



Kubernetes IoT Edge Working Group:

Introduction and survey of
Kubernetes related solutions for
IoT and Edge

Steven Wong, VMware
Luwei He, Huawei



KubeCon



CloudNativeCon

OPEN SOURCE SUMMIT

China 2019



Abstract

Hidden slide during presentation – included for those finding deck online later

Kubernetes describes itself as "a portable, extensible open-source platform for managing containerized workloads and services".

There are edge and IoT use cases that challenge some aspects of Kubernetes.

- Run Kubernetes full clusters at edge locations (e.g. retail store and remote branch office) under tight resource constraints
- Run containerized workloads at edge with a remote Kubernetes control plane at a cloud or central location
- Run devices using non containerized software at edge

This session will survey of some existing solutions in the edge and IoT space, covering how they address issues like security, messaging, resource constraints, data communications, application lifecycle management, etc. while working within Kubernetes limitations.

We will wrap up with a report on ongoing activity within the IoT Edge working group and details on how you can get involved.



Presenters

Luwei He



Standard Engineer,
Huawei

Luwei is currently working on some open source work about edge computing, and participating in and contributing to the related community, e.g., Kubernetes, KubeEdge, Akraino, OpenStack and more.

GitHub, WeChat, other?

Steven Wong



Los Angeles

Open Source Community Relations Engineer,
VMware

Co-lead of the Kubernetes IoT + Edge Working Group. Also active in other Kubernetes SIGs and WGs since 2015. Former engineer and architect of Wonderware HMI/SCADA industrial IoT products.

GitHub: @cantbewong

Agenda

Survey of survey of some existing solutions in the edge and IoT space. How they address:

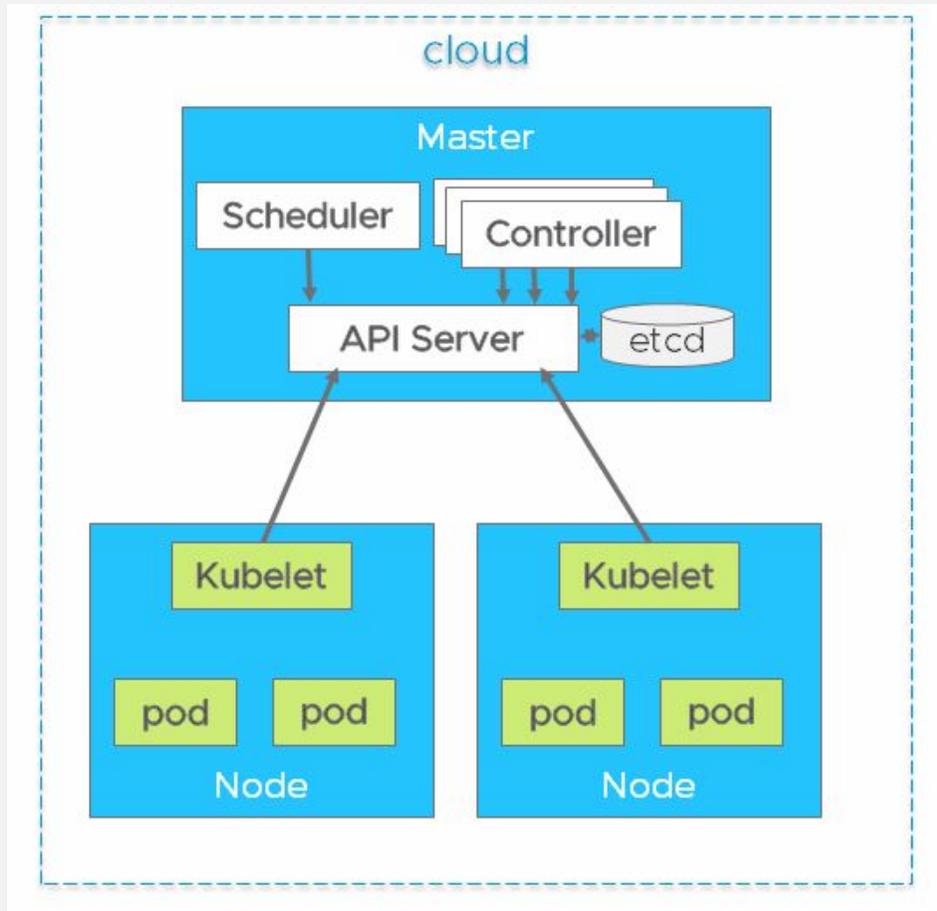
- Security
- Messaging
- resource constraints
- data communications
- application lifecycle management

Ongoing activity within the IoT Edge working group

How you can get involved.

Standard Kubernetes Architecture

Composed of a control plane + node(s) hosting workloads



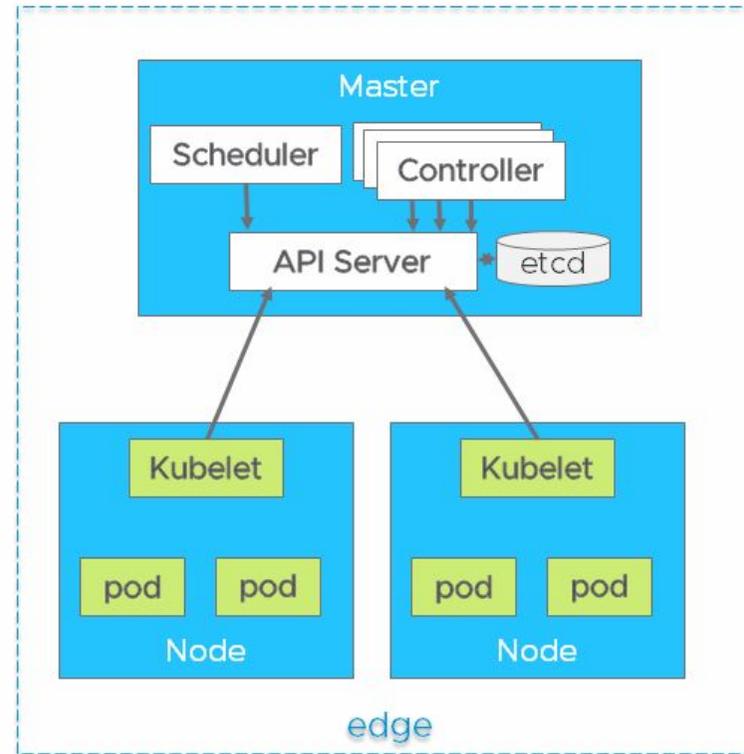
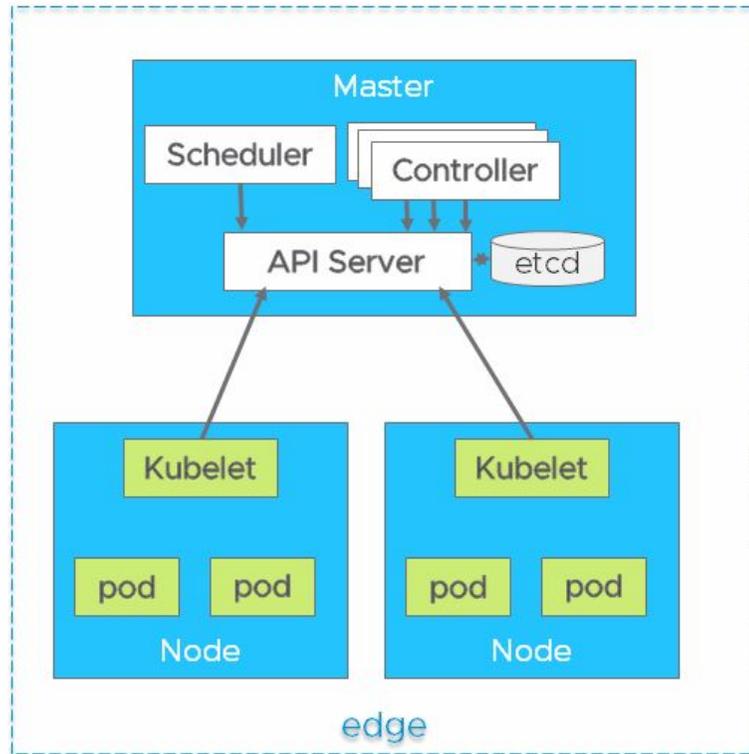
Originally designed for hosting workloads on large public clouds

Can have issues in IoT and edge applications:

- Do you want 1,000's of control planes?
- Is this suitable for:
 - Operation at locations without IT staff
 - Operation at sites with limited resources?

Option 1: whole clusters at edge

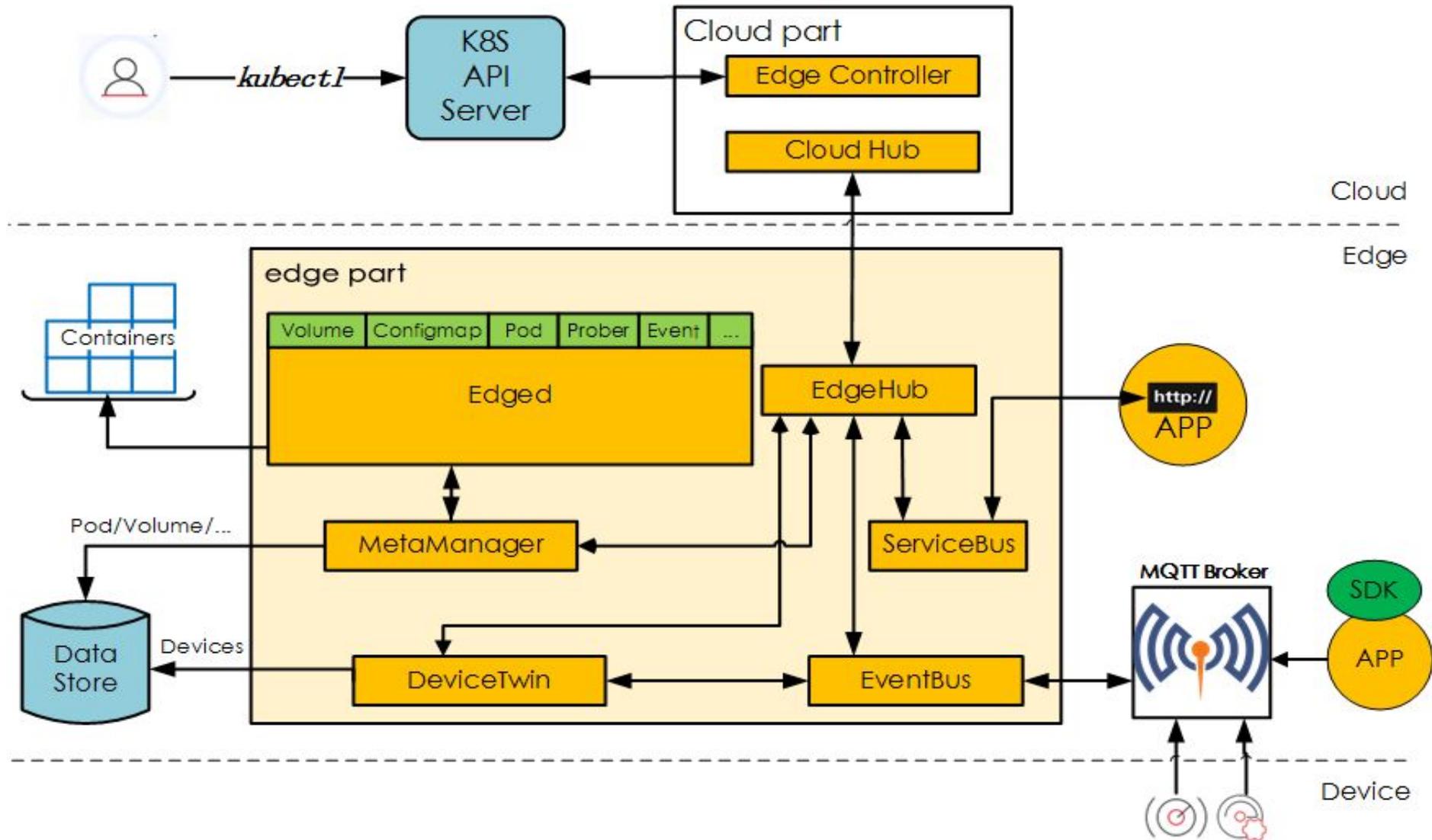
Maybe a lot of them



Kubernetes might be “forked” to remove unneeded features and alter components to reduce resource demands

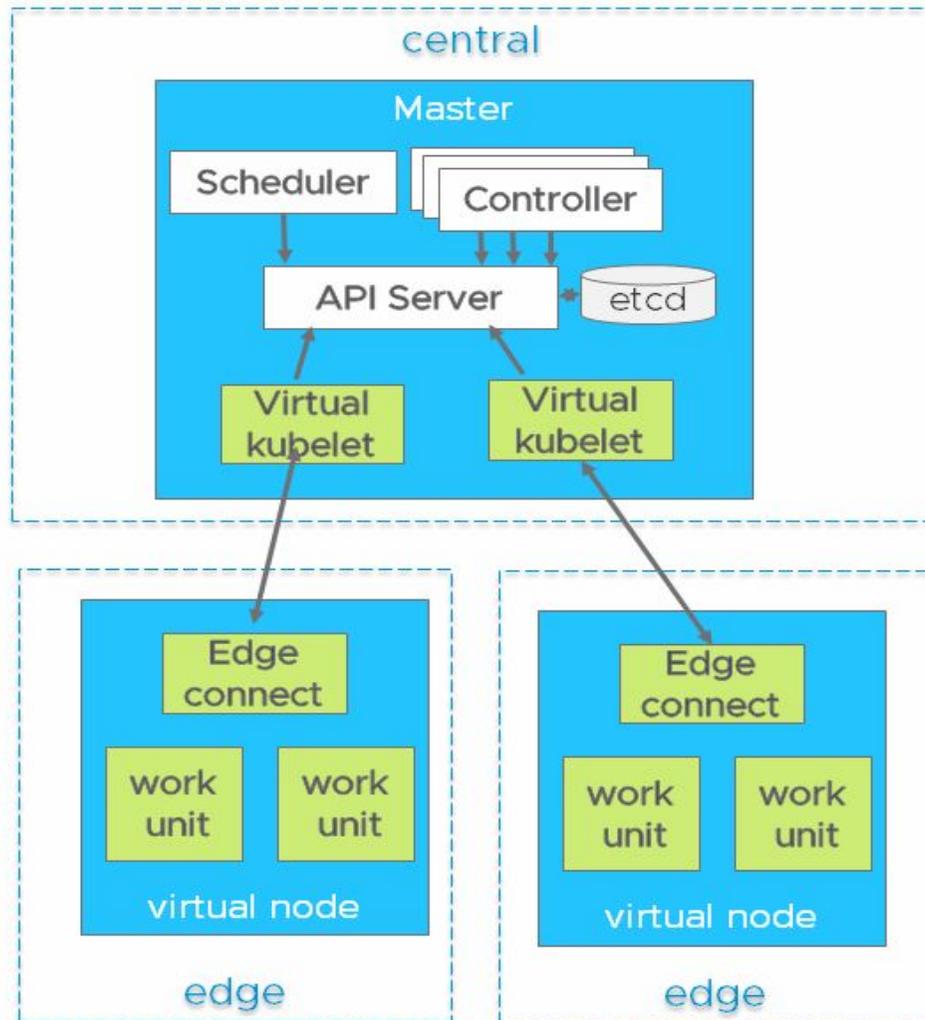
Option 2: central Kubernetes control plane managing edge

Edge runs containerized workloads using kubelets or “forked” lightweight kubelets



Option 3: central Kubernetes control plane managing edge

Virtual kubelet(s) are in cloud



Virtual kubelets implement “supervisory control” over edge workloads that might be containerized – but could also be non-containerized, non-Linux OS, or no OS at all

Might be possible to do this using CRDs and controllers instead of virtual kubelet

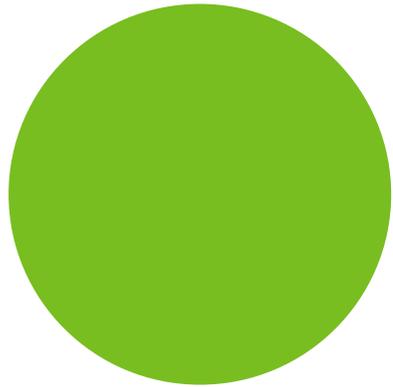
Survey of IoT Edge variants using Kubernetes

Open source

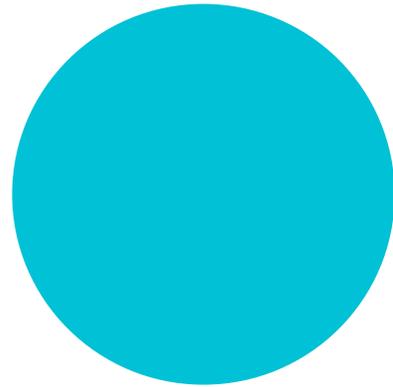


Differentiating factors

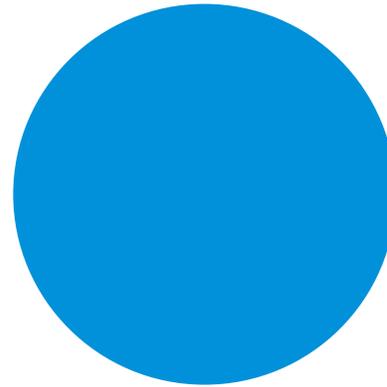
A few common Kubernetes enhancements for edge, some solutions may have more



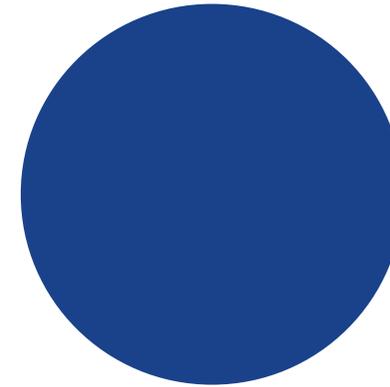
Adjunct to
“pure”
upstream
Kubernetes or
forked



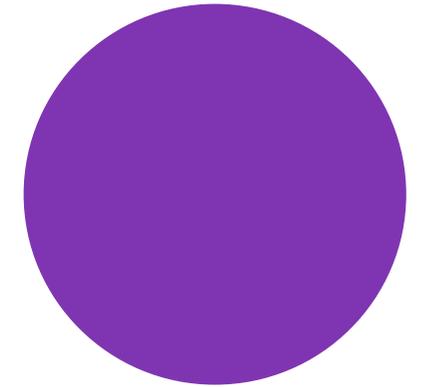
Added
networking
support



Added security
features



Adds
management
of devices and
hosting
hardware



Features to
deal with
reduced
resources at
edge

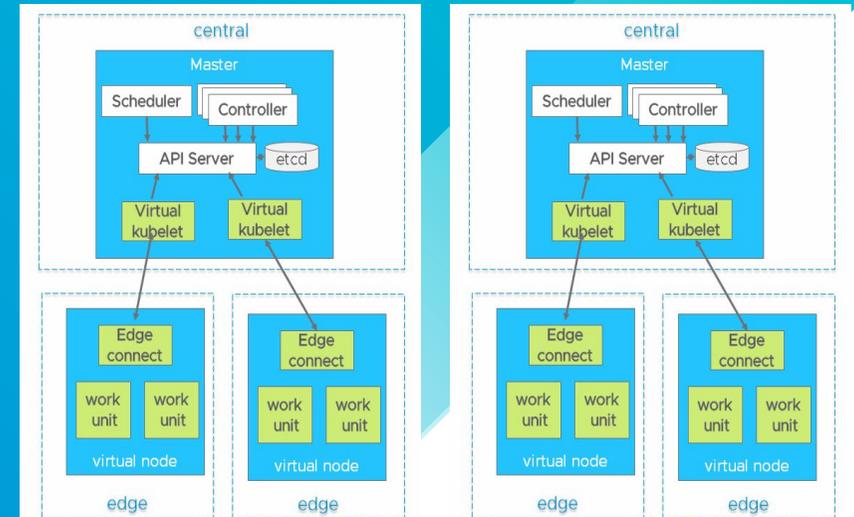
Whole clusters at edge

Common use case – remote office, retail

use case examples:

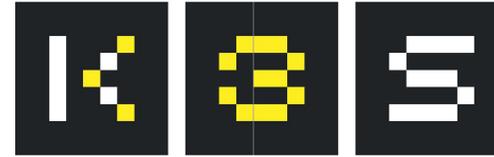
Target
tech.target.com/infrastructure/2018/06/20/enter-unimatrix.html

Chick-fil-A
sched.co/GrXv



Rancher K3s

source repo: github.com/rancher/k3s



Edge optimized micro distribution of Kubernetes

key edge related features

- supports both x86 + ARM CPU
- supports nodes with memory <4GB
- designed with simplified operation for embedded or offline (air-gapped)

Removes

Legacy and non-default features
Alpha features
In-tree cloud providers
In-tree storage drivers

Adds

Simplified installation
TLS management
Automatic Manifest and Helm Chart

Swaps & Opinions (some optional)

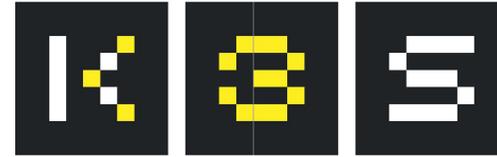
etcd -> SQLite2

Docker - > containerd

CNI = flannel, DNS = CoreDNS, Ingres = Traefik

Rancher k3s

resources



landing page:

k3s.io/

blogs:

rancher.com/blog/2019/announcing-k3os-kubernetes-operating-system

fedoramagazine.org/kubernetes-on-fedora-iot-with-k3s/

demo video:

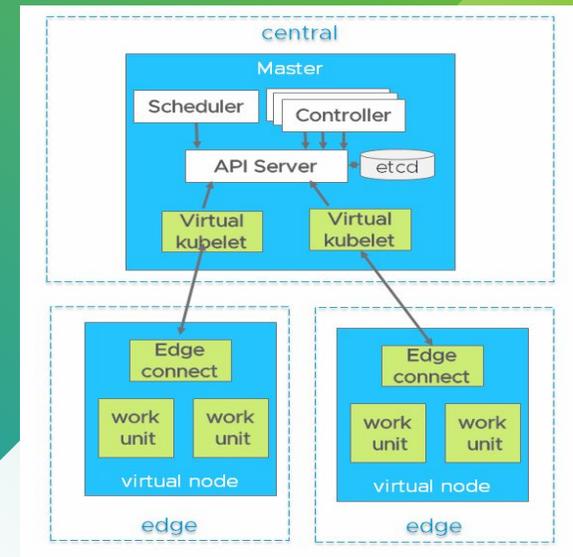
youtu.be/WYPd7i15X0g

Central Kubernetes control plane managing edge

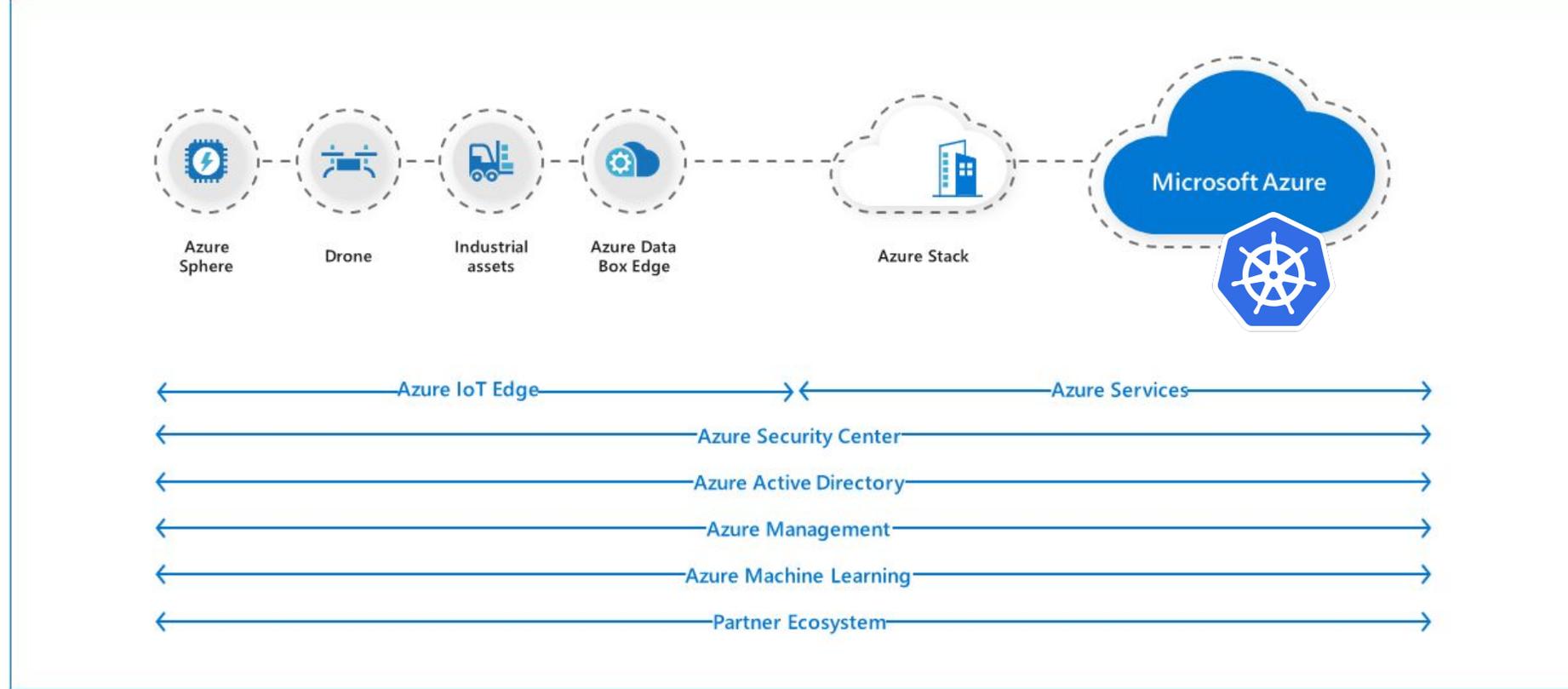
Device twins are in cloud
edge might not be Kubernetes

use case example: Bosch

blog.bosch-si.com/bosch-iot-suite/adopting-kubernetes-to-build-iot-solutions/

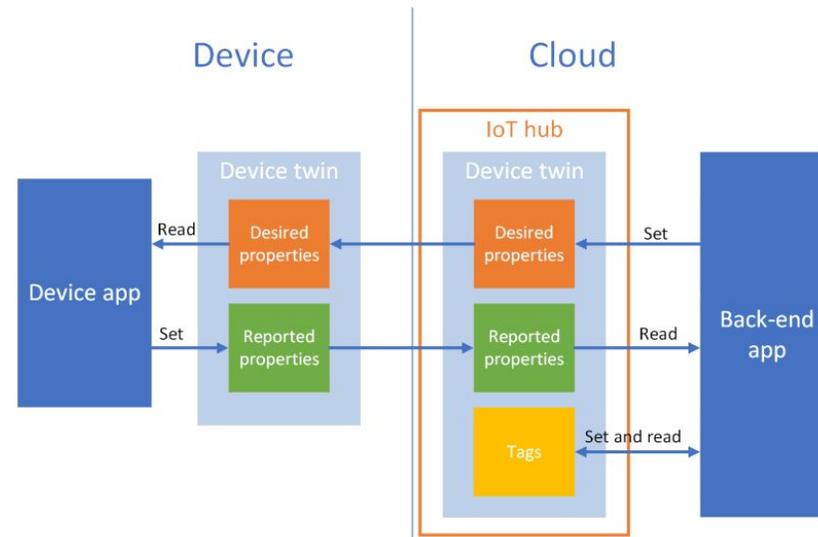


Azure IoT Edge



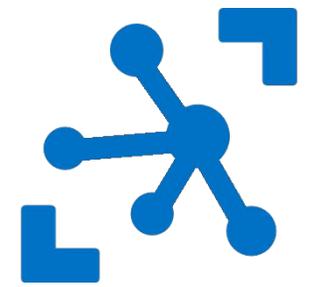
Kubernetes control plane in cloud -
Device twins - split:
one in cloud, one at edge

picture source: microsoft.com

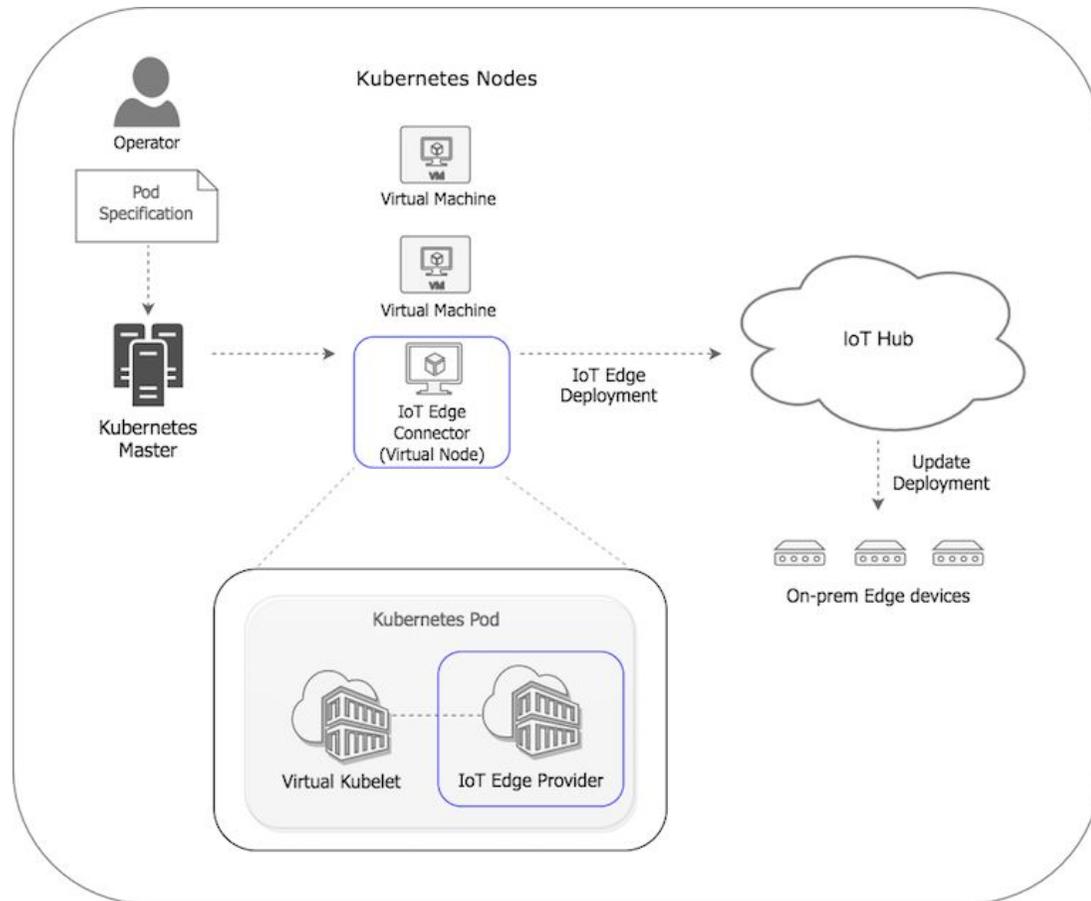


Azure IoT Edge

source repo: github.com/azure/iot-edge-virtual-kubelet-provider



Uses a a Kubernetes control plane with a cloud hosted [Virtual Kubelet](#).



The IoT Edge provider handles *kubelet* API calls forwarded by the virtual kubelet. It talks to the Azure IoT hub using the Azure IoT SDKs to submit an transformed container specification. Tags, annotations and configmaps can encode IoT edge specific information.

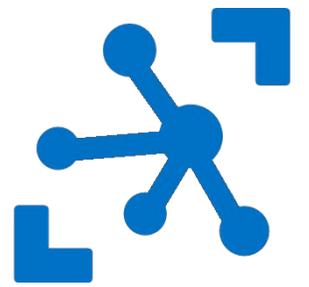
If the edge device is offline when applying a new configuration, it gets pushed as soon as the device gains connectivity.

Edge Hub acts as a communication broker facilitating local device communication. It supports standard protocols of IoT Hub including AMQP, MQTT.

picture source: microsoft.com

Azure IoT Edge

resources



landing page:

azure.microsoft.com/en-us/services/iot-edge/

blogs:

aka.ms/iotedgek8s-blog

azure.microsoft.com/en-us/blog/azure-iot-edge-generally-available-for-enterprise-grade-scaled-deployments/

blogs.microsoft.com/iot/2018/05/07/unlocking-the-iot-promise-from-the-intelligent-cloud-to-the-intelligent-edge/

demo video:

youtu.be/01kHKNN3z-0?t=3824

KubeEdge

source repo: github.com/kubeedge/kubeedge



KubeEdge

Extends cloud native containerized application orchestration capabilities to hosts at Edge.

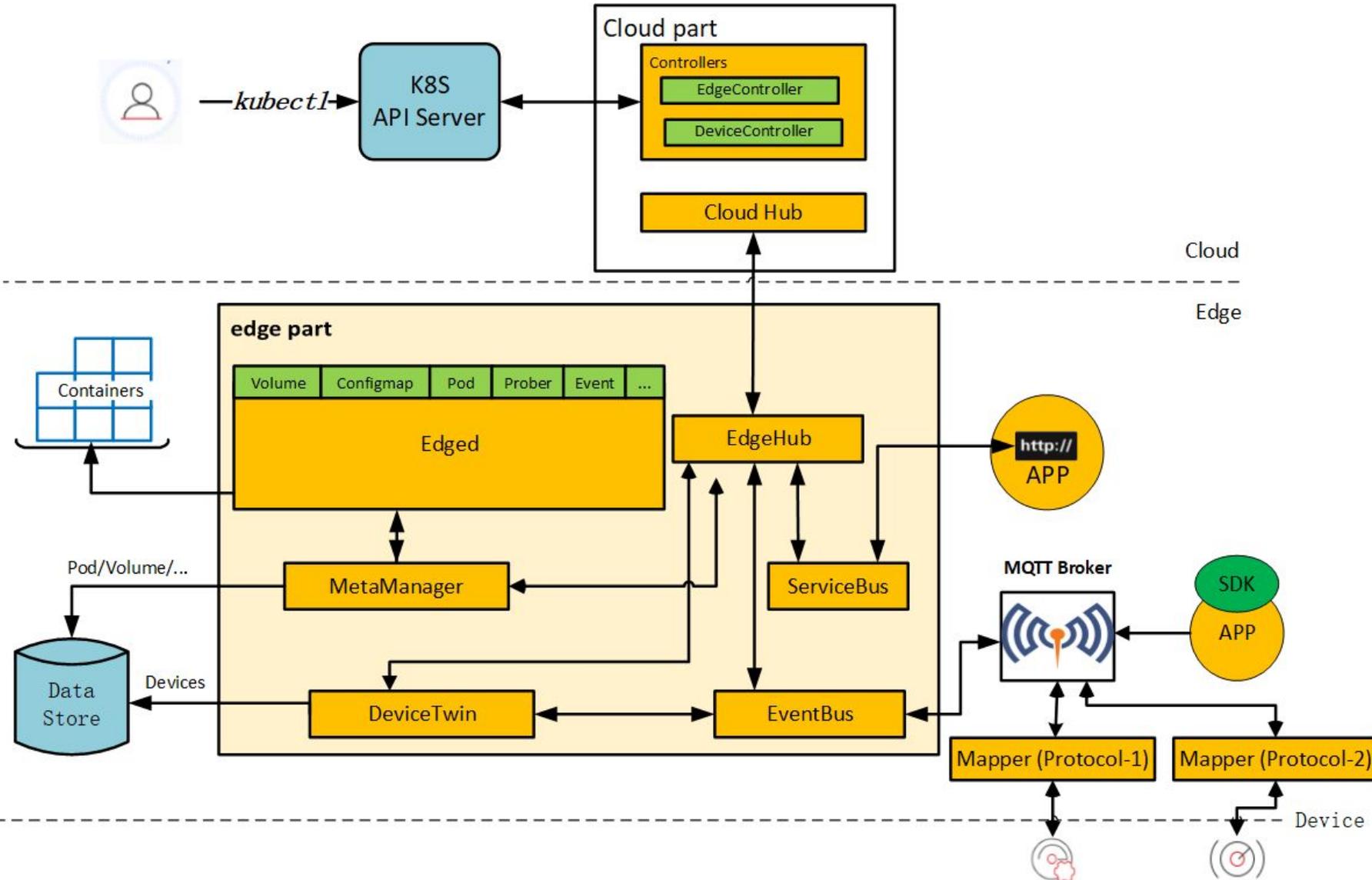
- Built upon Kubernetes
 - control plane is Kubernetes
 - edge nodes are Kubernetes based, with deployment of configmap, secrets, etc supported
- Supports app. deployment and metadata synchronization between cloud and edge
- Incorporates managed http and MQTT network service, cloud to edge, and edge to edge
- Uses “device twin” model to manage intermittent cloud to edge connectivity
- [CNCF sandbox](#) project

KubeEdge

architecture



KubeEdge



KubeEdge

resources

landing page:

kubedge.io

blogs:

thenewstack.io/kubedge-extends-the-power-of-kubernetes-to-the-edge/

kubernetes.io/blog/2019/03/19/kubedge-k8s-based-edge-intro/

videos:

intro youtu.be/pdq1ANkp0Ms

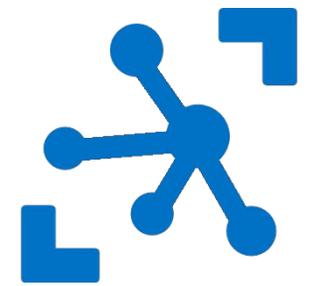
deep dive youtu.be/LuiYF2vi2nE



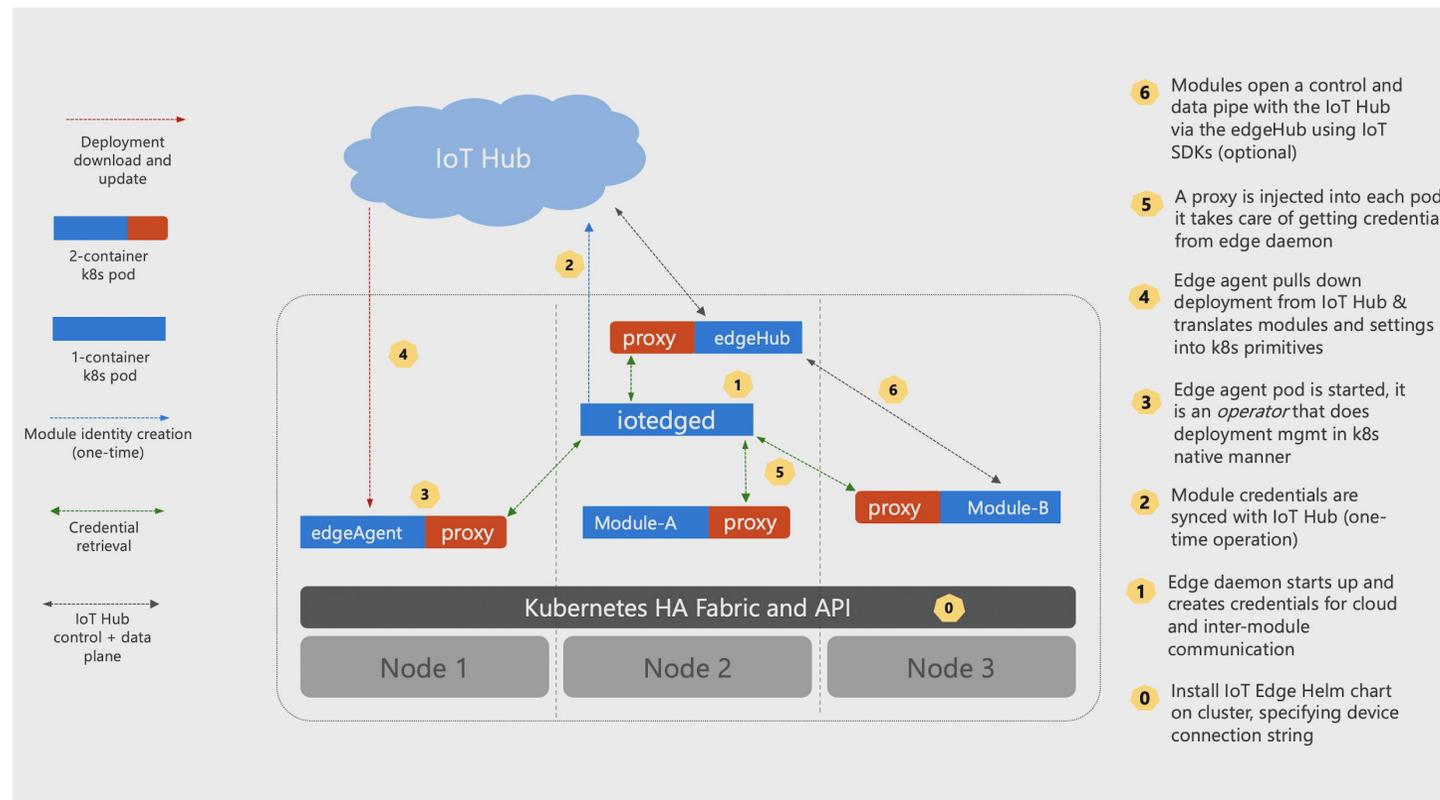
KubeEdge

Azure IoT Edge (preview) - Kubernetes clusters at edge

source repo: github.com/Azure-Samples/iotedge-gateway-on-kubernetes



Edge workloads deployed to on premise Kubernetes clusters. Uses [Custom Resource Definitions](#) (CRDs), with a [Controller](#) (IoT Edge Agent) that reconciles cloud managed desired state with the local device state



changes:

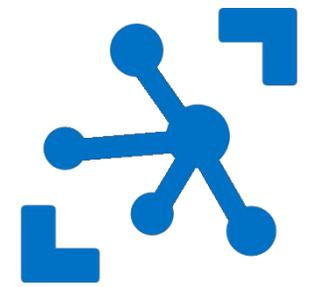
no virtual kubelet

at edge, no hard requirement for Docker CRI - Kubernetes instead

picture source: microsoft.com



Azure IoT Edge (preview) - Kubernetes clusters at edge resources



landing page:

docs.microsoft.com/en-us/azure/iot-edge/how-to-install-iot-edge-kubernetes

blog:

itnext.io/azure-iot-edge-workloads-on-kubernetes-1065b801cf4f

demo video:

youtu.be/tHCtuaY3qZA

deep dive video:

youtu.be/29j9RDCvIeY

Open Source IoT + Edge Tools

With support for use with Kubernetes

our charter:

“A Working Group dedicated to discussing, designing and documenting using Kubernetes for developing and deploying IoT and Edge specific applications”

Enmasse



source repo: github.com/EnMasseProject/enmasse

Managed, self-service messaging on Kubernetes. Built-in authentication and authorization of clients and identity management

Key features:

- Runs on Kubernetes: deploy on-premise or in the cloud
- Different messaging patterns like request-response, pub-sub and events
- Decouple operation of infrastructure from configuration and use by applications

landing page:

enmasse.io/

video:

youtu.be/ZIp9EPQ25eM



Eclipse Hono

source repo: github.com/eclipse/hono



Provides remote service interfaces for connecting **large** numbers of IoT devices to a back end and interacting with them in a **uniform** way regardless of the device communication protocol

Supports:

- data ingestion (telemetry data)
- command & control type message exchange patterns
- provisioning & security aspects, managing device identity and access control rules
- Deployment on Kubernetes (www.eclipse.org/hono/deployment/kubernetes/)

Hono is not a messaging system itself, it operates at a layer above messaging infrastructure to work such as MQTT.

landing page:

www.eclipse.org/hono/

video:

youtu.be/6m_boZU97Ks



Eclipse Ditto

source repo: github.com/eclipse/ditto



“Digital twins” is a concept of maintaining a cloud based representation of a living device that may be intermittently connected

Supports:

- An API to deal with devices
- State management, synchronization of state changes handles eventual consistency tracking desired state, last reported (known) state, and current (live) state of devices.
- Digital twin search functionality based based on metadata and state data
- Kubernetes deployable (described here: github.com/eclipse/ditto/tree/master/deployment/Kubernetes)

landing page:

www.eclipse.org/ditto/

video:

youtu.be/NpC4R0GqwKc



Eclipse hawkBit

source repo: github.com/eclipse/hawkbite

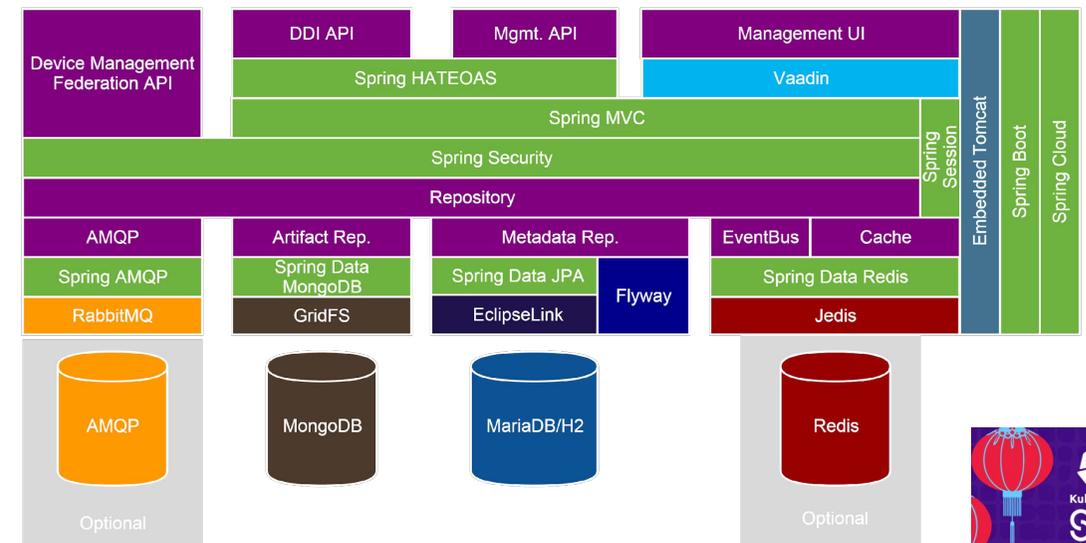


Framework for rolling out software updates to constrained edge devices as well as controllers and gateways connected to IP based networking infrastructure

Key features:

- Scalable to millions of devices, and terabytes of software, on a global scale
- Supports complex rollout strategies (grouping, cascading, error detection)
- Supports standard and proprietary protocols
- Can run on Kubernetes, deployed via Helm chart

(www.vand.io/chart/kiwigrd/hawkbite-update-server/)



landing page:

www.eclipse.org/hawkbite

video:

youtu.be/8vcLXs91c-4



More open source tools that run on or with Kubernetes

this session is not long enough to cover everything

[EdgeX Foundry](#) : deploy containerized workloads to edge, Docker is reference runtime but orchestrator agnostic

blog: www.edgexfoundry.org/blog/2018/01/30/huawei-runs-edgex-foundry-kubernetes/

[Akraino](#): designed to improve the state of edge cloud infrastructure for enterprise edge, OTT edge, and carrier edge networks

wiki: <https://wiki.akraino.org/display/AK/Kubernetes-Native+Infrastructure+%28KNI%29+Blueprint+Family>

Thank You

This deck: <http://tbd>

How to get involved with the IoT and Edge Working Group, learn more...

Regular Work Group Meeting:

USA WG Meeting Wednesday 9am PT, every 4 weeks, next on July 17

APAC WG meeting Wednesday 5 UTC every 4 weeks, next on July 3

- [Meeting notes and agenda](#)

Link to join the group

- groups.google.com/forum/#!forum/kubernetes-wg-iot-edge

Link to join Slack

- <https://kubernetes.slack.com/messages/wg-iot-edge>

White Paper

- <http://bit.ly/iot-edge-whitepaper>

Q&A