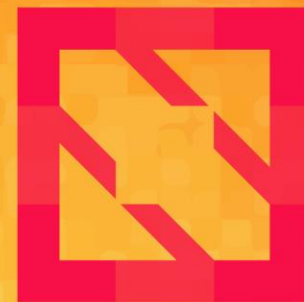




KubeCon



CloudNativeCon

North America 2019





KubeCon



CloudNativeCon

North America 2019

KubeEdge Deep Dive

Sean Wang <swang54@gmail.com>



Agenda



KubeCon



CloudNativeCon

North America 2019

- Product Decisions
- Key Components Discussion
- Application Scenarios Deep Dive
- Future Work Discussion

From Cloud to Edge

- **Low latency**
 - A solution is developed at the edge to reduce the latency.
- **Massive data**
 - An explosive growth in edge data; data migration to the cloud causes high costs.
 - Local data analysis and filtering saves network bandwidth.
- **Privacy security**
 - Sensitive, personal privacy data handled at edge, to protect production and business security.
- **Local autonomy**
 - Cloud-free offline processing and self-recovery



Cluster vs. Nodes at the Edge?

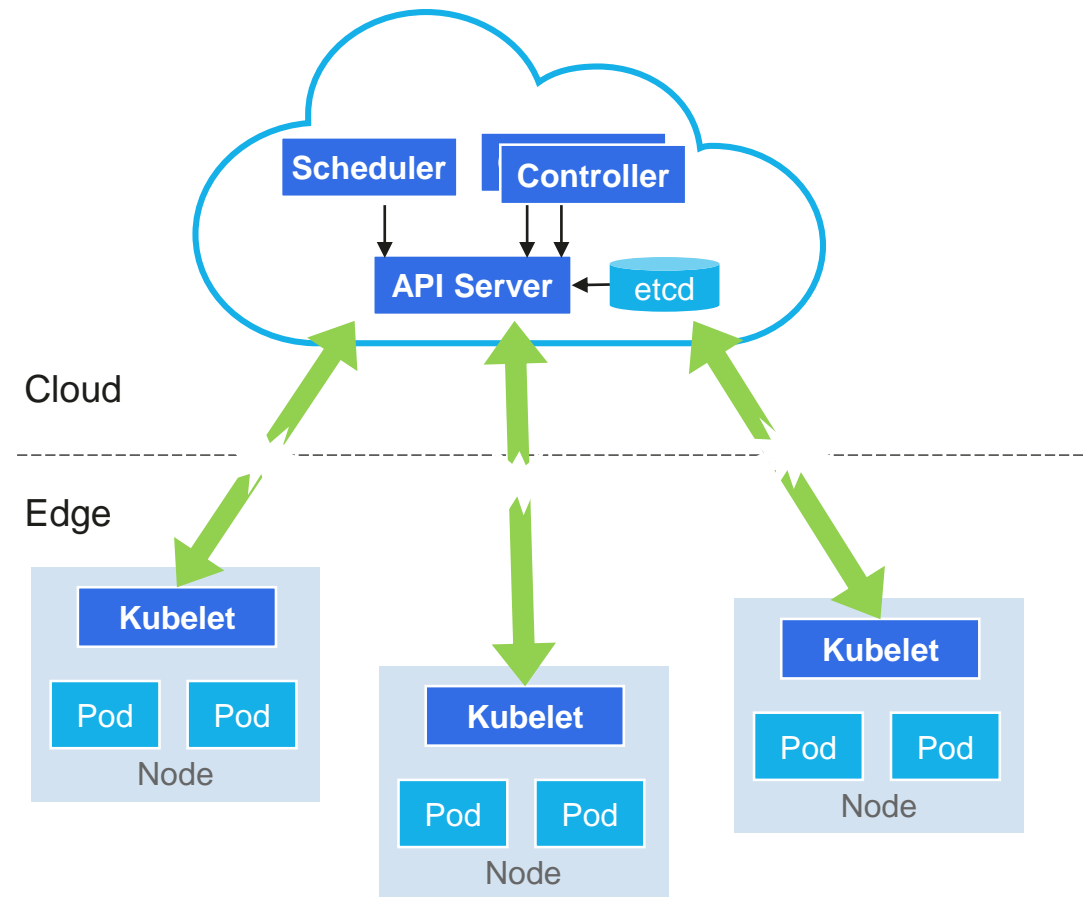
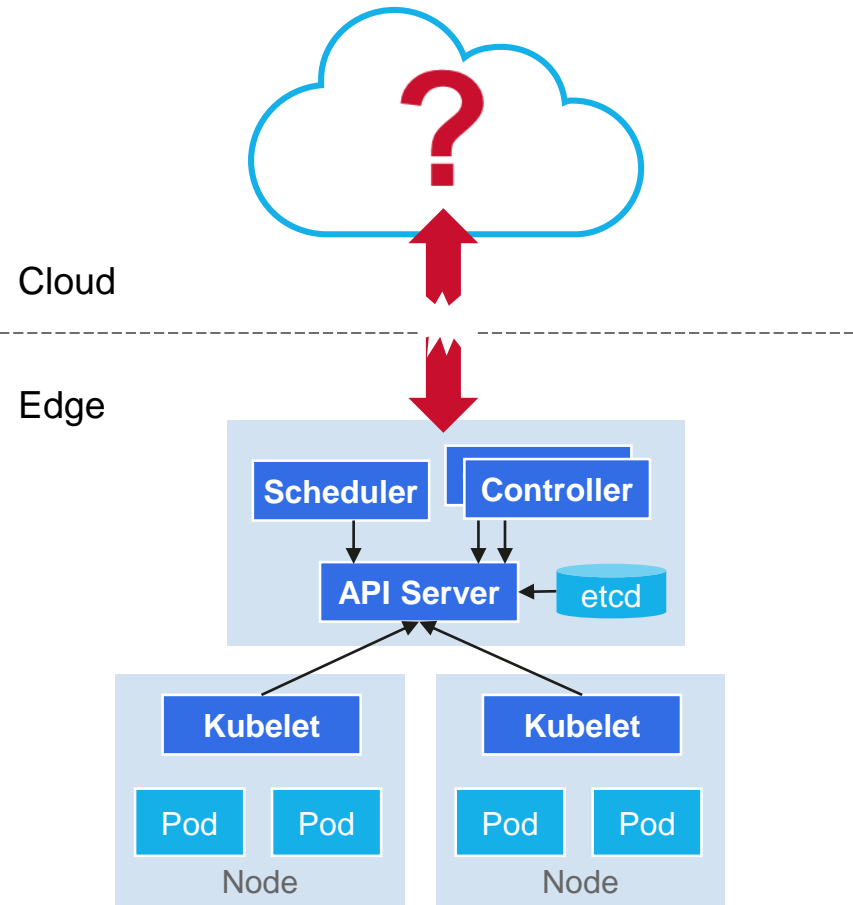


KubeCon



CloudNativeCon

North America 2019



What do Customers Want

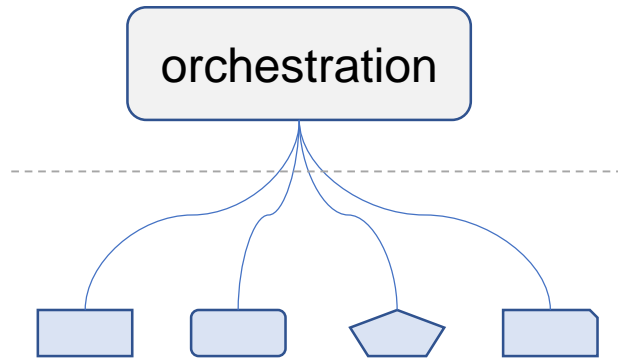


KubeCon

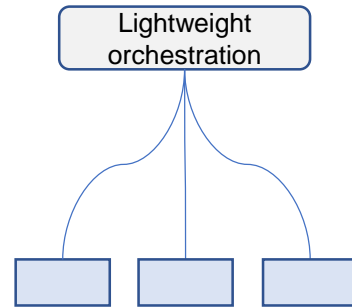


CloudNativeCon

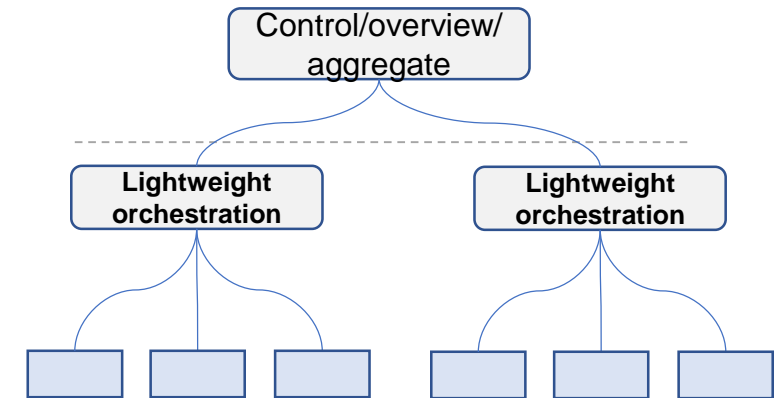
North America 2019



Heterogenous compute resources
from remote locations



Lightweight Kubernetes for a private
setup



Edge-cloud coordinated
orchestration

We can do more

KubeEdge Architecture

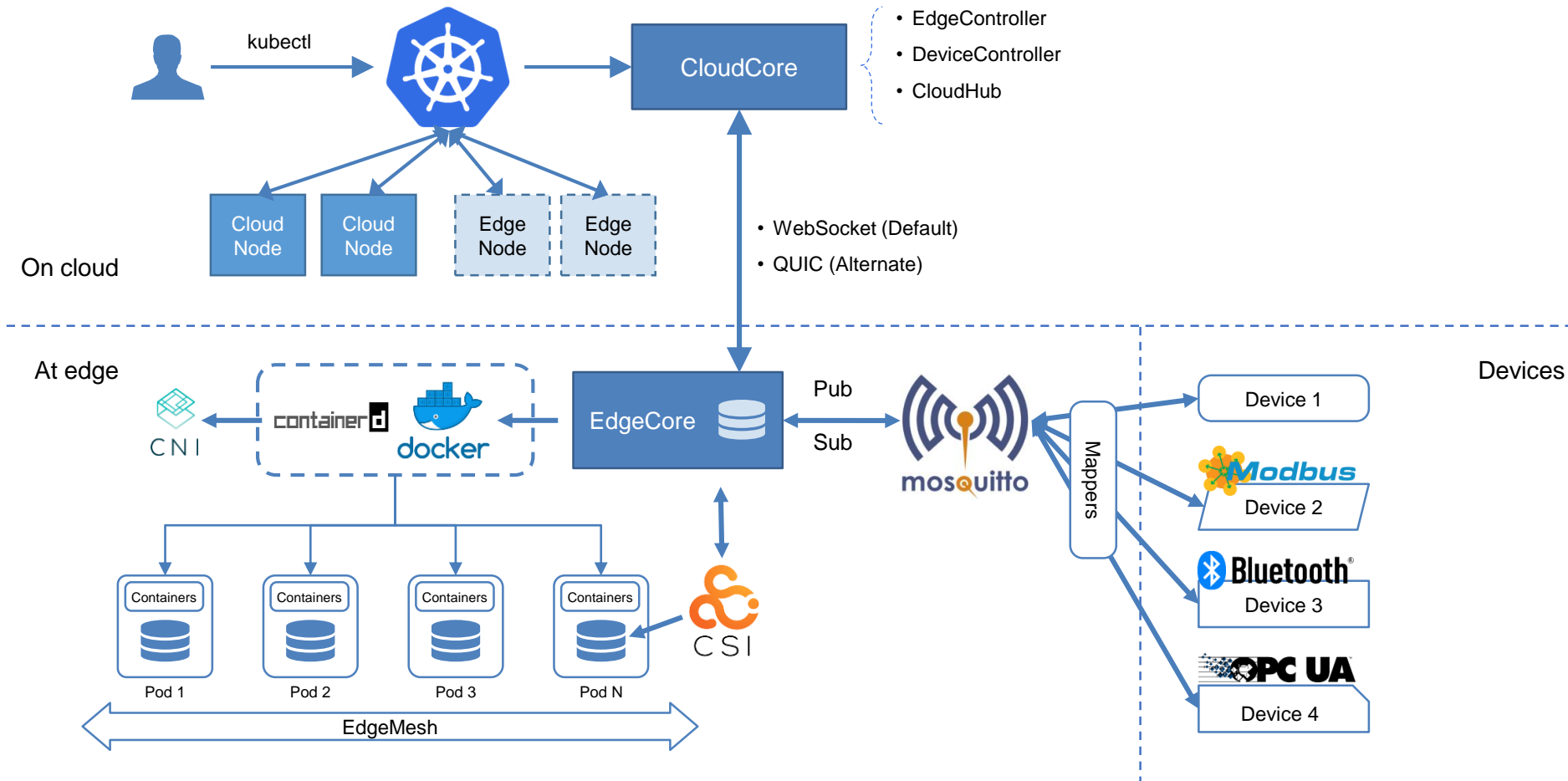


KubeCon



CloudNativeCon

North America 2019



Basic Frame

One of the design goals for KubeEdge is to build a modularized computing platform at edge, this applies to its own core component design as well.

Beehive is a messaging framework based on go-channels for communication between modules of KubeEdge.

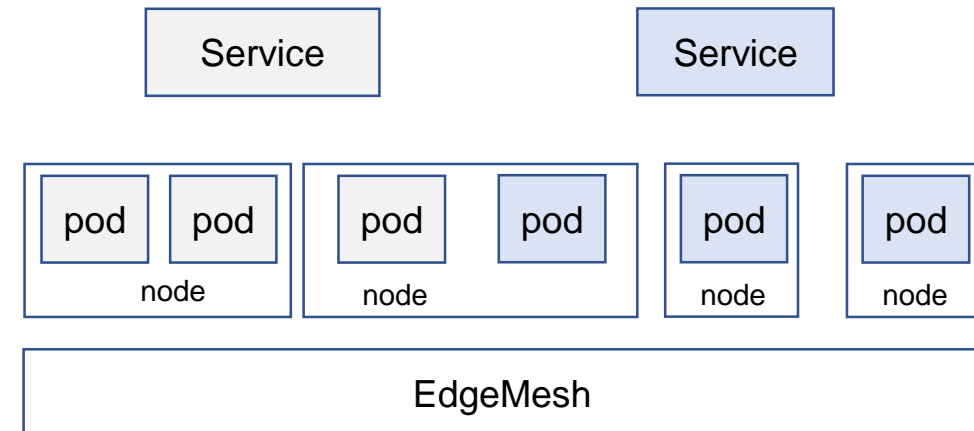
//each module need to implement the following interface

```
type Module interface {  
    Name() string  
    Group() string  
    Start(c *context.Context)  
    Cleanup()  
}
```

```
//Register the module to Beehive  
core.Register(& cloudHub{})
```

```
//Use Channel Context to commutate between modules and groups  
coreContext.Send("edged",message)  
msg, err := coreContext.Receive("edged")  
.....
```

EdgeMesh provides ServiceMesh at edge, enabling services running on different pods, nodes, locations to Mesh



KubeEdge is Bigger Than Kubelet



KubeCon



CloudNativeCon

North America 2019

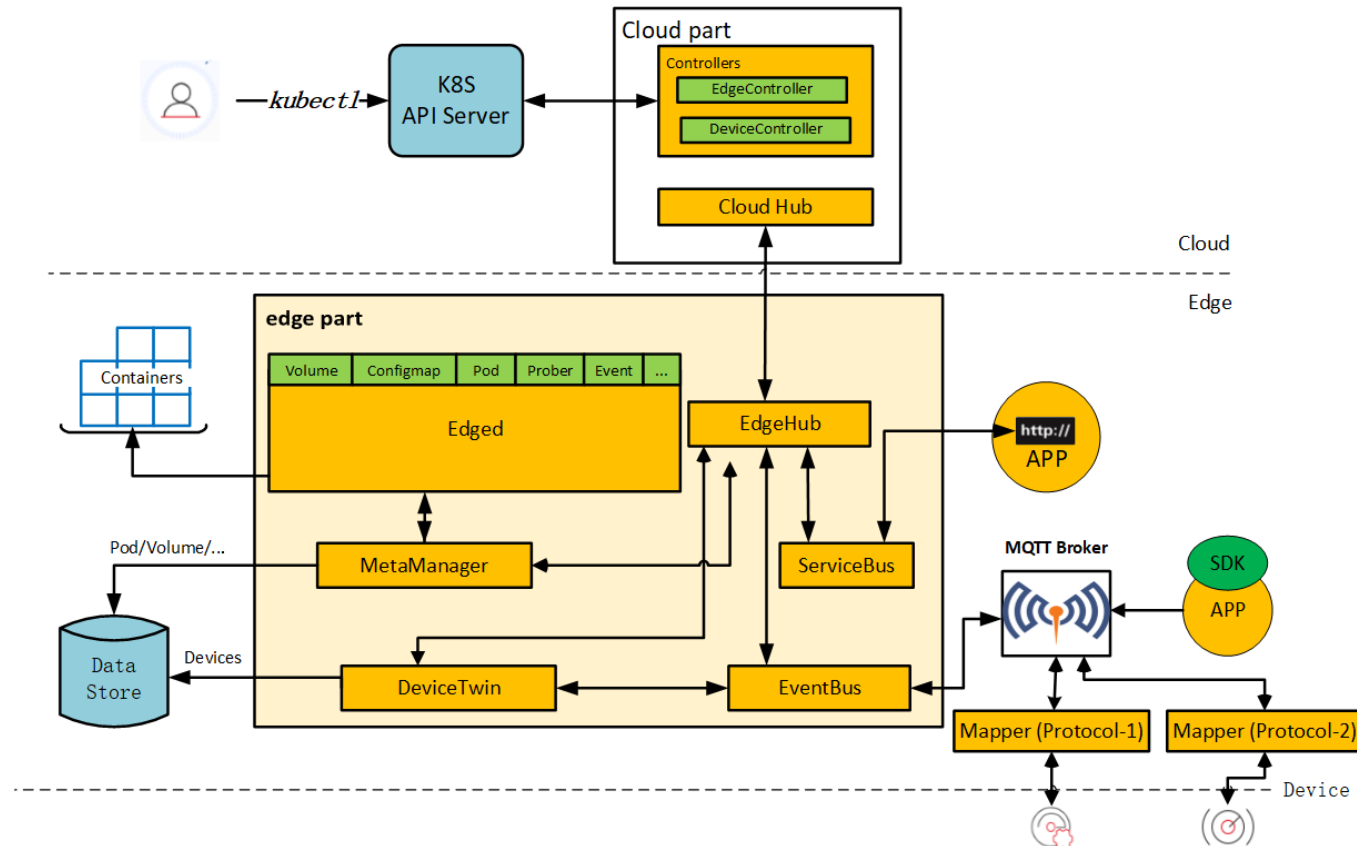
An extensible framework to fully open the compute power at edge

An Edge-Cloud channel not just for node control, but also for application

Enables node-cloud, node-node communications

Enabler for digital transformation of the physical world

Local persistent



CRI in KubeEdge

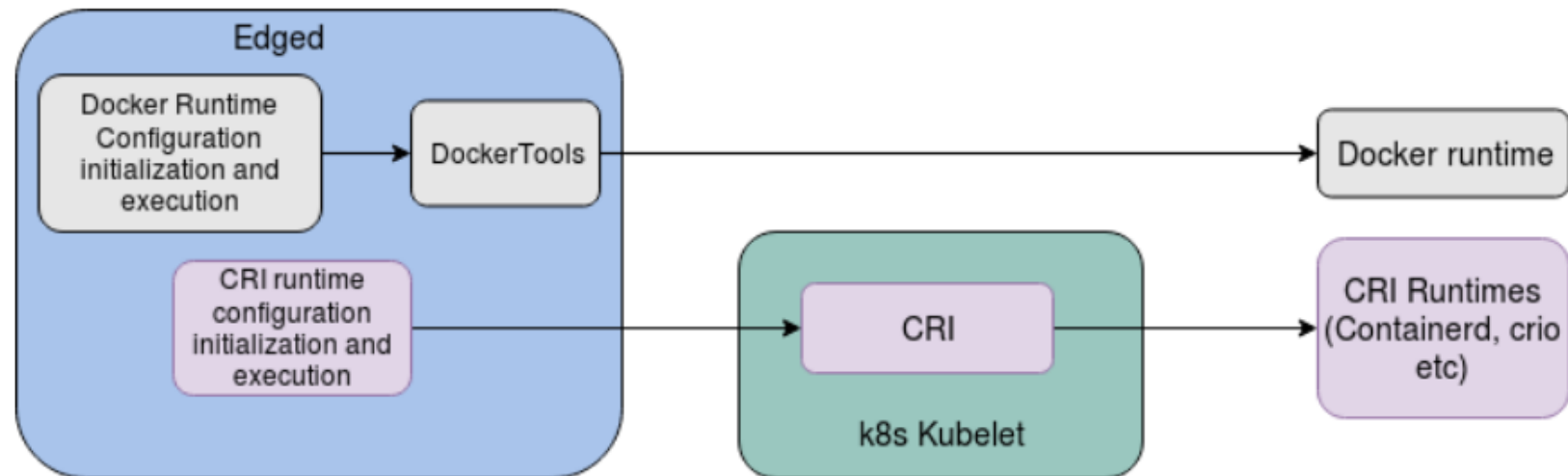


KubeCon



CloudNativeCon

North America 2019



Recommended CSI Drivers Deployment on K8s

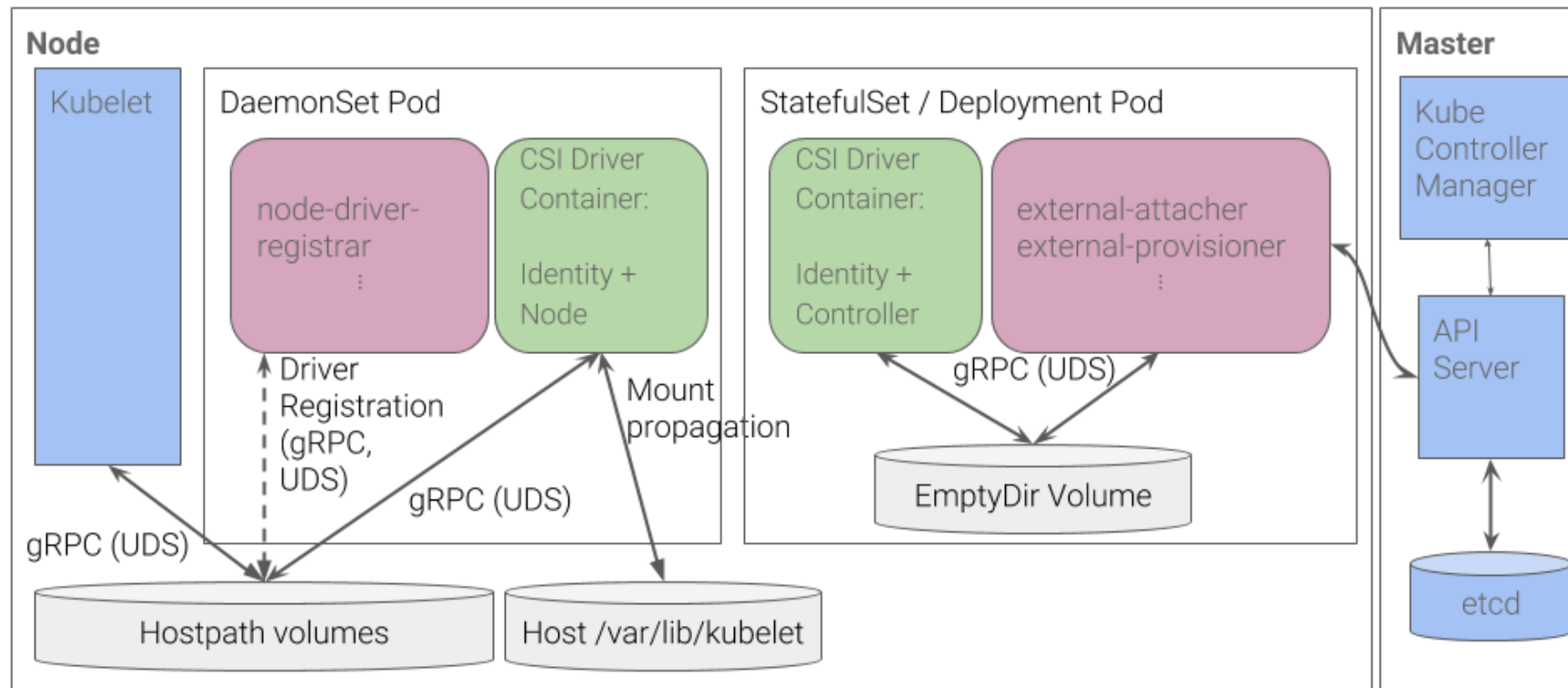


KubeCon



CloudNativeCon

North America 2019



CSI Integration on KubeEdge

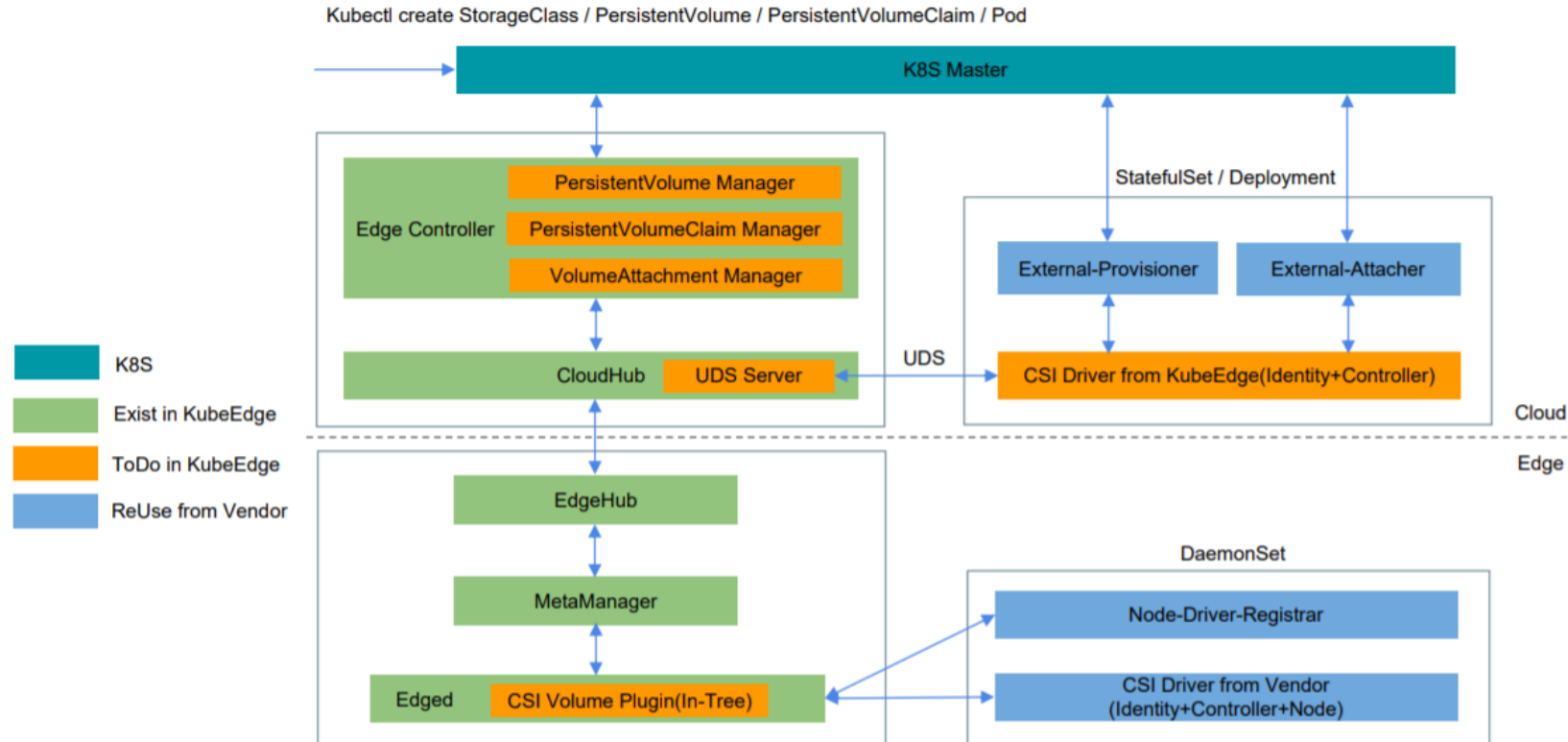


KubeCon



CloudNativeCon

North America 2019



Creating a CSI Volume

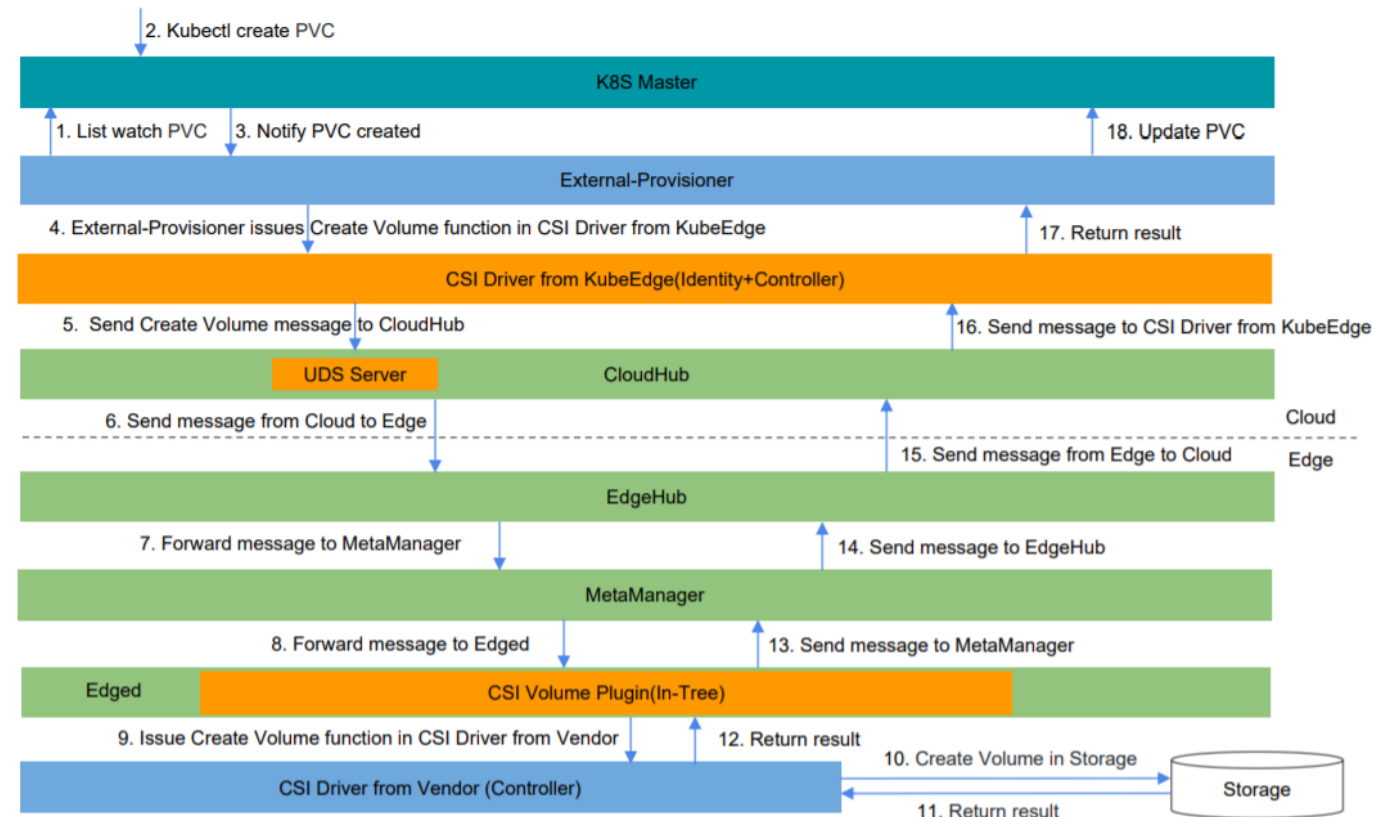


KubeCon



CloudNativeCon

North America 2019



Deleting a CSI Volume

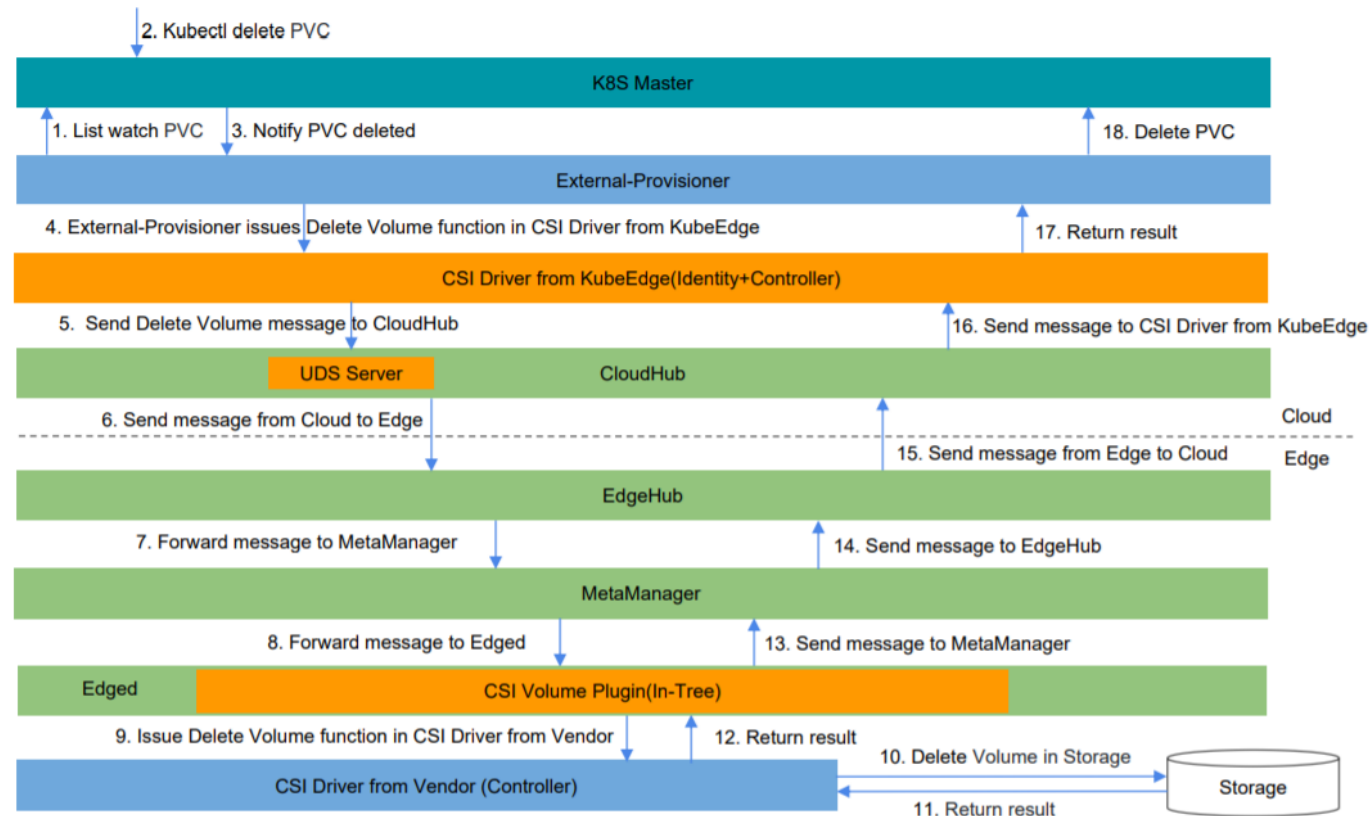


KubeCon



CloudNativeCon

North America 2019



Edge Device Management



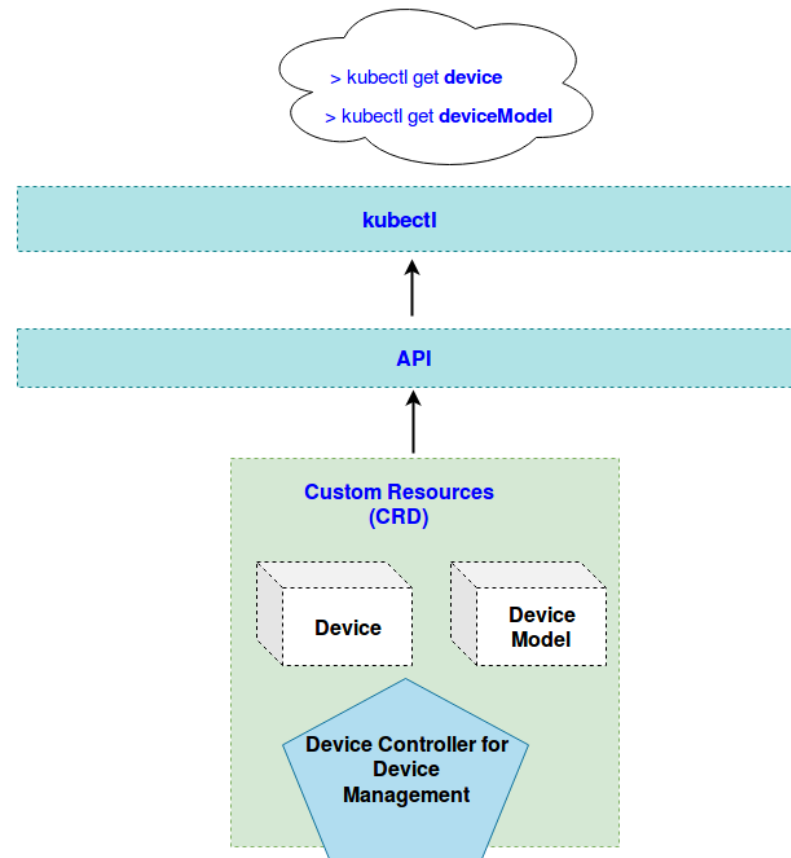
KubeCon



CloudNativeCon

North America 2019

- Edge Device APIs registered as K8s CRD
 - Same experience with K8s core APIs using kubectl
 - DeviceModel: template of devices
 - Defines common device properties including data type, read-only, default value, and max/min values, and communication protocols and arguments supported by each property.
 - DeviceInstance: instance of a device
 - Inherits properties from DeviceModel.
 - Obtains necessary arguments based on the protocol that used in reality.
 - Manages desired and reported states through DeviceTwin.



Edge Device Management



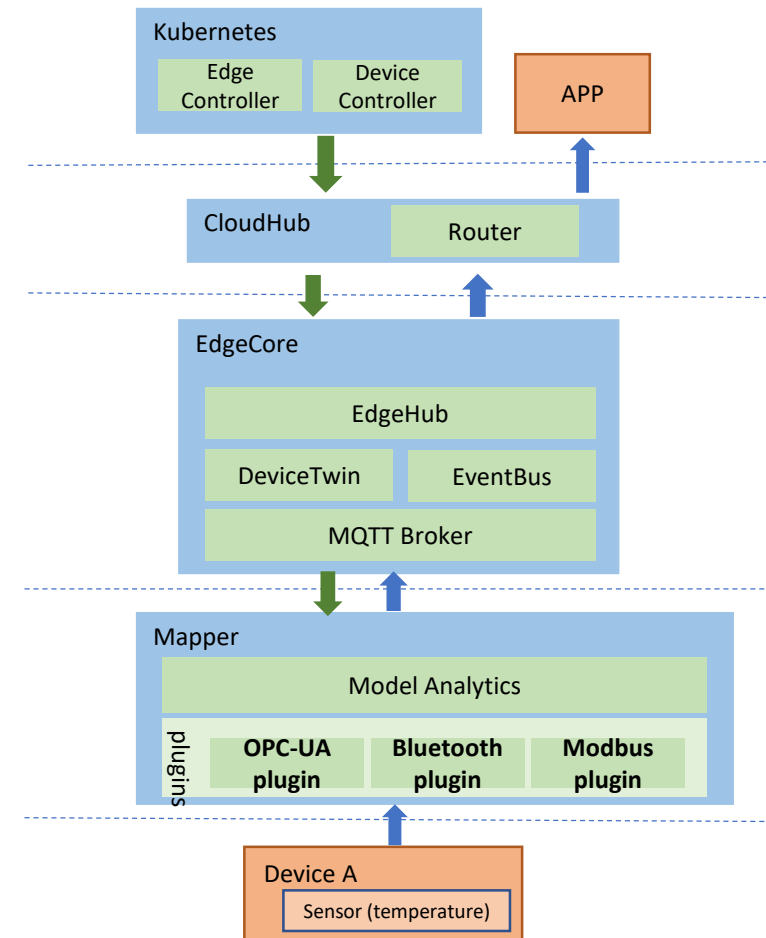
KubeCon



CloudNativeCon

North America 2019

- Pluggable Device Mapper framework
 - Easy to extend and customize
 - Mappers are managed by K8s DaemonSets, and easy to deploy, upgrade and roll back.



Setting Desired State from Cloud to Edge

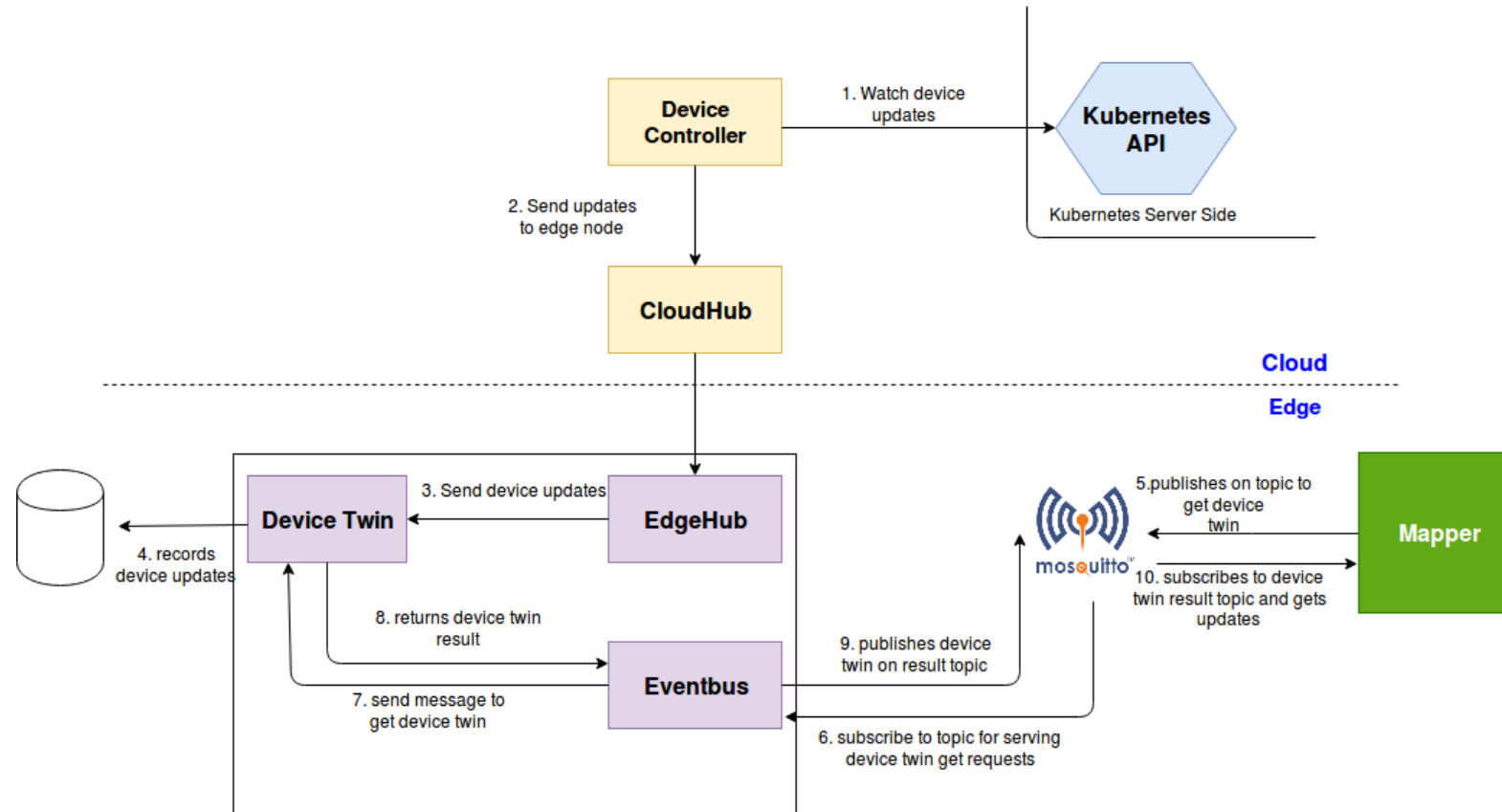


KubeCon



CloudNativeCon

North America 2019



Reporting Actual State from Edge to Cloud

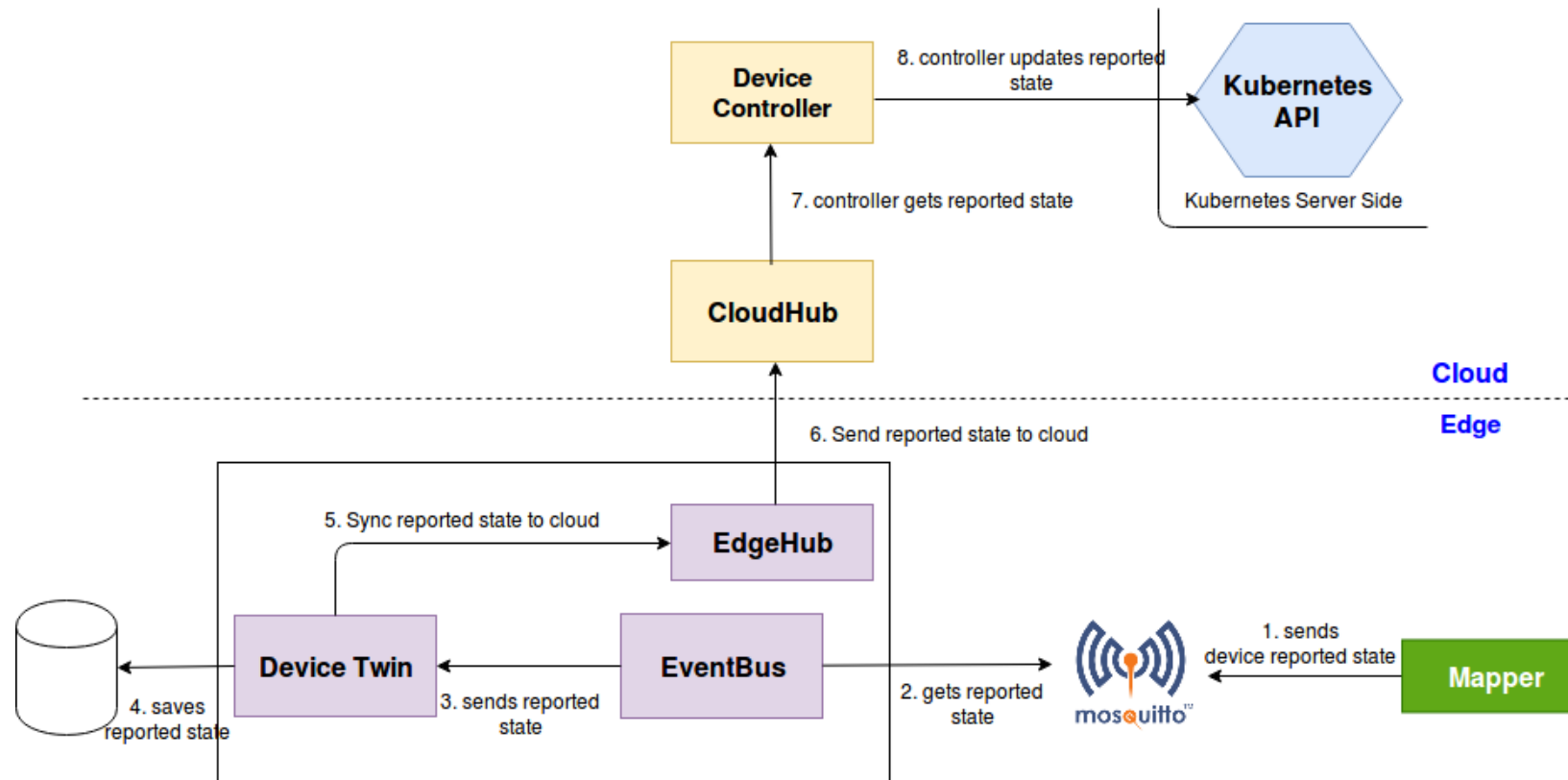


KubeCon



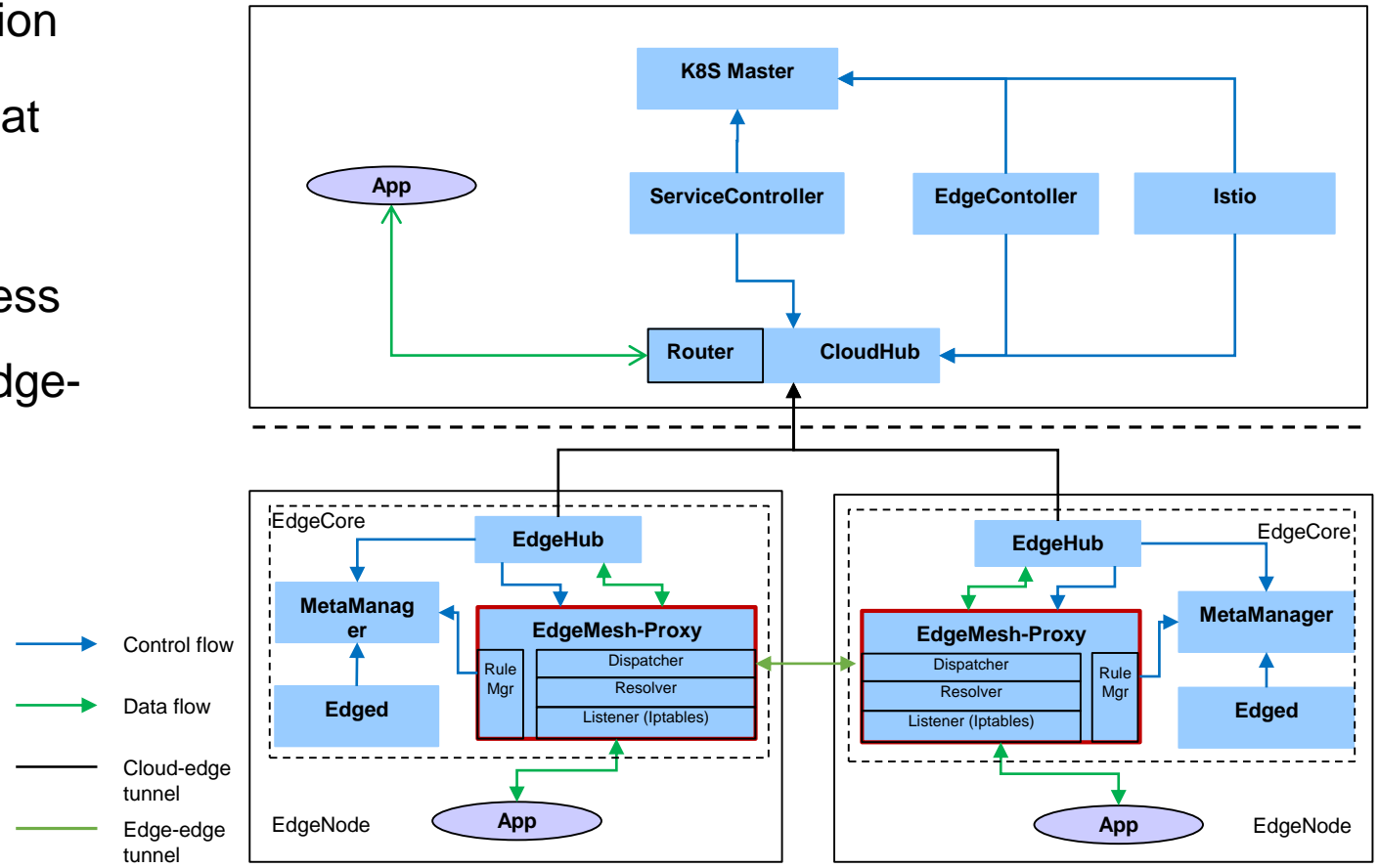
CloudNativeCon

North America 2019



EdgeMesh: ServiceMesh in KubeEdge

- Service governance with Istio integration
- EdgeMesh-proxy forwards data flows at the edge
- Consistent service discovery and access experiences across edge-edge and edge-cloud
- P2P tech is used for communication across subnets



Edge - Cloud communication

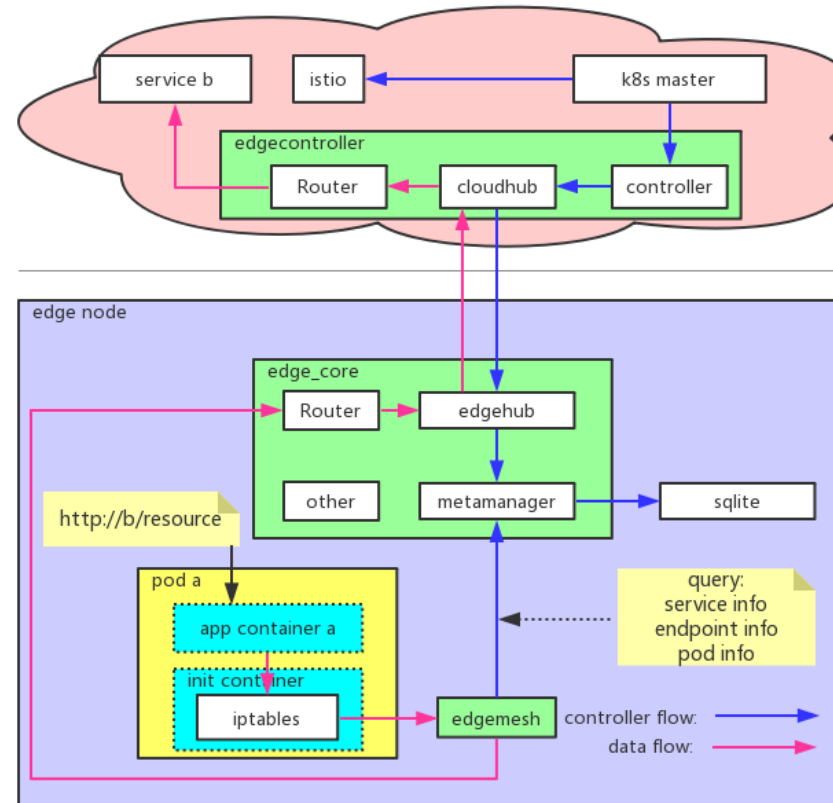


KubeCon



CloudNativeCon

North America 2019



EdgeSite: Clusters at edge



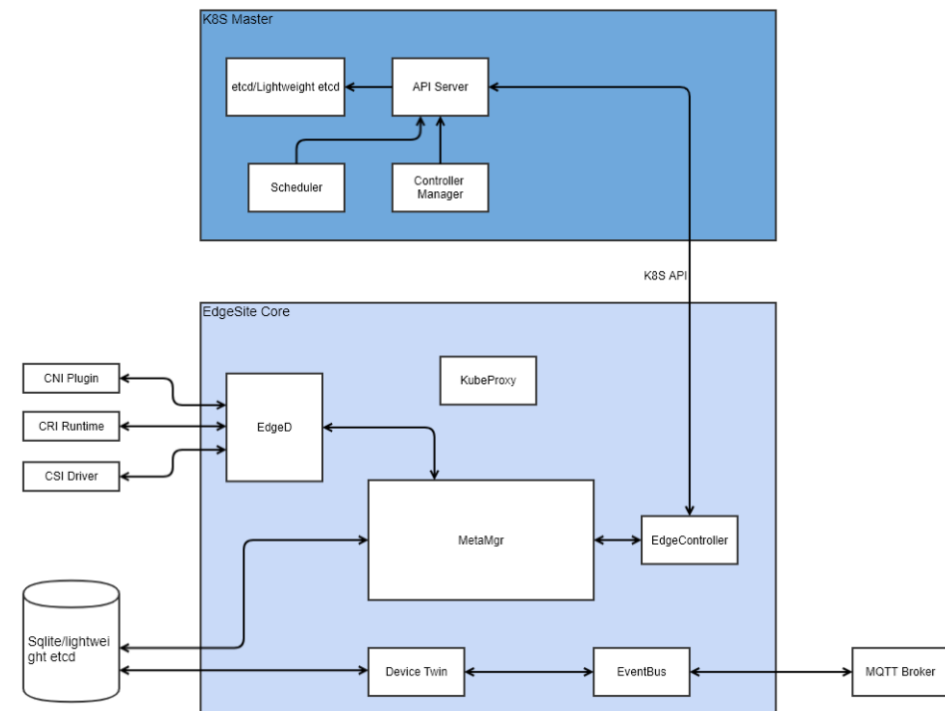
KubeCon



CloudNativeCon

North America 2019

- Enable customers to run a lightweight K8s cluster at the edge where the control plane can support HA.
- KubeEdge pluggable module framework with edged devices, edgemesh integrated
- Conformant K8s APIs/functionalities



Managing edge clusters from cloud

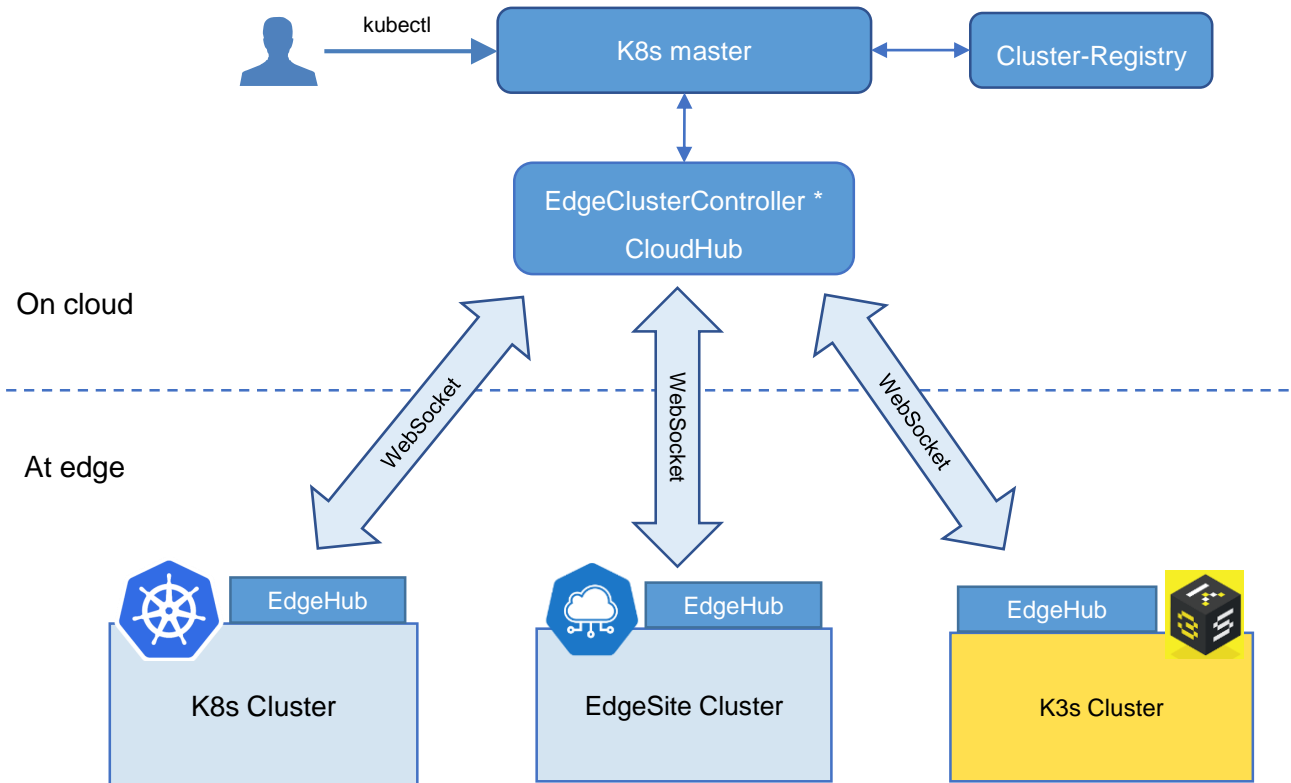


KubeCon



CloudNativeCon

North America 2019



Roadmap

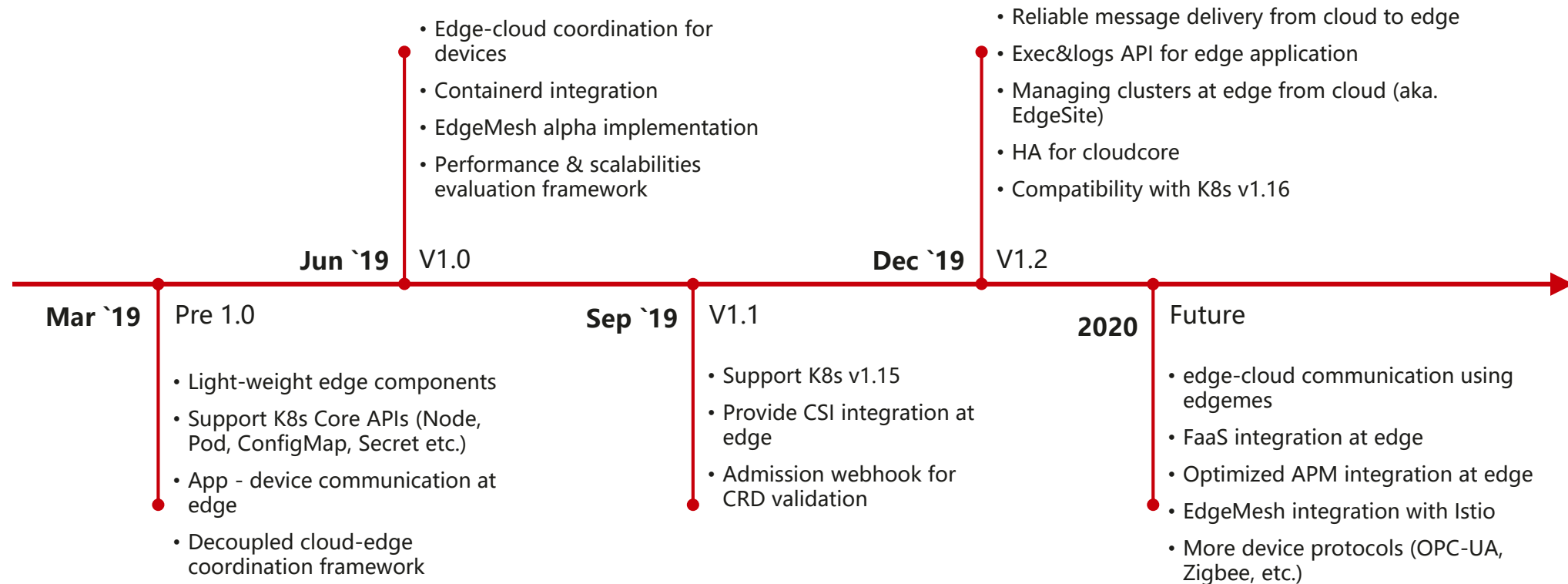


KubeCon



CloudNativeCon

North America 2019



Join Us



KubeCon



CloudNativeCon

North America 2019

- Website: <https://kubedge.io>
- Github: <https://github.com/kubedge/>
- Slack channel: <https://kubedge.slack.com>
- Mailing group: <https://groups.google.com/forum/#!forum/kubedge>
- Bi-weekly community meeting: <https://zoom.us/j/4167237304>
- Twitter: <https://twitter.com/KubeEdge>
- Documentation: <https://docs.kubedge.io/en/latest/>



KubeCon



CloudNativeCon

North America 2019

Thank you!

