



**KubeCon**



**CloudNativeCon**

THE LINUX FOUNDATION



**AI\_dev**  
Open Source GenAI & ML Summit

---

**China 2024**

---



China 2024

# containerd: Project Update and Deep Dive

Akhil Mohan, VMware by Broadcom

Iceber Gu(蔡威), DaoCloud



KubeCon



CloudNativeCon



China 2024

# Project Updates

# Usage Growth



KubeCon



CloudNativeCon



China 2024



PENETRATION OF CNCF TOP TEN GRADUATED OR INCUBATING CNCF PROJECTS BEING USED IN PRODUCTION OR EVALUATED FROM 2022 TO 2023

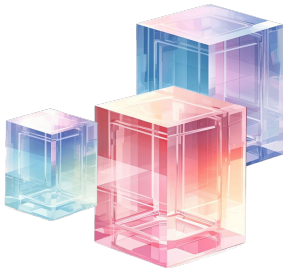


2022  
68%

2023  
74%

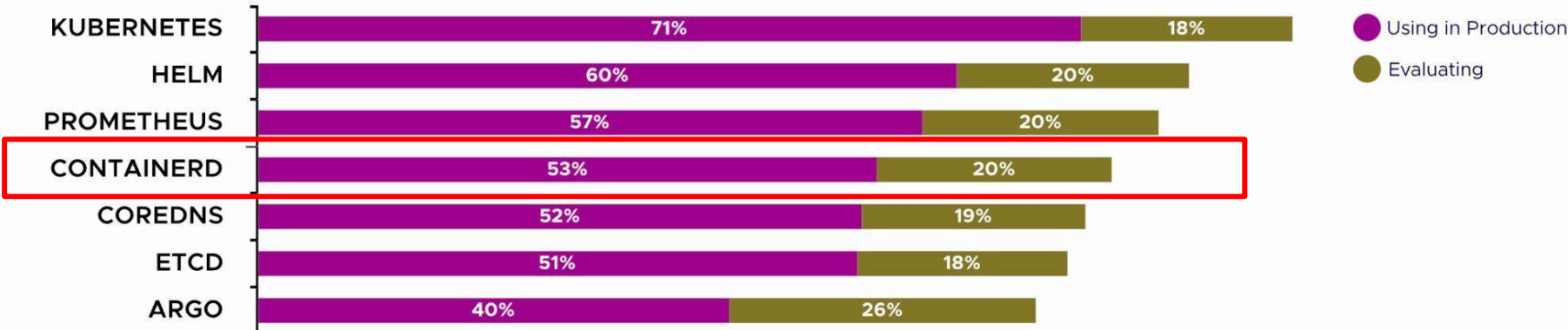
CONTAINERD

## CNCF 2023 ANNUAL SURVEY



Published in April, 2024

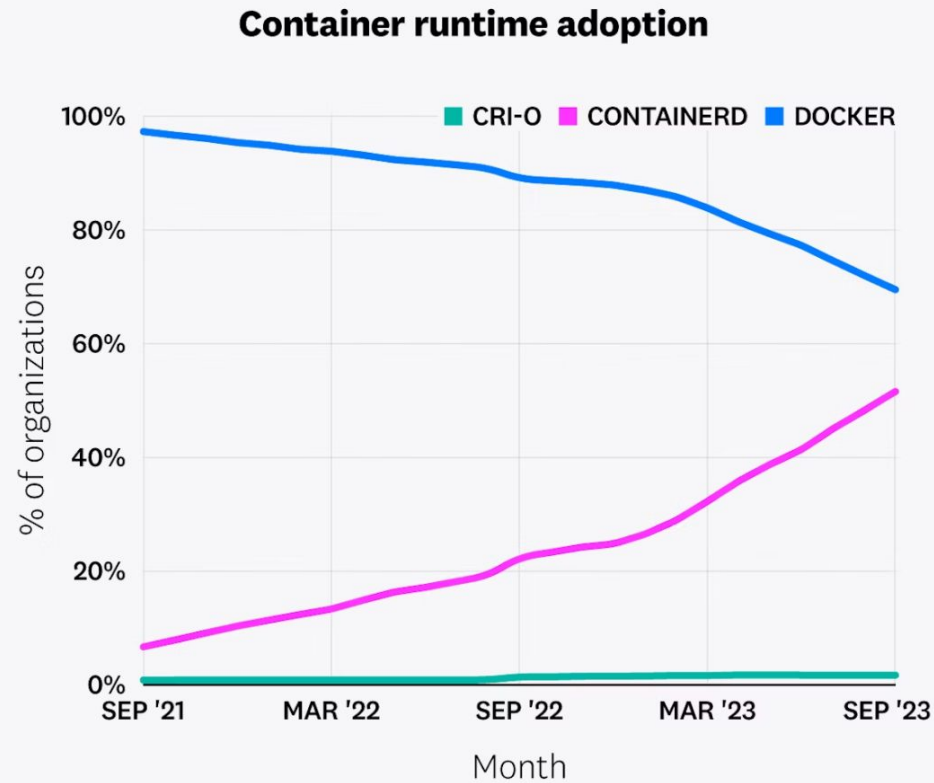
Which Of These Graduated CNCF Projects Is Your Organization Using In Production Or Evaluating?



# Usage Growth



China 2024



**Containerd  
adoption has  
more than  
doubled in  
the last year**

Source: Datadog

Containerd has continued to replace Docker as a predominant container runtime

# Community Growth

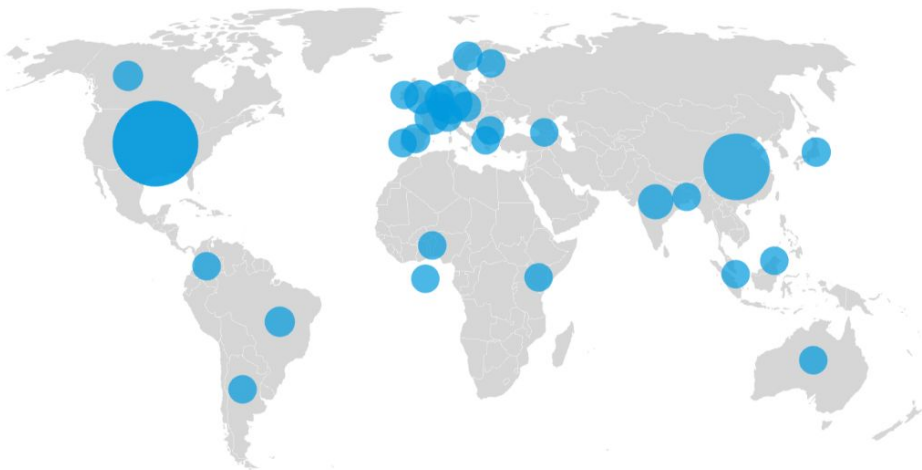
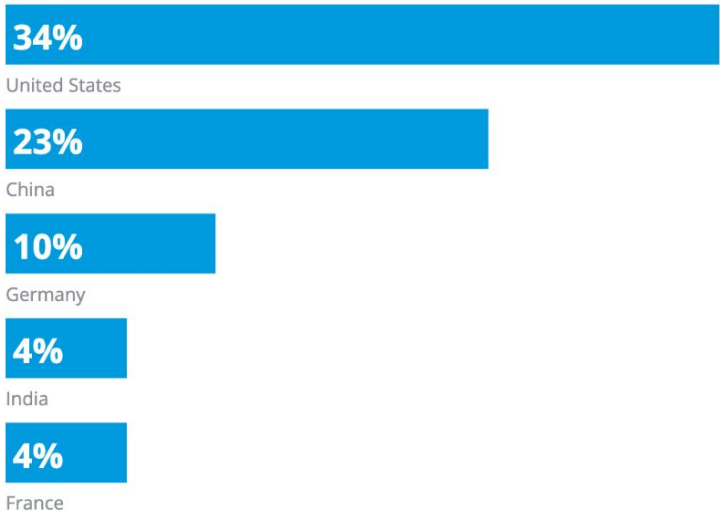


China 2024

## containerd/containerd last 12 month

Contributor:	Commits:	Issues:	Forks:	Stars:	PRs:
746	1513	372	372	2416	372

### TOP 5 REGIONS



- Kubernetes (kubernetes.io) 3567 authors
- OpenTelemetry (opentelemetry.io) 1591 authors
- Argo (argoproj.github.io) 913 authors
- Backstage (backstage.io) 642 authors
- Prometheus (prometheus.io) 512 authors
- gRPC (grpc.io) 457 authors
- Cilium (cilium.io) 454 authors
- Envoy (www.envoyproxy.io) 424 authors
- Istio (istio.io) 364 authors
- Keycloak (keycloak.org) 328 authors
- Meshery (layer5.io/meshery) 323 authors
- Crossplane (crossplane.io) 301 authors
- Fluentd (fluentd.org) 284 authors
- Dapr (dapr.io) 276 authors
- NATS (nats.io) 275 authors
- Fluid (github.com/fluid-cloudnative) 272 authors
- containerd (containerd.io) 268 authors
- Kubeflow (kubeflow.org) 243 authors
- Knative (knative.dev) 227 authors
- KubeVirt (kubevirt.io) 226 authors
- Falco (falco.org) 192 authors
- OpenFeature (openfeature.dev) 187 authors
- 151 more

# Community Growth



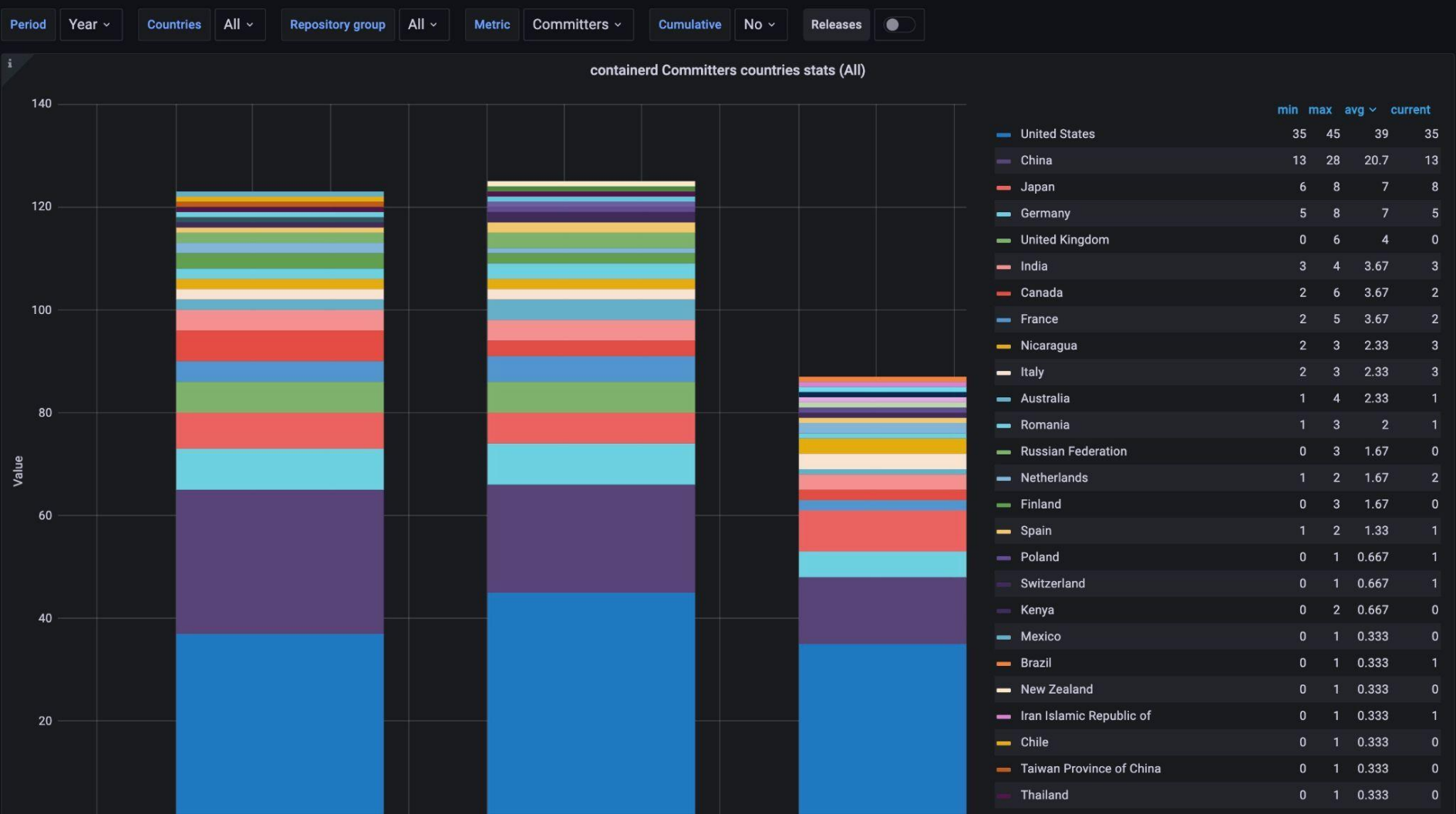
KubeCon



CloudNativeCon



China 2024



# New maintainers



China 2024



**Henry Wang**

henry118



**Ruiwen Zhao**

ruiwen-zhao



**Akhil Mohan**

akhilerm

\* since kubecon china 2023



# Adoption in Kubernetes distros



China 2024

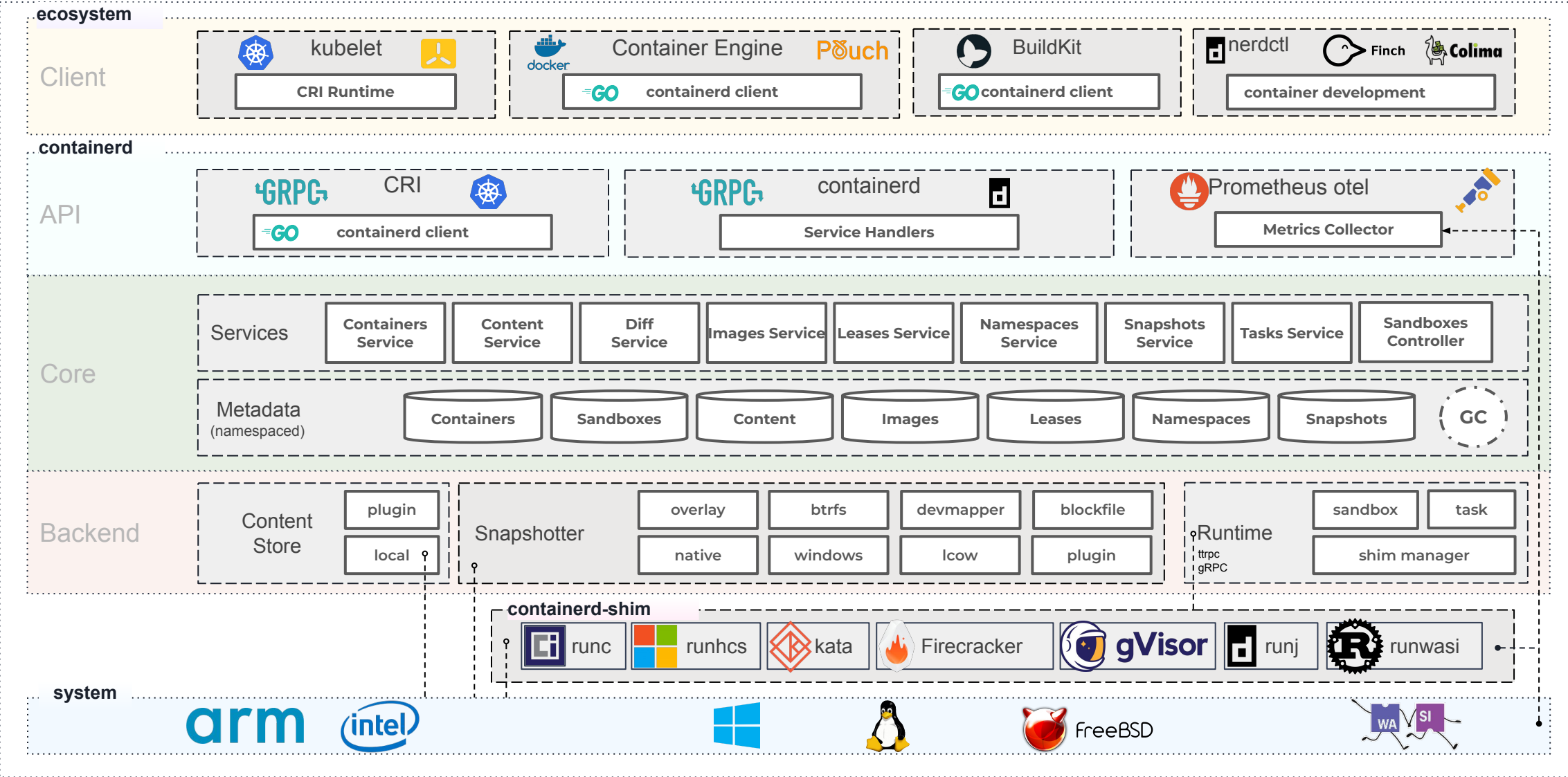
- Alibaba Cloud Container Service for Kubernetes
- Amazon Elastic Kubernetes Service
- Azure Kubernetes Service
- Google Kubernetes Engine
- Huawei Cloud Cloud Container Engine
- IBM Cloud Kubernetes Service
- Rancher K3s
- VMware Tanzu
- Volcengine Kubernetes Engine



# containerd Architecture



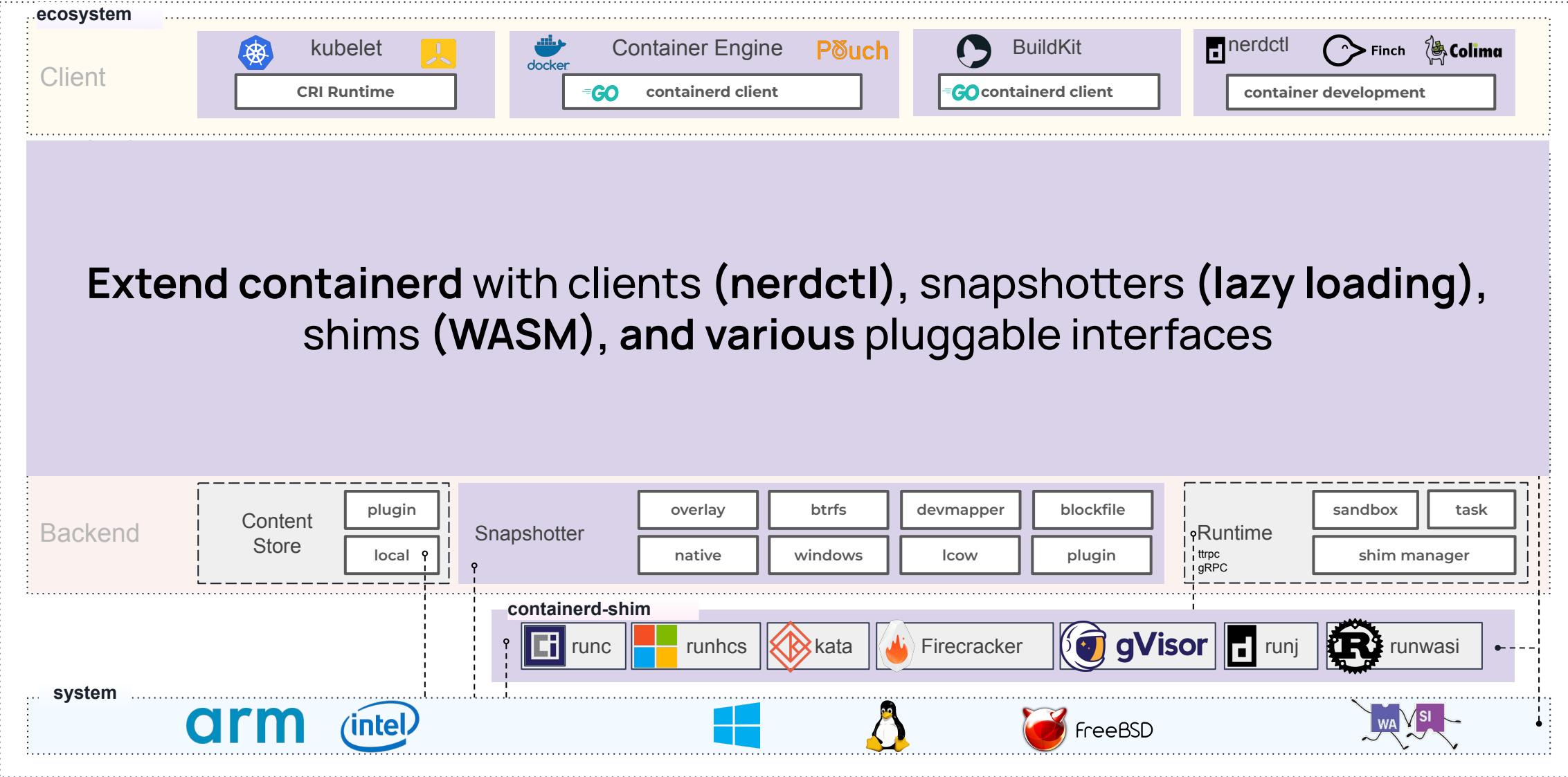
China 2024



# containerd Architecture



China 2024



# Supported Releases



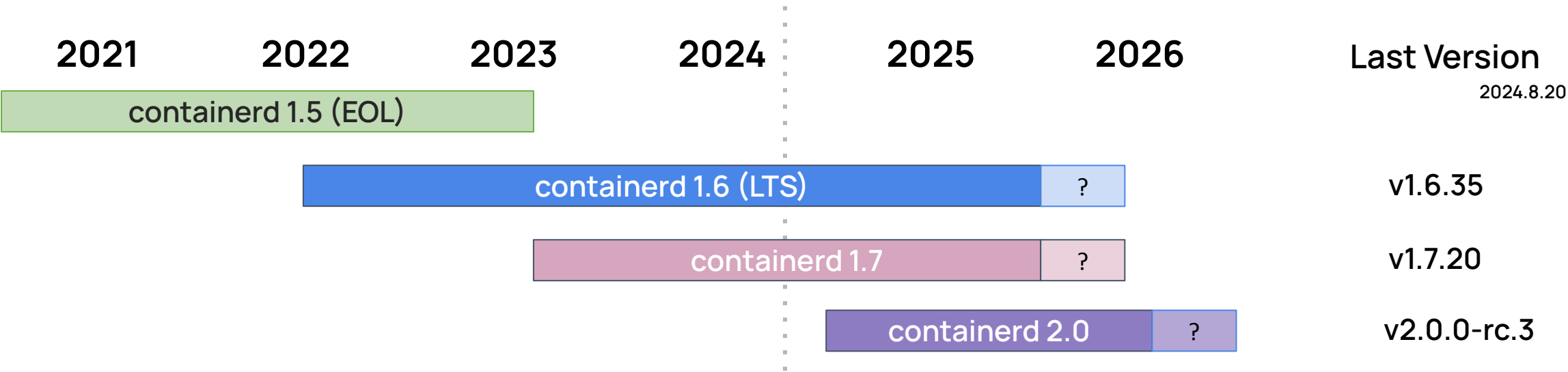
KubeCon



CloudNativeCon



China 2024



Release	Status	Start	End of Life
<a href="#">1.5</a>	End of Life	May 3, 2021	February 28, 2023
<a href="#">1.6</a>	LTS	February 15, 2022	max(February 15, 2025 or next LTS + 6 months)
<a href="#">1.7</a>	Active	March 10, 2023	active(release of 2.0 + 6 months), extended(EOL of 1.6)
<a href="#">2.0</a>	Next	TBD	TBD

# Kubernetes Support



China 2024

Kubernetes Version	containerd Version	CRI Version
1.26	1.7.0+, 1.6.15+	v1, v1alpha2 **
1.27	1.7.0+, 1.6.15+	v1
1.28	1.7.0+, 1.6.15+	v1
1.29	1.7.11+, 1.6.27+	v1
1.30	2.0(wip), 1.7.13+, 1.6.28+	v1
1.31	2.0(wip), 1.7.20+, 1.6.34+	v1

⚠ containerd v1.6.x and v1.7.x support CRI *v1* and *v1alpha2* through EOL as those releases continue to support older versions of k8s, cloud providers, and other clients using CRI v1alpha2.

**CRI v1alpha2 is deprecated in v1.7 and will be removed in containerd v2.0.**

<https://github.com/containerd/containerd/blob/main/RELEASES.md#kubernetes-support>

# Subprojects



China 2024

- nerdctl ( v2.0.0-rc.1 )
- nydus-snapshotter, stargz-snapshotter
- runwasi
- rust-extensions



KubeCon



CloudNativeCon



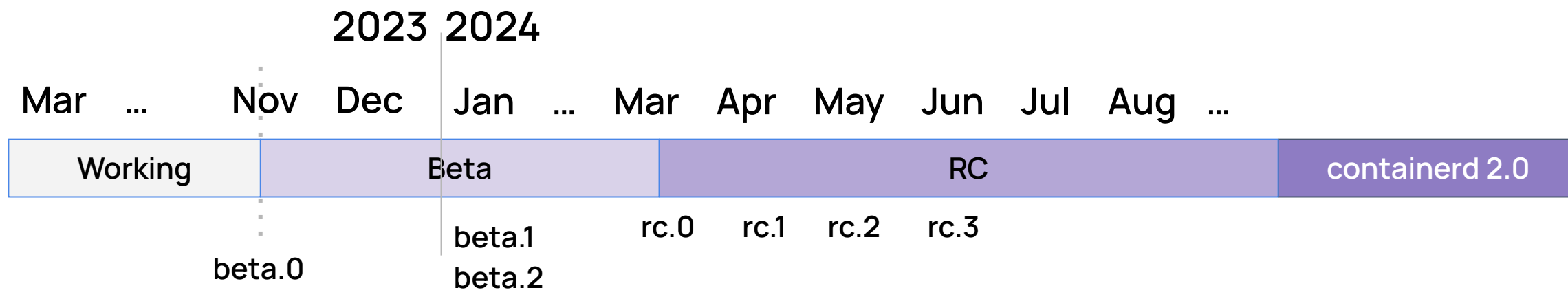
China 2024

# containerd 2.0 is coming

# Release Plan



China 2024



v2.0.0-rc.3 is released



# v2.0 - Incompatible Changes











China 2024

- Removed features
- Removed Config
- Config migration
- Project & Component restructuring

# v2.0 - Removed Features



China 2024

Component	Deprecation release	Target release for removal	Recommendation
Runtime V1 API and implementation ( <code>io.containerd.runtime.v1.linux</code> )	containerd v1.4	containerd v2.0 	Use <code>io.containerd.runc.v2</code>
Runc V1 implementation of Runtime V2 ( <code>io.containerd.runc.v1</code> )	containerd v1.4	containerd v2.0 	Use <code>io.containerd.runc.v2</code>
Built-in <code>aufs</code> snapshotter	containerd v1.5	containerd v2.0 	Use <code>overlayfs</code> snapshotter
Container label <code>containerd.io/restart.logpath</code>	containerd v1.5	containerd v2.0 	Use <code>containerd.io/restart.loguri</code> label
<code>cri-containerd-*.tar.gz</code> release bundles	containerd v1.5	containerd v2.0 	Use <code>containerd-*.tar.gz</code> bundles
Pulling Schema 1 images ( <code>application/vnd.docker.distribution.manifest.v1+prettyjvl</code> )	containerd v1.7	containerd v2.1 (Disabled in v2.0  )	Use Schema 2 or OCI images
CRI <code>v1alpha2</code>	containerd v1.7	containerd v2.0 	Use CRI <code>v1</code>
Legacy CRI implementation of podsandbox support	containerd v2.0	containerd v2.0 	
Go-Plugin library ( <code>*.so</code> ) as containerd runtime plugin	containerd v2.0	containerd v2.1	Use external plugins (proxy or binary)







- Pulling Schema 1 images has been disabled in containerd v2.0, but it still can be enabled by setting an environment variable `CONTAINERD_ENABLE_DEPRECATED_PULL_SCHEMA_1_IMAGE=1` until containerd v2.1. `ctr` users have to specify `--local` too (e.g., `ctr images pull --local`).

<https://github.com/containerd/containerd/blob/main/RELEASES.md#deprecated-features>

# v2.0 - Removed Config



China 2024

Property Group	Property	Deprecation release	Target release for removal	Recommendation
[plugins."io.containerd.grpc.v1.cri"]	systemd_cgroup	containerd v1.3	containerd v2.0 	Use <code>SystemdCgroup</code> in runc options (see below)
[plugins."io.containerd.grpc.v1.cri".containerd]	untrusted_workload_runtime	containerd v1.2	containerd v2.0 	Create <code>untrusted</code> runtime in <code>runtimes</code>
[plugins."io.containerd.grpc.v1.cri".containerd]	default_runtime	containerd v1.3	containerd v2.0 	Use <code>default_runtime_name</code>
[plugins."io.containerd.grpc.v1.cri".containerd.runtimes.*]	runtime_engine	containerd v1.3	containerd v2.0 	Use runtime v2
[plugins."io.containerd.grpc.v1.cri".containerd.runtimes.*.options.root]	root	containerd v1.3	containerd v2.0 	Use <code>options.Root</code>
[plugins."io.containerd.grpc.v1.cri".containerd.runtimes.*.options.criuPath]	criuPath	containerd v1.7	containerd v2.0 	Set <code>\$PATH</code> to the <code>criu</code> binary
[plugins."io.containerd.grpc.v1.cri".registry]	image_pull_secrets	containerd v1.3	containerd v2.1	Use <a href="#">ImagePullSecrets</a> . See also <a href="#">#8228</a> .
[plugins."io.containerd.grpc.v1.cri".registry]	config_path	containerd v1.5	containerd v2.1	Use <a href="#">config_path</a>
[plugins."io.containerd.grpc.v1.cri".registry]	mirrors	containerd v1.5	containerd v2.1	Use <a href="#">config_path</a>
[plugins."io.containerd.tracing.processor.v1.otlp"]	endpoint, protocol, insecure	containerd v1.6.29	containerd v2.0	Use <a href="#">OTLP environment variables</a> , e.g. <code>OTEL_EXPORTER_OTLP_TRACES_ENDPOINT</code> , <code>OTEL_EXPORTER_OTLP_PROTOCOL</code> , <code>OTEL_SDK_DISABLED</code>
[plugins."io.containerd.internal.v1.tracing"]	service_name, sampling_ratio	containerd v1.6.29	containerd v2.0	Instead use <a href="#">OTel environment variables</a> , e.g. <code>OTEL_SERVICE_NAME</code> , <code>OTEL_TRACES_SAMPLER*</code>

<https://github.com/containerd/containerd/blob/main/RELEASES.md#deprecated-config-properties>

# v2.0 - Config Migration



China 2024

Configuration Version	Minimum containerd version
1	v1.0.0
2	v1.3.0
3	v2.0.0

<https://github.com/containerd/containerd/blob/main/RELEASES.md#daemon-configuration>

Instead of deprecating the config v2 (containerd v1.3+) properties and format,

we introduced config migration which converts to the new config format and properties on the users behalf in v2.0

If you are confident that you don't need to rollback to containerd 1.x, you can migrate your config

```
$ containerd config migrate
```

# v2.0 - 🖐️ Containerd Config



China 2024

## Get Default Config

```
$ containerd config default
```

## Base Configuration

<https://github.com/containerd/containerd/blob/main/docs/ops.md#base-configuration>

## CRI Plugin Config Guide

<https://github.com/containerd/containerd/blob/main/docs/cri/config.md>

# v2.0 - 🖐️ Get Deprecations



China 2024

## Expose usage of deprecated features with Introspection Services and Warning Plugin

### containerd/ctr versions:

- v1.6.25+ (recommended v1.6.27+ )
- V1.7.9+ (recommended v1.7.11+ )
- v2.0

### Print with ctr command

```
$ ctr deprecations list
```

```
service Introspection {  
    // Plugins returns a list of plugins in containerd.  
    //  
    // Clients can use this to detect features and capabilities when using  
    // containerd.  
    rpc Plugins(PluginsRequest) returns (PluginsResponse);  
    // Server returns information about the containerd server  
    rpc Server(google.protobuf.Empty) returns (ServerResponse);  
    // PluginInfo returns information directly from a plugin if the plugin supports it  
    rpc PluginInfo(PluginInfoRequest) returns (PluginInfoResponse);  
}  
  
message ServerResponse {  
    string uuid = 1;  
    uint64 pid = 2;  
    uint64 pidns = 3; // PID namespace, such as 4026531836  
    repeated DeprecationWarning deprecations = 4;  
}  
  
message DeprecationWarning {  
    string id = 1;  
    string message = 2;  
    google.protobuf.Timestamp last_occurrence = 3;  
}
```

# v2.0 - Project Restructuring



China 2024

## package/dir updates

- .(root)
- cmd
- client
- api
- core
- internal
- plugins
- pkg

# v2.0 - Project Restructuring



China 2024

## package/dir updates

- **.(root)**
- cmd
- client
- api
- core
- internal
- plugins
- pkg

## root

The root directory is reserved for configuration and build files.



# v2.0 - Project Restructuring



China 2024

## package/dir updates

- .(root)
- **cmd**
- client
- api
- core
- internal
- plugins
- pkg

## cmd

Removed unsupported commands such as containerd-shim-runc-v1, and moved the main logic for shim from runtime/v2/shim to cmd/containerd-shim-runc-v2.

# v2.0 - Project Restructuring



China 2024

## package/dir updates

- .(root)
- cmd
- **client**
- api
- core
- internal
- plugins
- pkg

## client

The client go files that were scattered in the root are unified into the /client directory, so the client package changes from **containerd/containerd** to **containerd/v2/client**.

# v2.0 - Project Restructuring



China 2024

## package/dir updates

- .(root)
- cmd
- client
- **api**
- core
- internal
- plugins
- pkg

## api

All the containerd APIs are available here as a separate module

# v2.0 - Project Restructuring



China 2024

## package/dir updates

- .(root)
- cmd
- client
- api
- **core**
- internal
- plugins
- pkg

## core

This maintains the interface definitions, utility functions/modules, and some built-in implementations of the core underlying modules.

# v2.0 - Project Restructuring



China 2024

## package/dir updates

- .(root)
- cmd
- client
- api
- core
- **internal**
- plugins
- pkg

## internal

contains implementations of private modules such as cri, as well as packages that have no need to be made public and are not currently referenced by any other project.

# v2.0 - Project Restructuring



China 2024

## package/dir updates

- .(root)
- cmd
- client
- api
- core
- internal
- **plugins**
- pkg

## plugins

The vast majority of containerd's plugins are registered here via init, and also include specific implementations of some plugins, such as containerd grpc services, snapshots.

# v2.0 - Project Restructuring



China 2024

## package/dir updates

- .(root)
- cmd
- client
- api
- core
- internal
- plugins
- **pkg**

## pkg

Non-core packages used by client and containerd server. In 2.0, the contents of the pkg have been reorganized, some modules from the original pkg have been moved to others such as internal, and the contents of other modules have been moved over, such as the shim manager framework.



KubeCon



CloudNativeCon



China 2024

# New Features / Deep Dive



# v2.0 - Features



China 2024

Experimental features added in 1.7 and **officially supported in 2.0**

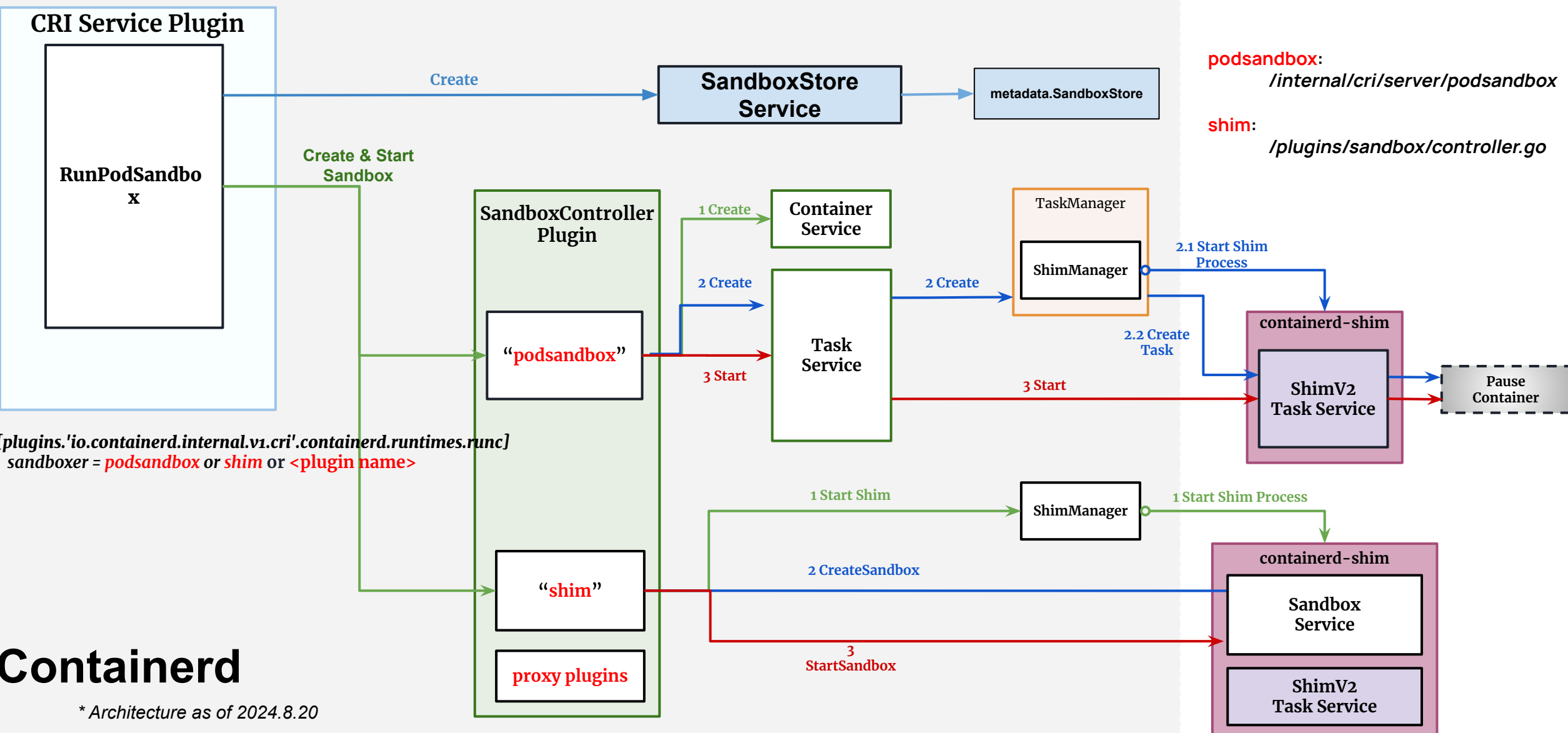
Component	Initial Release	Target Supported Release
<a href="#">Sandbox Service</a>	containerd v1.7	containerd v2.0
<a href="#">Sandbox CRI Server</a>	containerd v1.7	containerd v2.0
<a href="#">Transfer Service</a>	containerd v1.7	containerd v2.0
<a href="#">NRI in CRI Support</a>	containerd v1.7	containerd v2.0
<a href="#">gRPC Shim</a>	containerd v1.7	containerd v2.0
<a href="#">CRI Runtime Specific Snapshotter</a>	containerd v1.7	containerd v2.0
<a href="#">CRI Support for User Namespaces</a>	containerd v1.7	containerd v2.0

<https://github.com/containerd/containerd/blob/main/RELEASES.md#experimental-features>

# Sandbox API

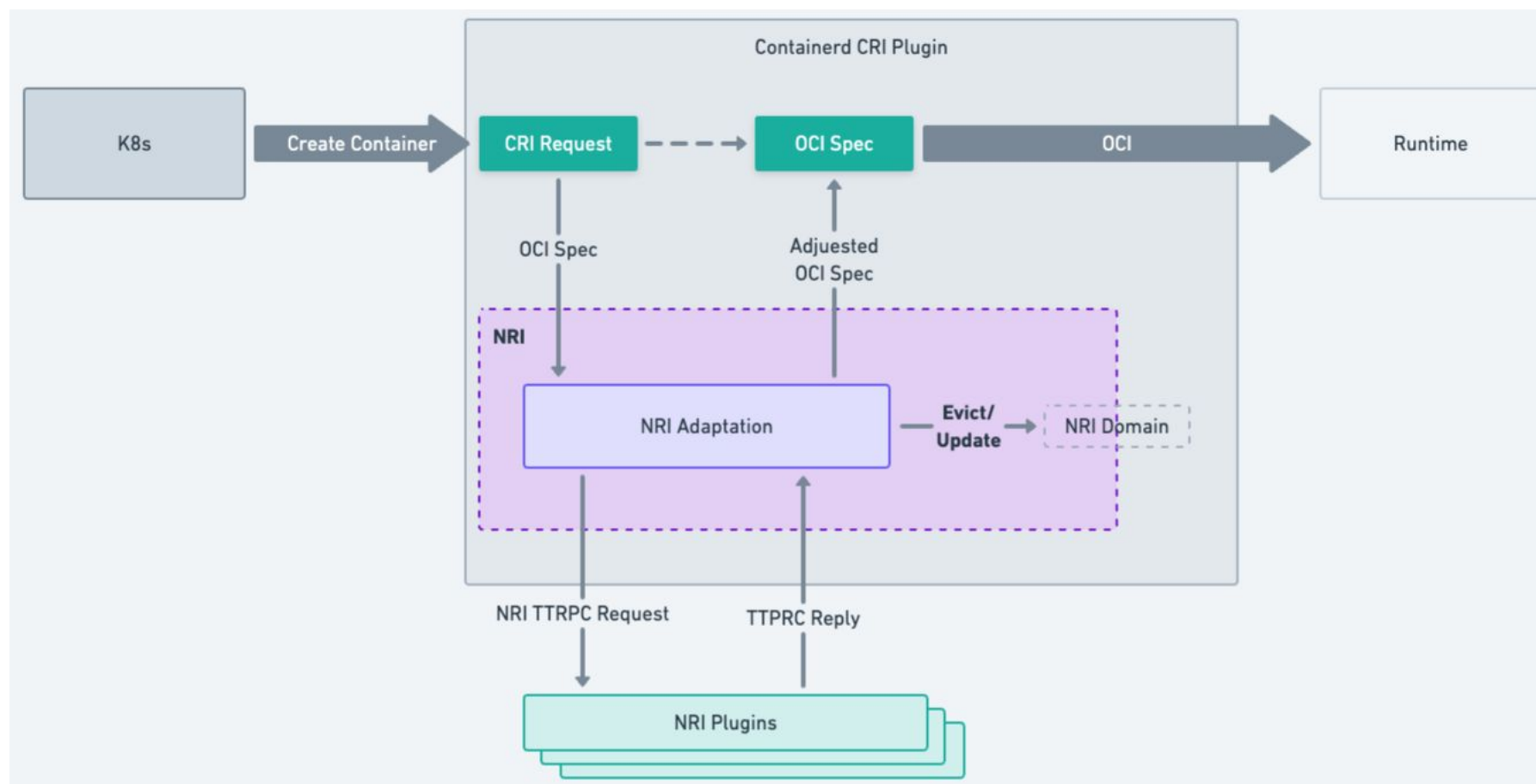


China 2024



# Node Resource Interface

## Middleware extension between CRI and OCI



# Transfer API



China 2024

Source	Destination	Description
Registry	Image Store	"pull"
Image Store	Registry	"push"
Object stream (Archive)	Image Store	"import"
Image Store	Object stream (Archive)	"export"
Object stream (Layer)	Mount/Snapshot	"unpack"
Mount/Snapshot	Object stream (Layer)	"diff"
Image Store	Image Store	"tag"
Registry	Registry	mirror registry image

```
type Transferrer interface {  
    Transfer(ctx context.Context, source interface{}, destination interface{}, opts ...Opt) error  
}
```

# Future & In-Development



China 2024

- **Transfer Service in CRI (in development)**
  - Credential Management
- **Higher level image service**
  - Simplify clients such as nerdctl and Moby
- **Higher level container service**
  - Simpler interface for starting containers
  - Use of streaming service for IO through the API

# Getting involved



China 2024

- [#containerd](#) and [#containerd-dev](#) channel on
  - CNCF Slack (<https://slack.cncf.io>)
- **Community Meeting on the 2nd and 4th Thursday of every month**
  - See CNCF Calendar for your timezone (<https://cncf.io/calendar>)
- Build something in the ecosystem!
- Discussion, issues and pull requests welcome!
  - <https://github.com/containerd/containerd>