



KubeCon



CloudNativeCon

North America 2022

BUILDING FOR THE ROAD AHEAD

DETROIT 2022

SIG Cluster Lifecycle Intro

Cecile Robert-Michon, Microsoft
Fabrizio Pandini, VMware

SIG Cluster Lifecycle Intro



BUILDING FOR THE ROAD AHEAD

DETROIT 2022



KubeCon



CloudNativeCon

North America 2022

BUILDING FOR THE ROAD AHEAD

DETROIT 2022

October 24-28, 2021



Cecile

Robert-Michon

Microsoft



Fabrizio Pandini

VMWare

first, let's introduce SIG Cluster Lifecycle

Who are we?



We are a little, peaceful and happy army!

~ 400
Developers

> 80
Companies

21
Subprojects

> 3k
*Issues**

> 5k
*PRs**

~ 70k
*Comments**

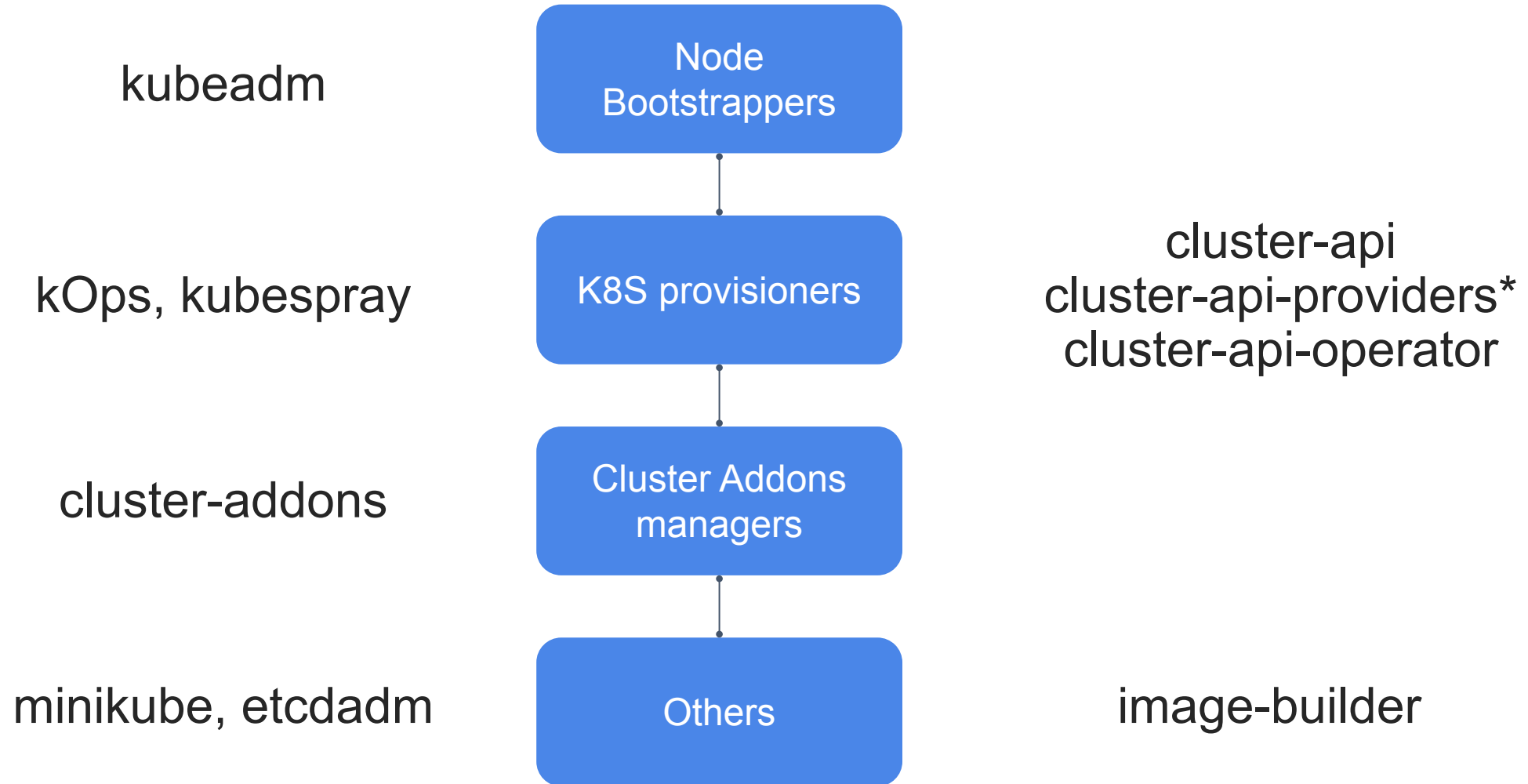
We make cluster lifecycle boring*

(*) So reliable that no one pay anymore attention to it, like cat, sed, grep etc.



“SIG Cluster Lifecycle’s objective is to simplify creation, configuration, upgrade, downgrade, and teardown of Kubernetes clusters and their components.”

The stack



(*) currently there are 12 cluster API providers hosted by the SIG: AWS, Azure, Cloudstack, DigitalOcean, GCP, IBM Cloud, kubemark, kubevirt, nested, OpenStack, Packet, VSphere

The Voltron moment



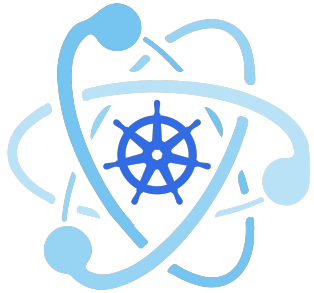
There are many projects in Cluster Lifecycle;
users can choose their own mix and also plug-in
custom components if they want

**Batteries included, but
swappable**

Using them together is
The Voltron moment*

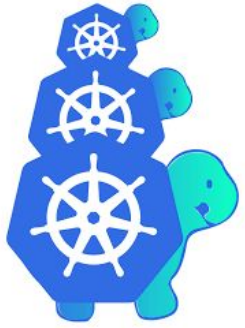
(*) Timothy St. Clair, Introduction to SIG Cluster Lifecycle, KubeCon 2019, San Diego

some additional info about key subprojects



kubeadm's task is to set up a best-practice Kubernetes node

- The user experience should be simple, and the cluster reasonably secure
- kubeadm's scope is limited, it is intended to be a building block
 - Agnostic to how exactly the machine is provisioned
 - Agnostic to how exactly the kubelet is run
 - Only ever deals with the local filesystem and the Kubernetes API
 - Setting up or favoring a specific CNI network is out of scope
- Integrate well with other higher order tools (Cluster API, kubespray, ansible/etc.)
 - Composable architecture with everything divided into phases



Cluster API provides declarative APIs to operate multiple Kubernetes clusters.

- Kubernetes all the way down:
 - `kubectl apply -f my-cluster.yaml`, `kubectl scale my-node-pool --replicas=5`
 - controllers continuously reconciling your Cluster state
- Pluggable providers, with more than 30 provider listed, +10 in the last year!
 - 23 providers targeting different type of infrastructure (AWS, Azure, etc.)
 - 7 Node bootstrappers or control-plane controllers (kubeadm, MicroK8s etc.)
- Manages a fleet of workload Clusters from a single management Cluster
- Implements an immutable Machine deployment model

a little digression, notable Cluster API updates

Introducing the release team!

Let's make Cluster API release process more reliable and predictable

- Improve communication to end users and contributors about Cluster API release cadence and target release dates
- Make Cluster API release management more sustainable:
 - Involve a bigger, more diverse set contributors in cutting releases
 - Improve tooling, documentation, and automation of release process

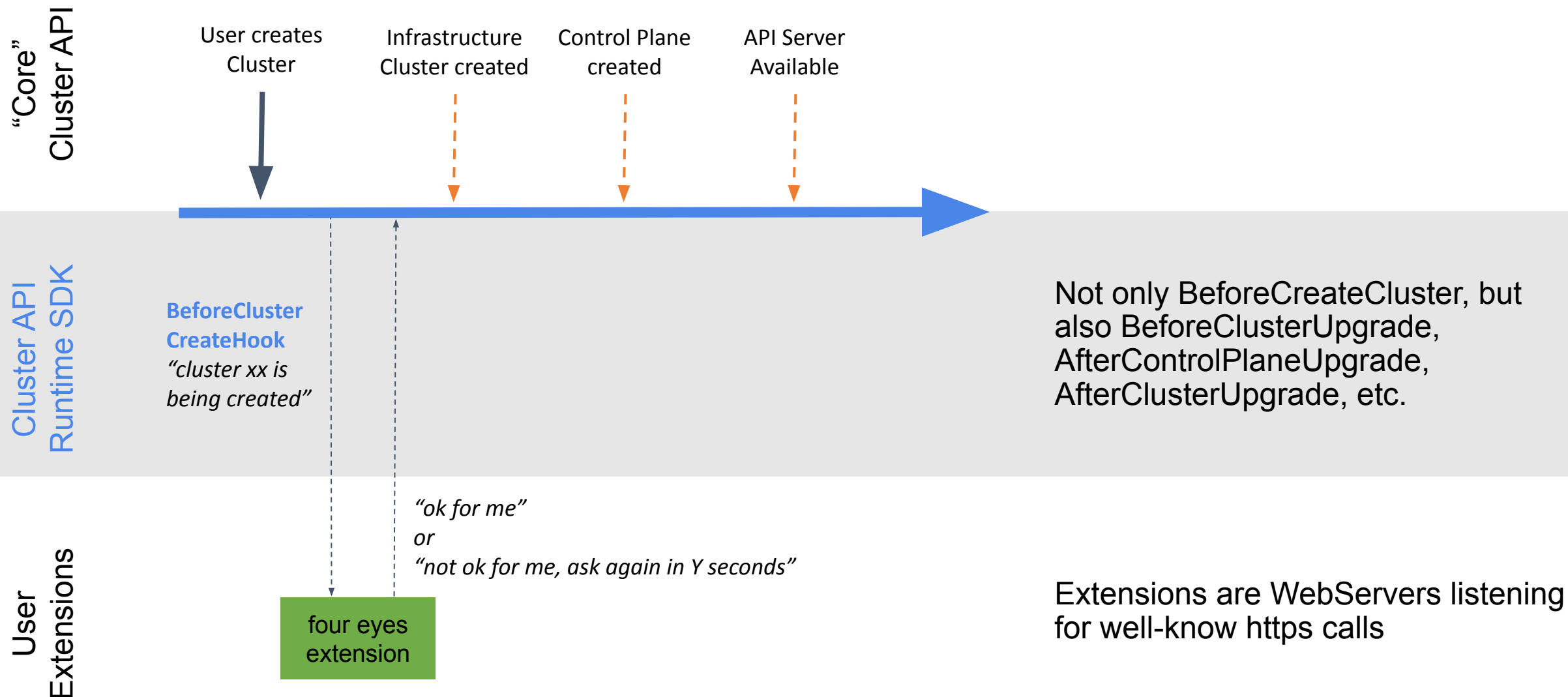
/help

Allow external components to hook in the Cluster Lifecycle

Let's explain this with an example

- Problem:
 - A company requires four eyes approval before creating a new Cluster, but this is not supported by core Cluster API (and it will never be).
- Solution:
 - With Runtime SDK you can develop a component that checks for four eyes approval, and then deploy and register it as a Cluster API runtime extension.
 - Before creating a Cluster, Cluster API will call registered extensions; Creation will continue only when the component returns a success message.

Extensibility with Runtime SDK



And there is much more ...

More and more innovation is driven by Cluster API users !

- Support for IPAM providers
- Autoscaler for Cluster API can now scale up from 0
- [Cluster API Visualizer](#) project allows to dig into Cluster resources
- Supports structured logging, searching and filtering logs now is much easier
- kube-state-metrics for Cluster API resources are now available (tech preview)
- Many ongoing improvements for Managed Cluster providers (AKS, EKS)
- Proposal for Cluster API addon provider for Helm as well as improvements in ClusterResourceSet
- Proposal for managing Node labels and streamline label propagation



BUILDING FOR THE ROAD AHEAD

DETROIT 2022

back to the future. our roadmap

**Cluster API adoption is continuously growing
and we should work to provide higher
guarantees of stability for our APIs**

- Before getting to GA there will be at least one more beta release addressing
 - Refs “GitOps” & upgrade friendly
 - Taint management
 - A queue of small improvements
 - A general audit/check on latest API guidelines

/help

We should identify a way forward for add-ons that are essential to the proper function of the Cluster

This is a complex and highly opinionated space:

- We had a first wave of DIY solutions (cluster-addon project, ClusterResourceSet in Cluster API)
- We are now exploring “pluggable” solutions relying on existing package management tools (e.g Helm, kapp, git-ops tools etc.)
- There are other solutions in the ecosystem (e.g. HelmRelease in Flux)

/help

How can we leverage the same declarative approach to manage the lifecycle of managed Kubernetes service clusters?

- Cluster API was originally designed with unmanaged Kubernetes in mind
- Since v1alpha1, major cloud providers have released managed Kubernetes services
- Users want the benefits of a managed k8s service while also leveraging the Cluster API goodness
- CAPZ and CAPA have added support for AKS and EKS, GKE support in CAPG is TODO

Today Kubernetes is used everywhere. How can we make it better for people running it outside datacenters?

First crawl, then walk, finally run:

- Refine support for single node Clusters
- Make our tooling resilient to slow/temporarily disconnected networks
- Investigate current footprint and tune for resource limited environments
 - We can be creative! e.g. In Cluster API
 - bootstrap without kind, management cluster in a Pod
 - faster Machine bootstrap & rollout
 - A/B rollouts at OS level
 - etc.

/help

how to get more info

SIG Cluster lifecycle sessions

Wednesday, October 26

2:30pm EDT

- [Tutorial: So You Want To Develop a Cluster API Provider? - Anusha Hegde, Nirmata; Winnie Kwon, VMware; Richard Case, Weaveworks; Avishay Traeger, Red Hat](#)
- [Simplified Experience Of Building Cluster API Provider In Multitenant Cloud - Sahithi Ayloo & Arun Krishnakumar, VMware](#)

3:25pm EDT

- [Running Isolated VirtualClusters With Kata & Cluster API - Chris Hein & Eric Ernst, Apple, Inc](#)

Friday, October 28

11:00am EDT

- [How Adobe Planned For Scale With Argo CD, Cluster API, And VCluster - Joseph Sandoval, Adobe & Dan Garfield, Codefresh](#)

11:55am EDT



- [Bare-Metal Chronicles: Intertwinement Of Tinkerbelle, Cluster API And GitOps - Katie Gamanji, Apple](#)

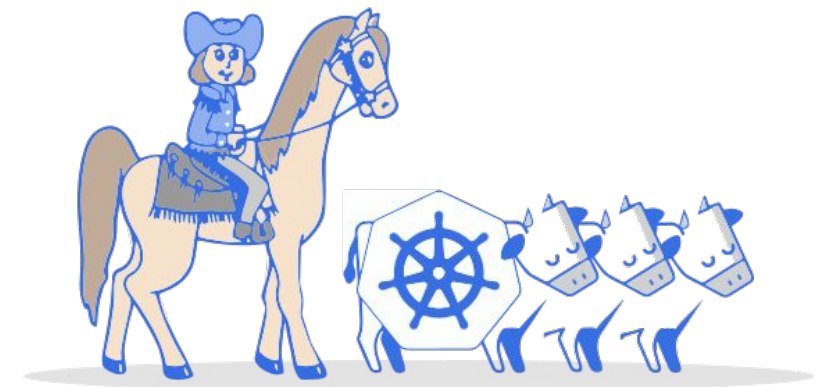
2:00pm EDT

- [Tutorial: kubectl Create Cluster: Production-ready Kubernetes with Cluster API 1.0 - Killian Muldoon, Shivani Singhal, Yuvaraj Balaji Rao Kakaraparthi & Stefan Büringer, VMware; Jack Francis, Microsoft](#)

how you can get involved

How to contribute

- Checkout our [contributing guide](#) and our [community page](#)
- Attend office hours on Zoom, ask questions
- Introduce yourself on Slack, join project channels
- Look for "good-first-issue" and "help-wanted" label on GitHub issues
 - Docs and testing  
 - Developer tooling, release automation
- Help with the release team!
- Chop wood, carry water, be kind



Q&A



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