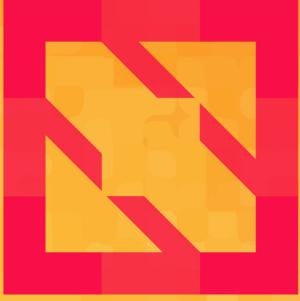




KubeCon



CloudNativeCon

North America 2019





KubeCon



CloudNativeCon

North America 2019

Going beyond the node..

Virtual Kubelet Deep Dive

Brian Goff @cpuguy83

Deep Kapur @deepkkapur



Why are you here

- Learn about Virtual Kubelet
- Why is it useful / how can it be used
- Contribute to the project

What is a Kubelet?



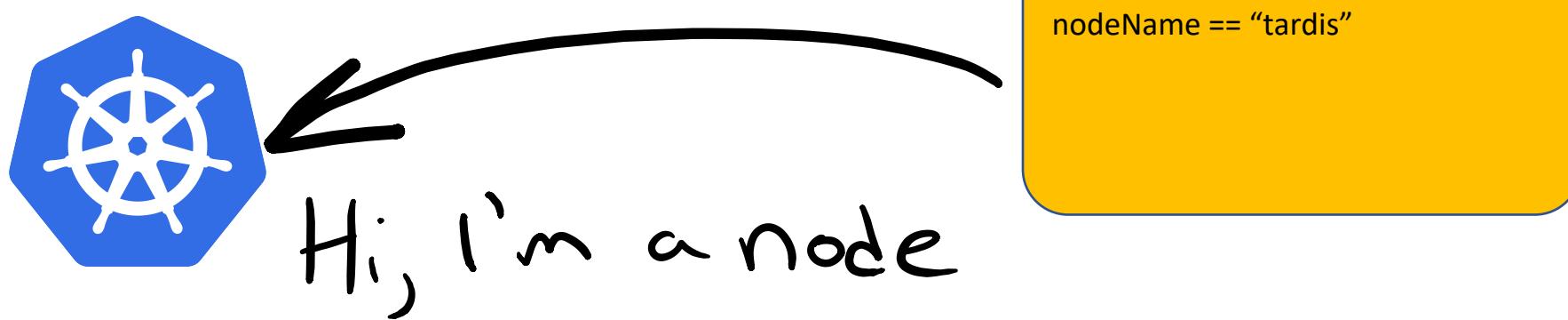
kubelet

kube•let

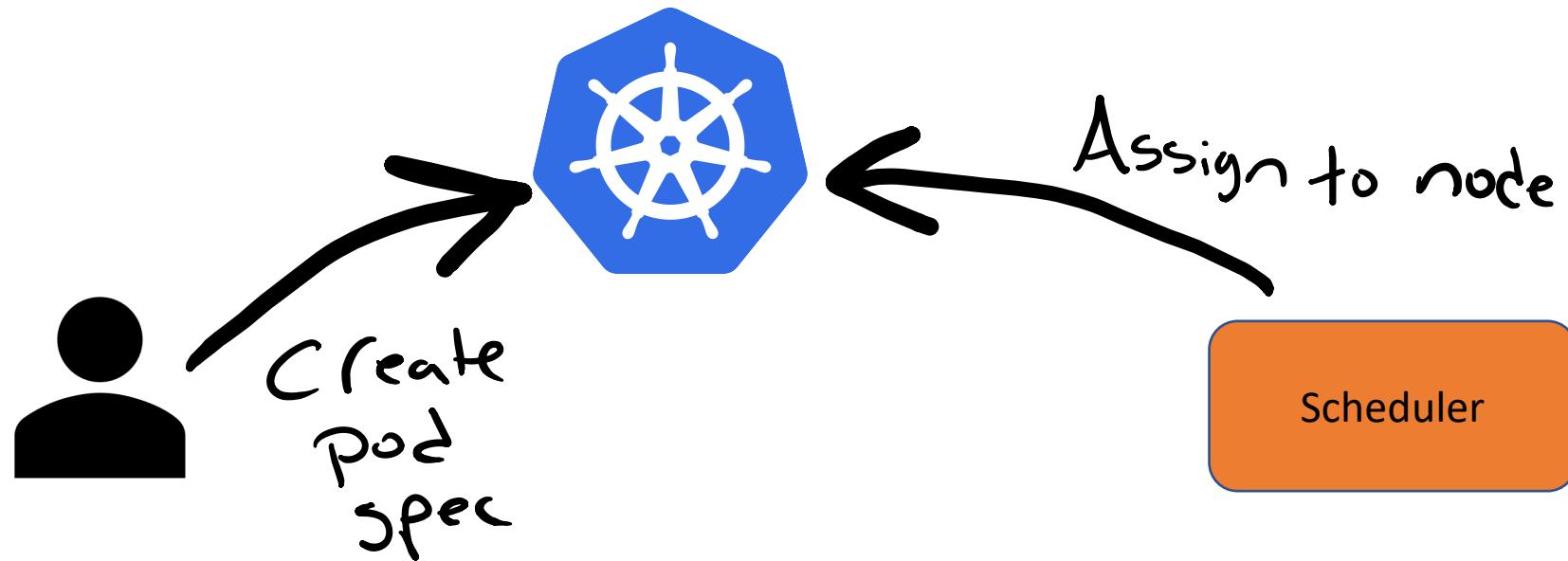
noun

An agent that runs on **each node** in the cluster. It makes sure that **containers are running** in a pod.

What is a Kubelet?



What is a Kubelet?



What is a Kubelet?

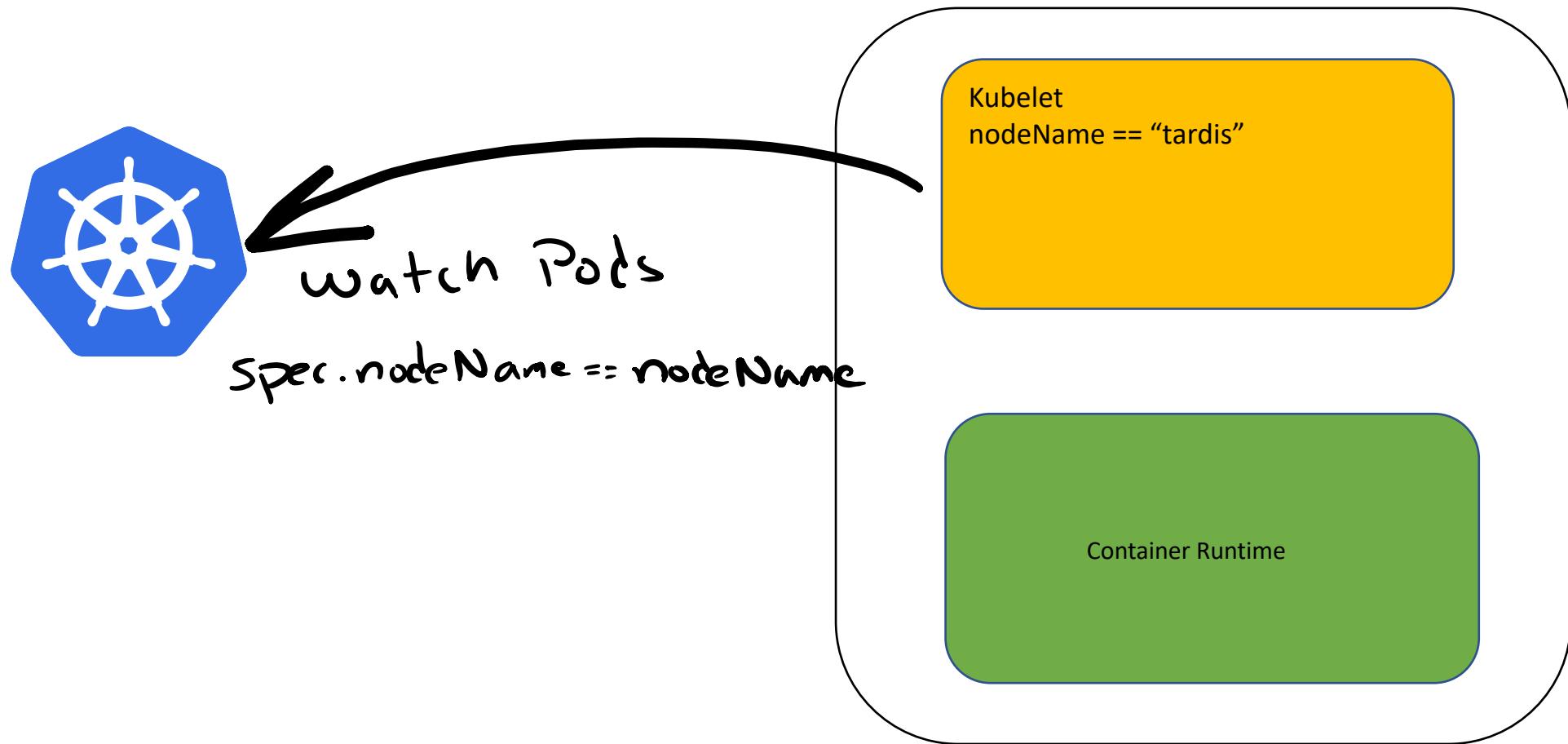


KubeCon



CloudNativeCon

North America 2019



What is a Kubelet?

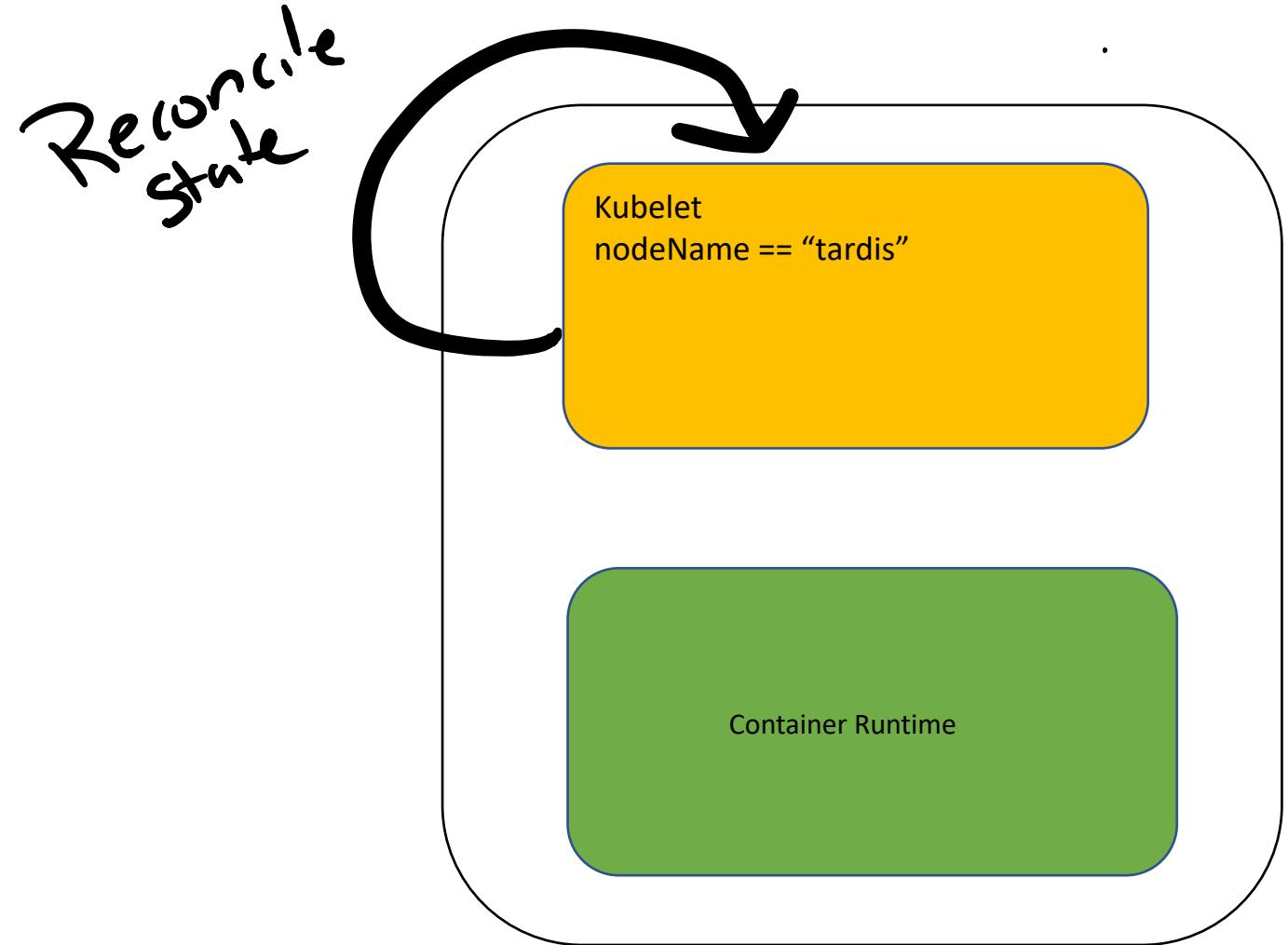


KubeCon



CloudNativeCon

North America 2019



What is a Kubelet?



KubeCon

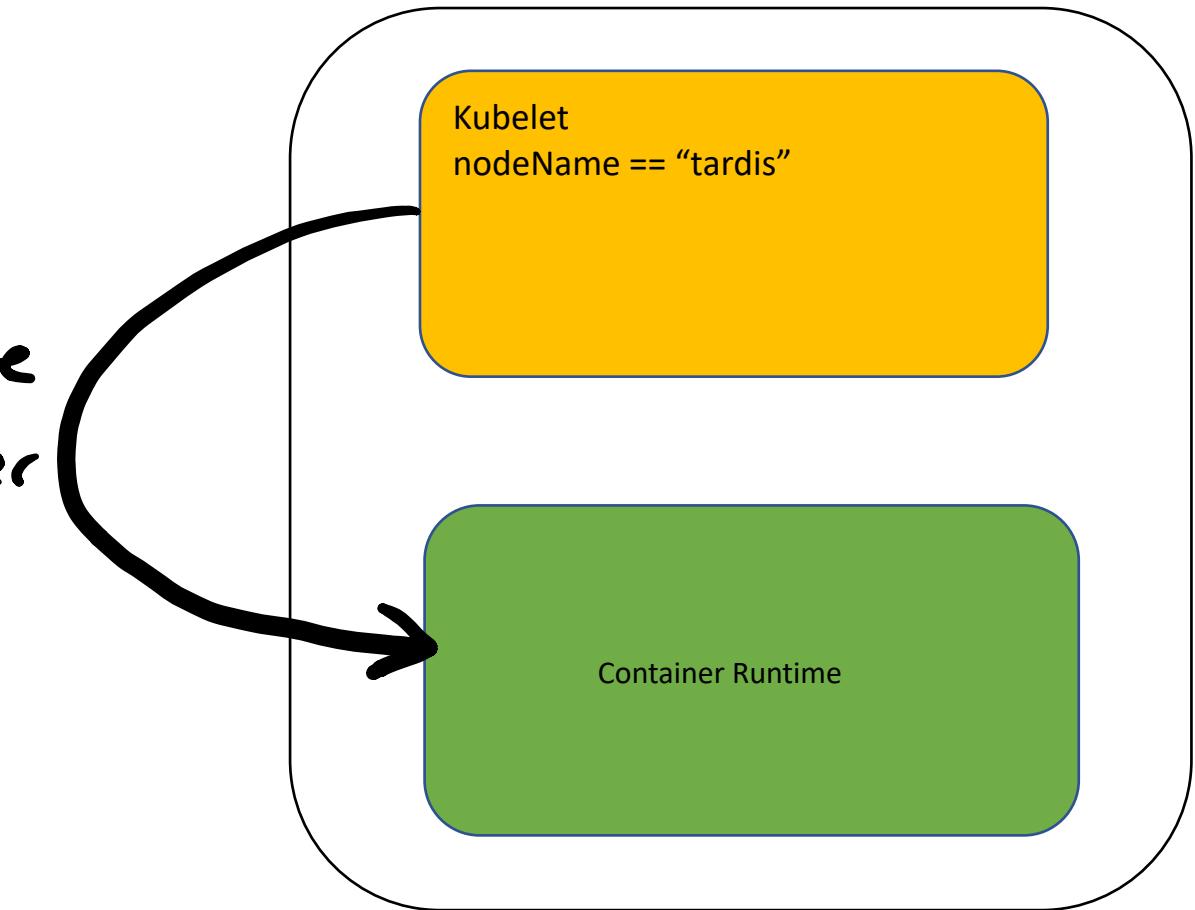


CloudNativeCon

North America 2019



Create
Container



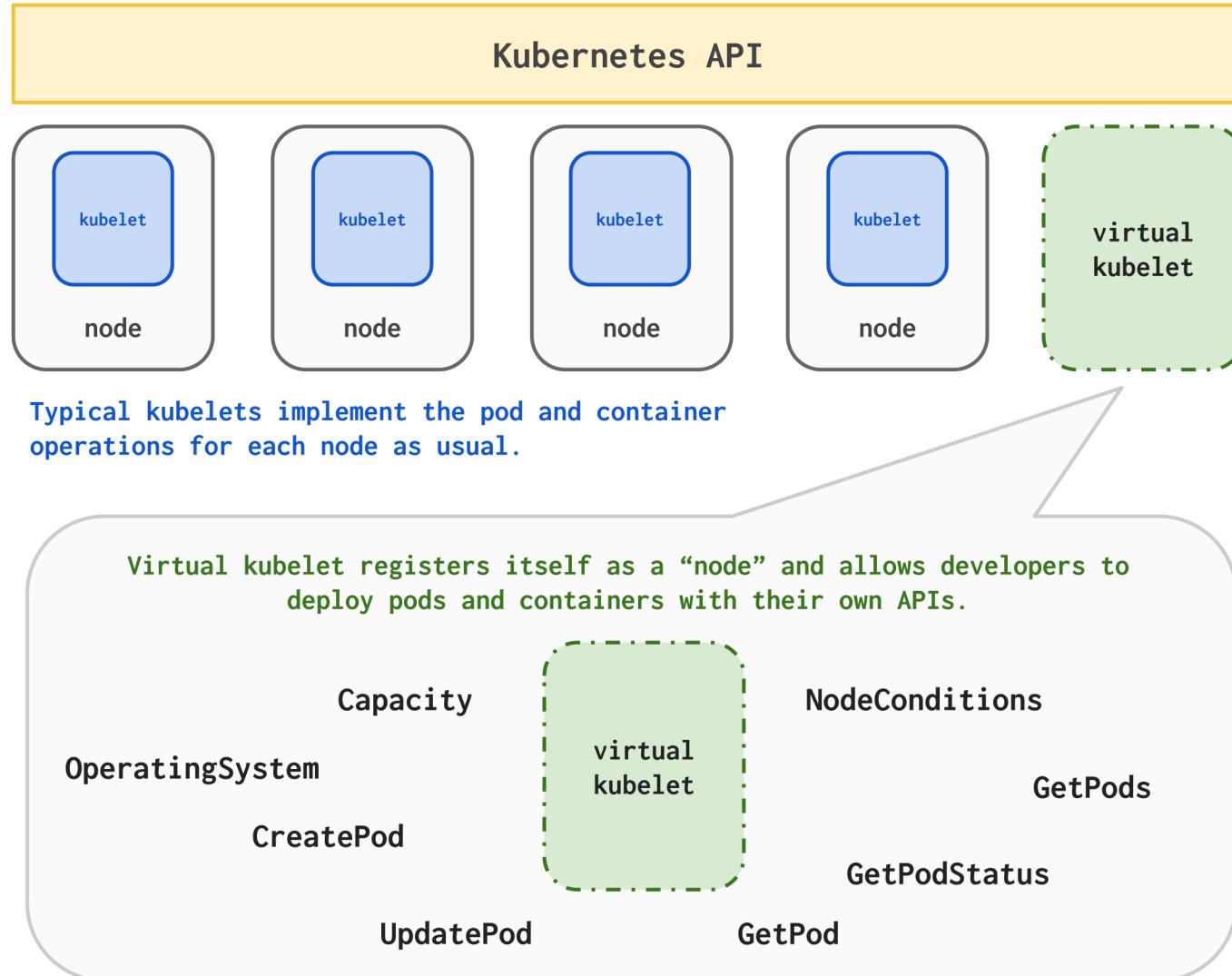
What is Virtual Kubelet?



KubeCon

CloudNativeCon

North America 2019



How it works



KubeCon



CloudNativeCon

North America 2019



How it works



KubeCon

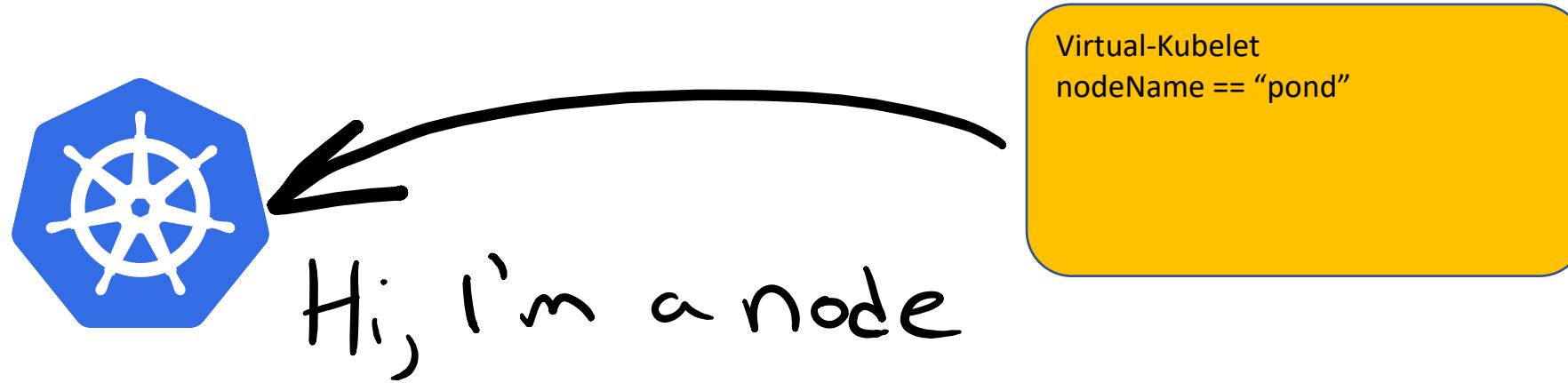


CloudNativeCon

North America 2019



How it actually works



How it actually works

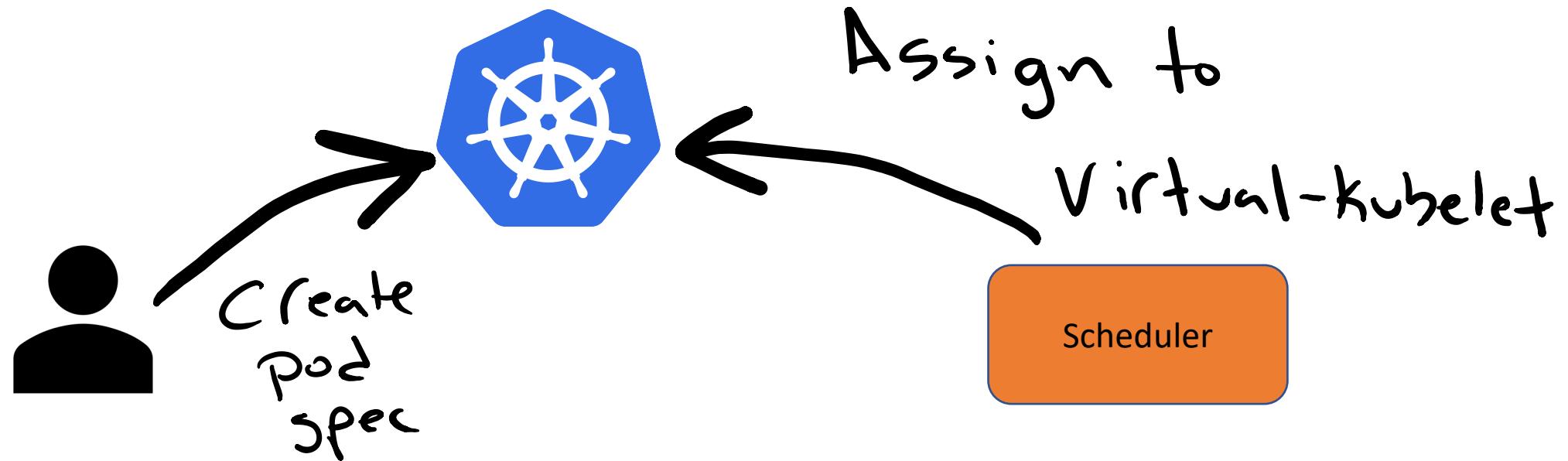


KubeCon



CloudNativeCon

North America 2019



How it actually works



KubeCon



CloudNativeCon

North America 2019



How it actually works



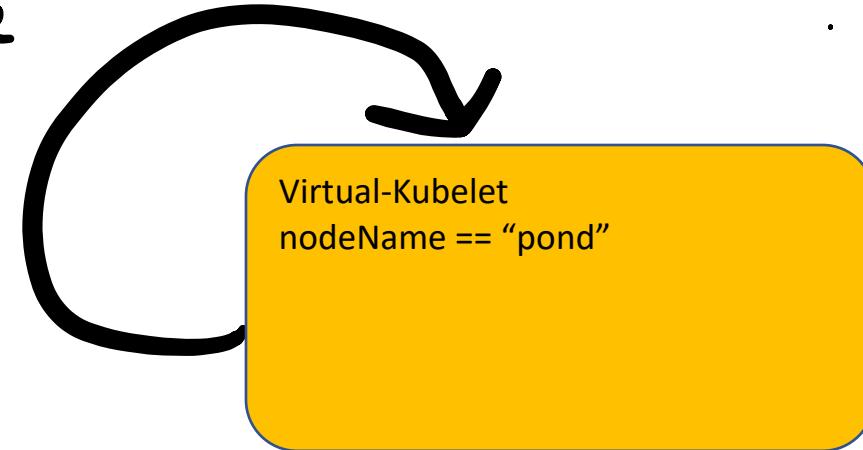
KubeCon

CloudNativeCon

North America 2019



Reconcile
State



External Service

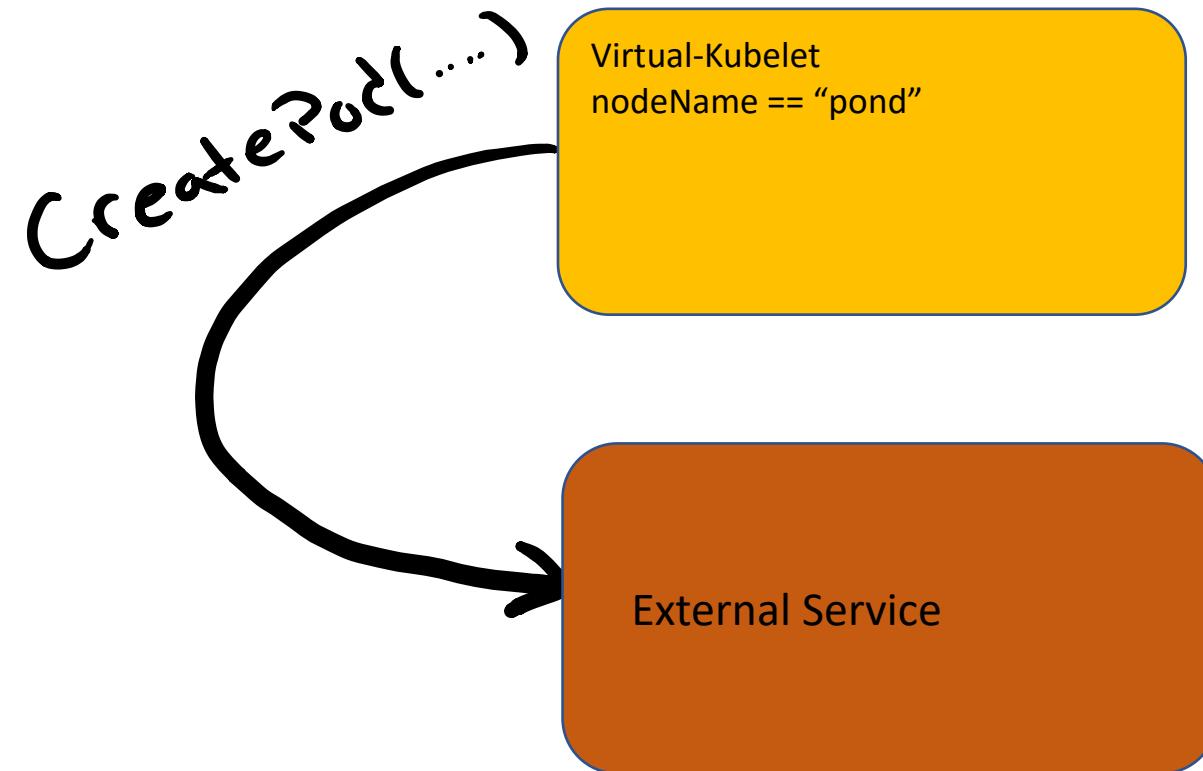
How it actually works



KubeCon

CloudNativeCon

North America 2019



VK Project

- GitHub: <https://github.com/virtual-kubelet/virtual-kubelet>
- 5 core maintainers
 - Netflix, Microsoft, VMWare
- Latest release - v1.2.1 (5 days ago!)
- 6 providers in the virtual-kubelet GitHub organization, 3+ additional
- Currently a Sandbox project in CNCF
 - Working towards Incubation stage

Since last we met!

- 1.0 released on July 9th, 2019
 - Stabilized provider interfaces
 - Moved all providers out of tree, all now consume VK as a library
- 1.1 released on September 18, 2019
 - Mostly bug fixes
 - Minor version bump due to changes to underlying Kubernetes libs
- 1.2 released on November 6, 2019
 - Support for graceful pod deletion

Anatomy of Virtual Kubelet

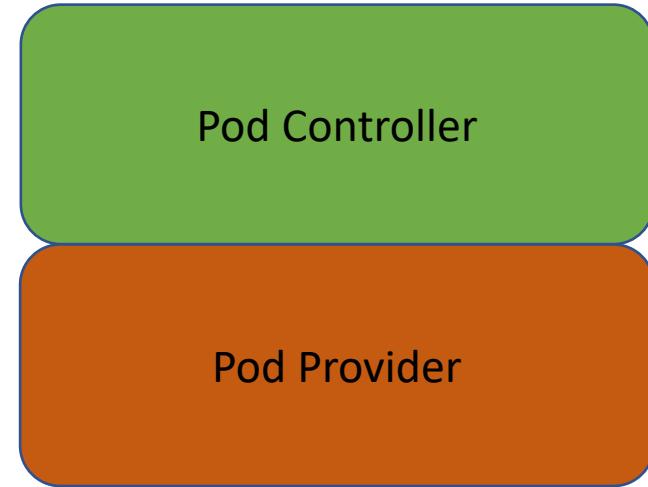
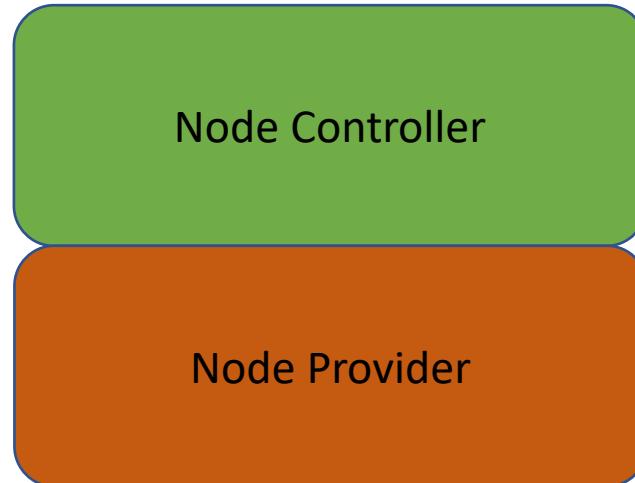


KubeCon



CloudNativeCon

North America 2019



Anatomy of Virtual Kubelet



Node Controller

Registers and ensures node is active in Kubernetes by providing status updates and node leases.

Node Provider

Notifies the node controller when there is a status change for the node.

Anatomy of Virtual Kubelet



Pod Controller

Sets up watches for pods scheduled to the specified node. When pods are added, updated, deleted in Kubernetes, calls the appropriate handler on the pod provider.

Also populates pod specs from the downward API.

Pod Provider

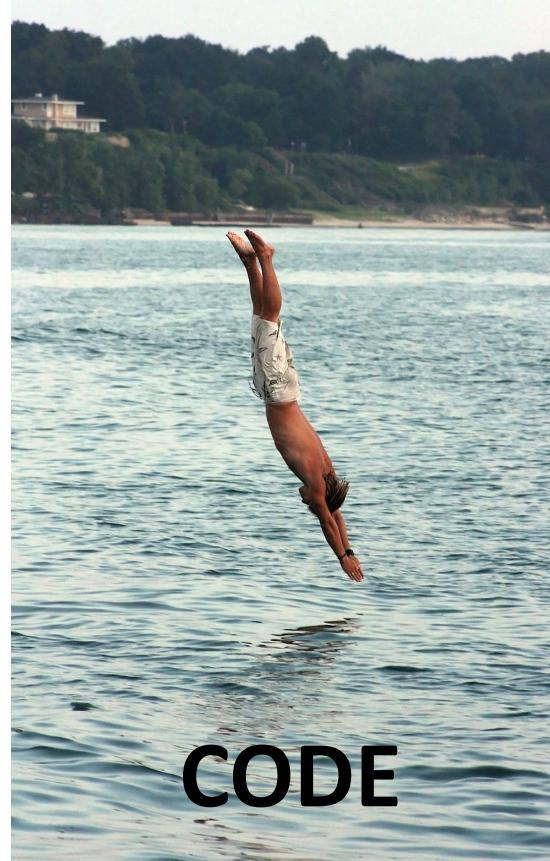
Provide the back-end plumbing necessary to support the lifecycle management of pods, containers and supporting resources in the context of Kubernetes.

This is your code.

Anatomy of Virtual Kubelet



What it takes to build a VK instance



Provider code



KubeCon



CloudNativeCon

North America 2019

type PodLifecycleHandler

```
type PodLifecycleHandler interface {
    // CreatePod takes a Kubernetes Pod and deploys it within the provider.
    CreatePod(ctx context.Context, pod *corev1.Pod) error

    // UpdatePod takes a Kubernetes Pod and updates it within the provider.
    UpdatePod(ctx context.Context, pod *corev1.Pod) error

    // DeletePod takes a Kubernetes Pod and deletes it from the provider. Once a pod is
    // expected to call the NotifyPods callback with a terminal pod status where all the
    // state, as well as the pod. DeletePod may be called multiple times for the same pod.
    DeletePod(ctx context.Context, pod *corev1.Pod) error

    // GetPod retrieves a pod by name from the provider (can be cached).
    // The Pod returned is expected to be immutable, and may be accessed
    // concurrently outside of the calling goroutine. Therefore it is recommended
    // to return a version after DeepCopy.
    GetPod(ctx context.Context, namespace, name string) (*corev1.Pod, error)

    // GetPodStatus retrieves the status of a pod by name from the provider.
    // The PodStatus returned is expected to be immutable, and may be accessed
    // concurrently outside of the calling goroutine. Therefore it is recommended
    // to return a version after DeepCopy.
    GetPodStatus(ctx context.Context, namespace, name string) (*corev1.PodStatus, error)

    // GetPods retrieves a list of all pods running on the provider (can be cached).
    // The Pods returned are expected to be immutable, and may be accessed
    // concurrently outside of the calling goroutine. Therefore it is recommended
    // to return a version after DeepCopy.
    GetPods(context.Context) ([]*corev1.Pod, error)
}
```

VK Provider definition



KubeCon



CloudNativeCon

North America 2019

- Provide the plumbing to support pod lifecycle and supporting resources in a different infrastructure
 - Transform the declarative API model to an imperative one
- Conform to current VK API
- Use a well-defined callback mechanism for getting data like secrets and configmaps without direct access to K8s API server

Challenges in building a provider



- Networking! Network configuration, private networking, kube-proxy...
- What is the max node resource?
- VK versioning
- Monitoring pods spun out via VK (supporting HPA)

Demo



KubeCon



CloudNativeCon

North America 2019

Why build a provider?

- Flexible abstraction to extend a cluster
- More granular resource consumption = unit consumption is a pod
- Hybrid clusters – traditional cluster with a cloud based VK
- High availability deployments
- Alternative to Kubelet – VK as a node agent in a different kind of node

Current Providers

- Alibaba ECI
- AWS Fargate
- Azure ACI, IoT Edge, Batch
- CRI-containerd
- HashiCorp Nomad
- Huawei CCI
- OpenStack Zun



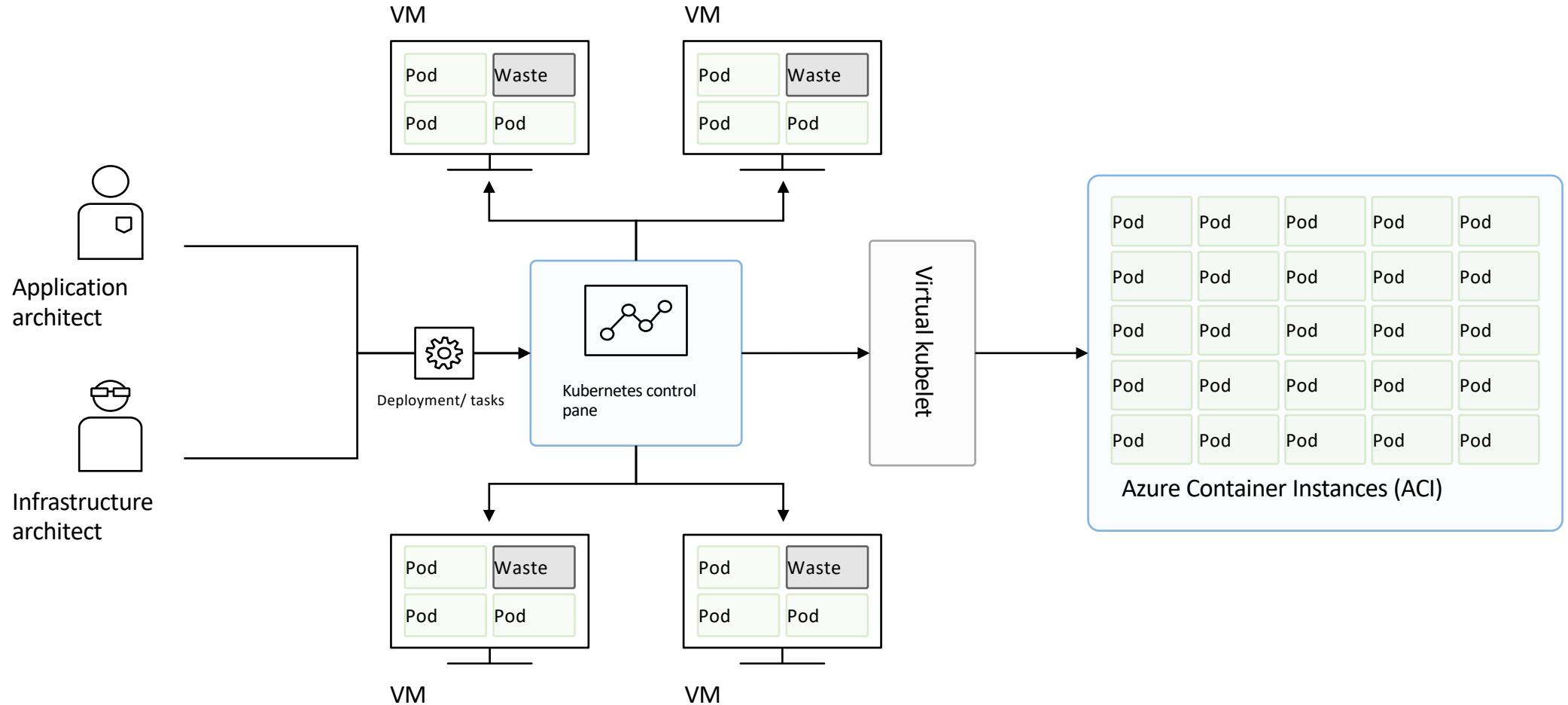
HashiCorp



HUAWEI



Demo



The Road Ahead

- Custom Pod GC policies
- Metrics
- Helpers for building providers
- Runtime-less (maybe?)

Are we missing some functionality you'd like to see? Open an issue and/or bring it to the community sync meeting.

Join us!



<http://github.com/virtual-kubelet/virtual-kubelet>

Slack channel in Kubernetes #virtual-kubelet

VK Community meetings every other week on Wednesdays at 11am PST

