs Pod network setup.

DRANET: DRA preparation + NRI lifecycle events



DRA prepares device configuration; NRI triggers Pod network setup.

Sources: DRANET README · DRANET design

Hooks and Debugging

Register lifecycle work and inspect plugin behavior.

Register a hook



Execute at the OCI stage

Inspect the plugin chain

logger — **What events arrived?**

Log subscribed requests and events.

differ — **What did plugins change?**

Compare changes before / after plugins.

Sources: hook-injector · logger · differ · OCI hooks

Using NRI

Deploy a trusted node extension with clear ownership and failure handling.

Choose how to run it

Runtime-launched

CRI-O → plugin binary

Externally launched

systemd / DaemonSet → NRI socket

Decide before rollout

Trust socket access

Plugins can change container settings.

Assign setting ownership

Coordinate plugins and kubelet managers.

Check failure behavior

Error ≠ timeout / disconnect. Validate at creation.

Sources: CRI-O 1.37.3 config · Validation plugins · NRI 0.12.2 calls

Ulimits and Stop Signals

KubeCon CloudNativeCon North America 2026

Workload-level settings are moving into standard Pod APIs.

Per-container ulimits

A file server needs a higher nofile limit.

Proposed Pod API

`securityContext.ulimits`

Upstream PR open

Container stop signals

An app needs SIGUSR1 for graceful shutdown.

Pod API

`lifecycle.stopSignal`

CRI-O runtime support since 1.33

Sources: KEP-5758 · Kubernetes #137023 · KEP-4960 · CRI-O #9086

Checked: 2026-11-04

Exec Identity and Writable Cgroups

New controls need both an API contract and runtime implementation.

Exec session identity

Which API request started this exec process?

Proposed flow · correlate with audit logs



CRI-O #9956: open

Writable cgroups

Let workers manage their child cgroups.

Own subtree · parent limits remain in force



CRI-O #10384 open

Sources: KEP-6035 · CRI-O #9956 · KEP-5474 · CRI API #139824 · CRI-O #10384

Status checked: 2026-11-04

MemoryQoS: What Beta Added in 1.37

KubeCon CloudNativeCon North America 2026

The `MemoryQoS` gate is Beta and on by default. Here is the work that got it there.

Recap from EU26: requests.memory becomes `memory.min` for Guaranteed and `memory.low` for Burstable, on cgroup v2

1 Safe default

Throttling is opt-in: `memoryThrottlingFactor` went from 0.9 to nil, so no memory.high until you set it.

Reservation stays `None` by default.

`kubernetes#140007`

2 Clean rollback

Gate off: the kubelet resets memory.min and memory.low at startup.

memory.high goes back to max on new, restarted and resized containers.

`kubernetes#138903 · #139377`

3 Observability

Counts of memory.high and memory.max hits, from memory.events:

`container_memory_events_high_total · max_total`

`cadvisor#3870 · /metrics/cadvisor · 1.37`

4 Benchmarks

Tiered protection, memory.high throttling, kubelet overhead and rollback safety.

`reports linked in KEP-2570`

Upgrading to 1.37 changes nothing until you set a knob

KEP-2570 Beta v1.37 · a 1.36 node with the Alpha gate on loses the implicit 0.9 factor on upgrade

MemoryQoS 1.38: Why Soft and Hard

KubeCon CloudNativeCon North America 2026

TieredReservation locks every Guaranteed request. One node-wide rule doesn't fit a mixed node.

Node: 56 GiB for Pods

Pod A Guaranteed · 32 GiB, 24 GiB idle page cache

Pod B Burstable · request 16 GiB, grows to 26 GiB

TieredReservation

1.36+

`A: memory.min · B: memory.low`



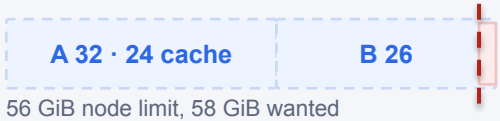
56 GiB node limit, 58 GiB wanted

B is OOM-killed or evicted. The kernel cannot reclaim A's idle cache.

Soft

1.38 · planned

`A and B: memory.low`



56 GiB node limit, 58 GiB wanted

Both run. Under pressure the kernel reclaims 2 GiB of A's idle cache.

Hard

1.38 · planned

`A and B: memory.min`



56 GiB node limit, 58 GiB wanted

B is OOM-killed or evicted. Use on dedicated, latency-sensitive nodes.

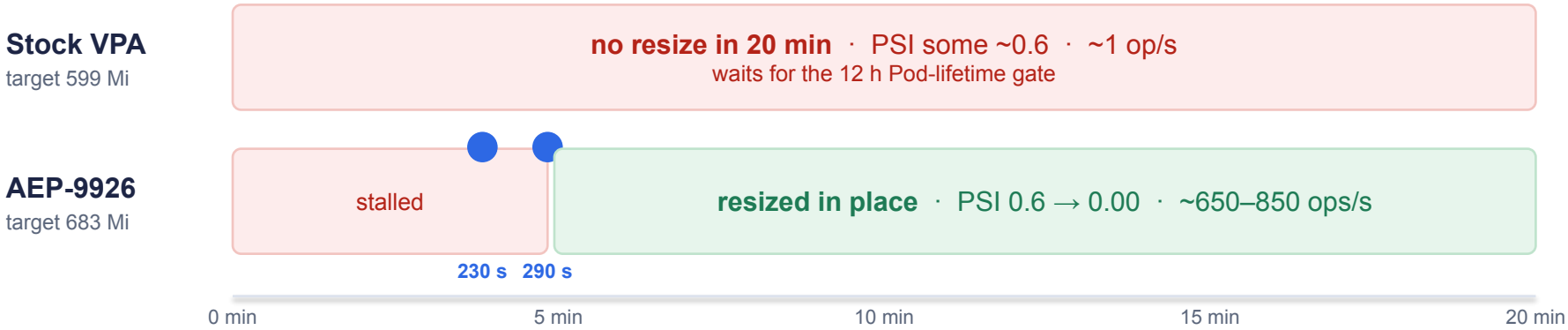
Planned Alpha gate `NodeMemoryReservationPolicy`, off by default · per-Pod control: separate KEP

Example sizes; B grows with anon memory, no swap · why: kubernetes#140246 · kubelet PR kubernetes#142854 (draft)

VPA: OOM Has a Fast Path, Stalls Don't

A rollout needs more memory. The new Pods stall on reclaim, and nothing kills them.

Deployment · 2 replicas · ~503 Mi in total	8 days of VPA history	one node · MemoryQoS on



Why it waits: the updater acts outside [lowerBound, upperBound] or after 12 h. History keeps the lower bound under the request.

AEP-9926 Story 1 · measured on one node, synthetic workload, history injected by checkpoint · autoscaler#10377

VPA: Reactive Pressure Detection

Opt in per container with `pressureDetection: Enabled` . No new update mode.

One container: request 256 Mi · limit 512 Mi · kubelet Summary API polled every 60 s

1 Observe

41.9 s → 55.7 s

PSI some.total in two polls 60 s apart

2 Detect

0.23 > 0.10
0.95 ≥ 0.90

stall rate 13.8 / 60
working set 486 / 512 Mi
both, two windows in a row

3 Record

≈ 559 Mi

max(486 + 64, 486 × 1.15)
peak into the histogram,
lastPressureTime in status

4 Resize

in place

within 10 min
raise only, never evict

Open in review: kubelet transport · reuse of the OOM bump flags · threshold defaults

AEP-9926 worked example · POC autoscaler#10376 · thresholds measured on one synthetic workload

CRI List: The Kubelet Still Pays

KubeCon CloudNativeCon North America 2026

Every second, PLEG lists every container and decodes all of it, to find the few that changed.

Dense node: ~18k containers with realistic annotations (measured on master)

PLEG
every 1 s

List →

CRI-O
returns all ~18k, every field

→ **kubelet decodes**
1.57M allocs · ~79 MB

→ **PLEG events**
use id · state · pod UID

→ **events**
the few that changed

Fields in each Container

`id` `state` `pod UID label` `metadata` `image`

`created_at` `hash annotation` `the rest`

solid: PLEG events · light: relist conversion · dashed: not needed by PLEG

When the list gets too big

response > 16 MiB → **ListContainers fails** → **PLEG not healthy**

Node goes NotReady. Container GC needs the same list, so it stops too.

Streaming (KEP-5825) lifts the 16 MiB cap. It doesn't cut the decode cost.

kubernetes#142022 · measured on master with ~18k containers · 16 MiB = kubelet CRI client max message size

CRI List: Projection and Delta Listing

KubeCon CloudNativeCon North America 2026

Two opt-in, additive changes to the list calls · `kubernetes#142022`

1 Field projection

Ask only for the fields PLEG reads

`id` `state` `labels` `metadata` `image` …

−45% memory **−65%** wire **−34%** allocs

measured

Old runtimes ignore the request and send everything.

2 Delta listing

Return only what changed since the last relist



≥99% less, at ≤1% churn

modeled

Churn isn't measured upstream yet. Old kubelets never send the request.

triage/accepted in SIG Node · KEP next · benchmarks and feedback welcome

The client declares the fields per call. The set must cover what GetPods reads: metadata, image, created_at, the hash annotation

Spegel + CRI-O: How We Got Here

KubeCon CloudNativeCon North America 2026

A peer must serve the exact compressed blob that the registry digest names.

containerd

Keeps the compressed blob (`discard_unpacked_layers` is false by default), so Spegel serves it as is.

CRI-O (container-libs storage)

Extracts the layer and drops the blob. Compressing it again is not guaranteed to give the same digest.

1 Rough demo
KubeCon EU 2026

A CRI-O-side blob cache feeds Spegel.

`cri-o#9628` · `spegel#1119`

→

2 Feedback
After the demo

Let container-libs own it: the blob lives and dies with its layer.

→

3 Proposal
July 2026

Opt-in PreserveCompressedBlob in container-libs storage (overlay).

`container-libs#1011` · `#1012`

The storage design is still under discussion with the container-libs maintainers

Spegel + CRI-O: Where We Are

KubeCon CloudNativeCon North America 2026

Work in progress. Three questions shape the container-libs design.

1 Which blob?

One uncompressed layer can match several compressed blobs. #1012 now returns the digest with the path.

2 Which pulls?

Partial pulls (zstd:chunked) and additional layer stores never see the whole blob. The API must say when nothing was kept.

3 Why P2P?

HA registry mirrors cover availability. P2P must show the bandwidth case: many nodes, large images, same time.

Pending work

`container-libs#1012`
storage PR · changes requested

→

`cri-o#9628`
CRI-O side cache · needs rebase

→

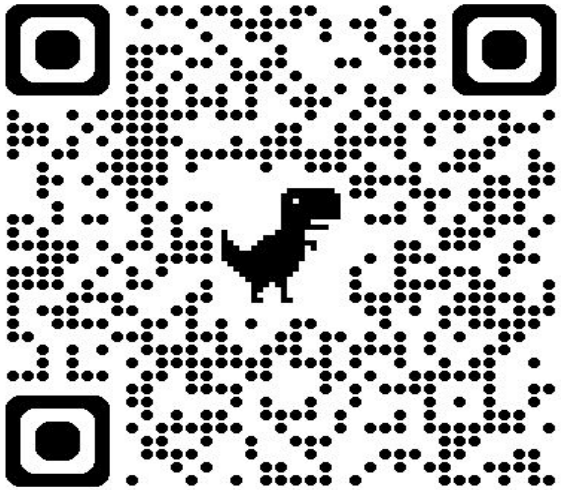
`spegel#1119`
CRI-O support · WIP, waits on #1011

Need P2P pulls on CRI-O? Bring your use case to `container-libs#1011`

Users asking for it: cri-o#9628 (June 2026), spegel#1119 (August 2026)

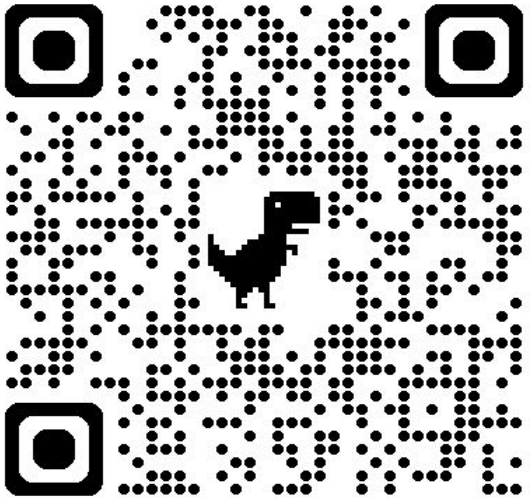
Future Work

Explore and contribute to CRI-O's future roadmap

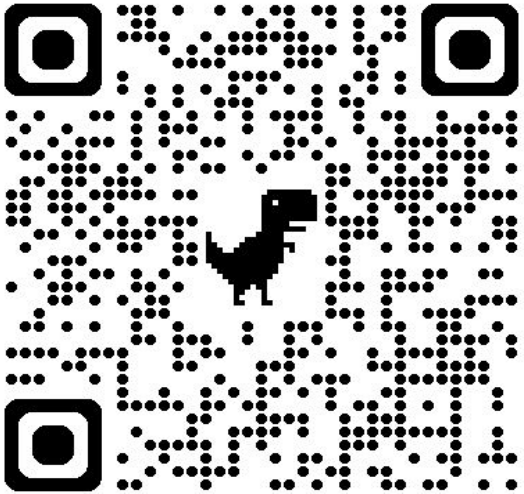


Reach out to us

KubeCon CloudNativeCon North America 2026



GitHub



Slack

KubeCon CloudNativeCon
North America 2026

Thanks!

Questions?

CRI-O: Plug It, Tune It, Share It – Extending and Accelerating Your Runtime

Rate this session

