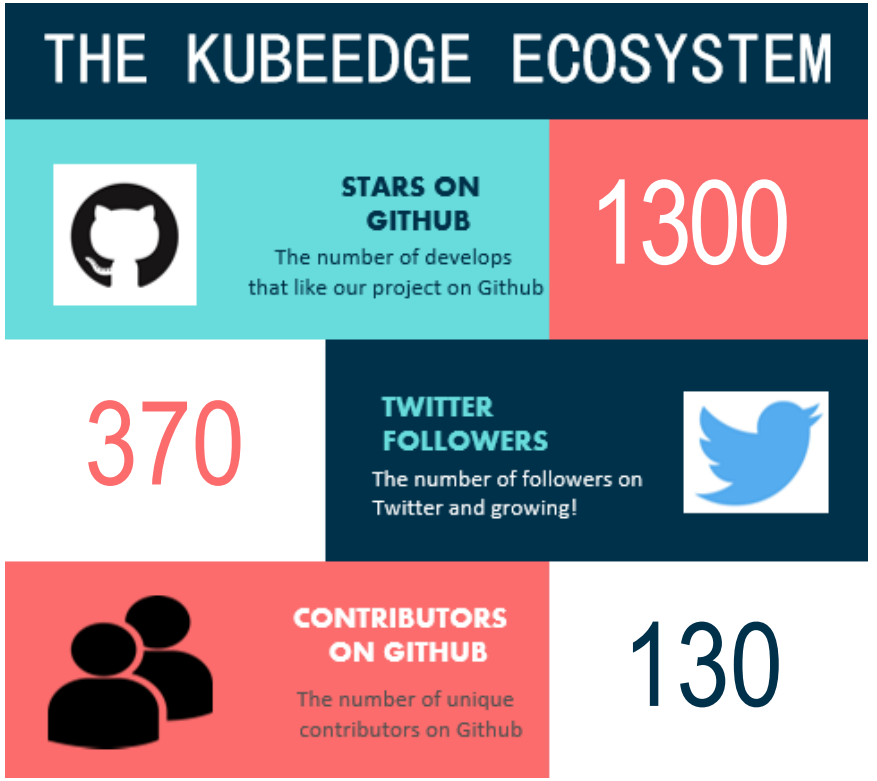
KubeEdge

Kubernetes Native Edge Computing Framework

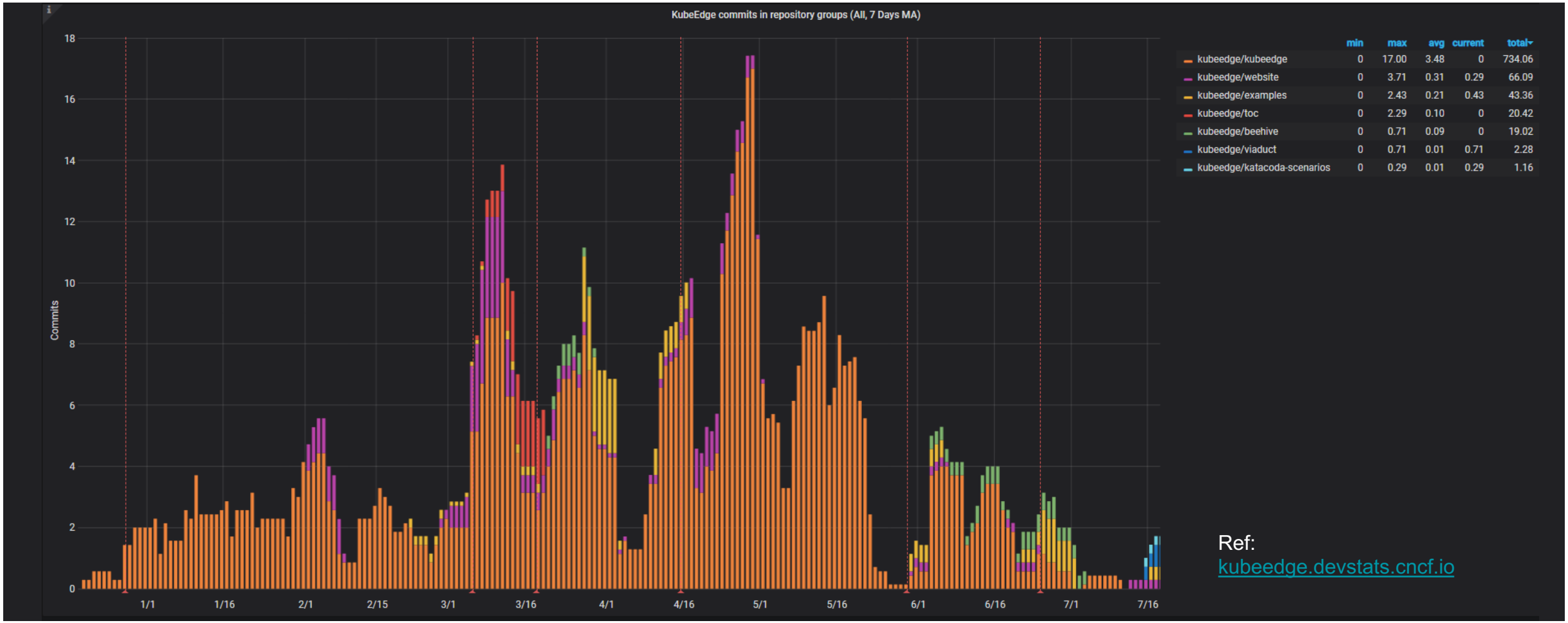
Kevin Wang
@kevin-wangzefeng

About KubeEdge

- KubeEdge targets to achieve edge computing with Kubernetes:
  - CNCF Sandbox project
  - Coordination between cloud and edge
  - 4 releases and v1.0
  - Reference architecture by K8s IOT/Edge WG
- Contributors from companies: China Mobile, China Unicom, ARM, TenxCloud, Intel, Amazon, JD.com, Zhejiang University, EMQ, etc.
- Companies shown interests: Microsoft etc.



KubeEdge Devstats



Ref:
kubeedge.devstats.cncf.io

Our vision and mission

- Enable users to manage resources and run applications natively at cloud and edge
  - A K8s based infrastructure for IOT/Edge computing.
  - Manage resources and orchestrate applications in the same way, regardless of the location: cloud or edge
  - K8s APIs and primitive types
  - Edge is an extension of cloud
  - Worker nodes can be at cloud or edge
  - Highly extensible covering diversified network topology, device protocol, storage backend, etc.

Building Edge platform With Kubernetes

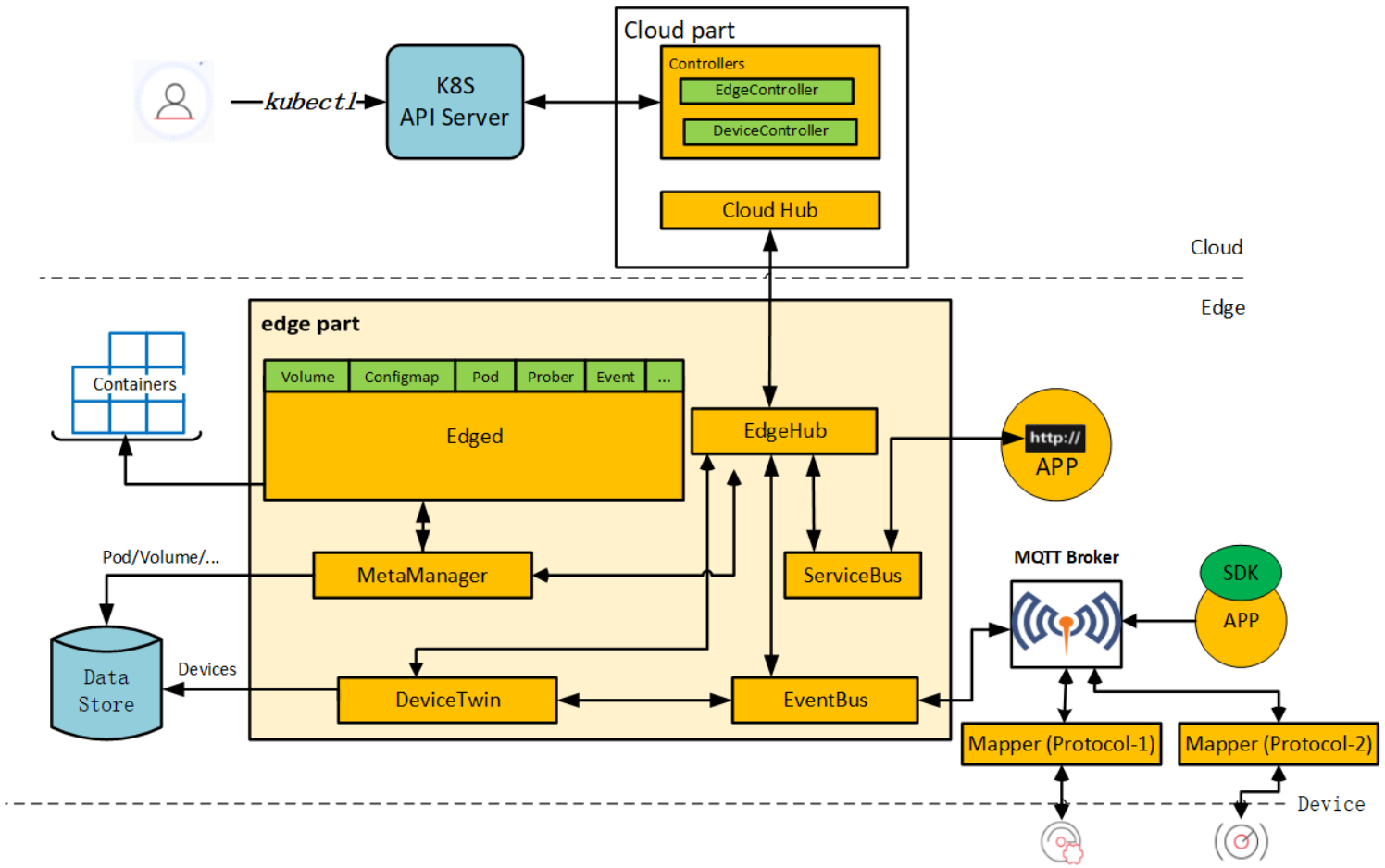
Benefits

- Containerized Application
  - Build once, run anywhere
  - Lightweight base image
- General application abstraction
  - Already become standard
  - Same experience across cloud and edge
- Extendable Architecture
  - Extendable API machinery
  - Easy to add customized components

Gaps

- Limited resource at Edge
  - Not enough for vanilla K8s, even just a Kubelet.
- Unstable network
  - Private network, limited bandwidth, latency, etc.
- Need autonomy at Edge
  - Edge may get offline/disconnected often
  - Should not evict/migrate applications when disconnected
- Device management at Edge
  - Quite different requirement with current device plugin

KubeEdge Architecture

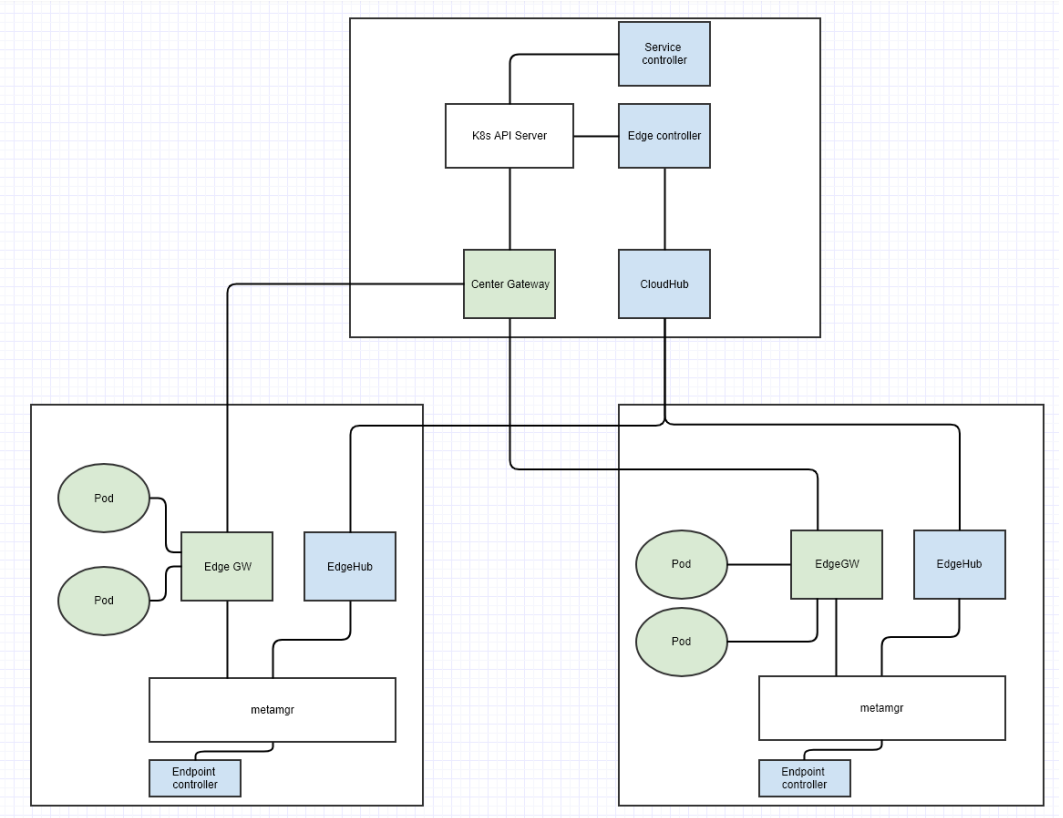


- Coordination between cloud and edge
- Fundamental infrastructure support
- Device management and messaging
- **Versatile device protocols:** Bluetooth, Modbus, OPC-UA etc.
- **Edge side autonomy:** works fine with intermittent network
- **Small footprint:** ~10MB mem for nodes

KubeEdge

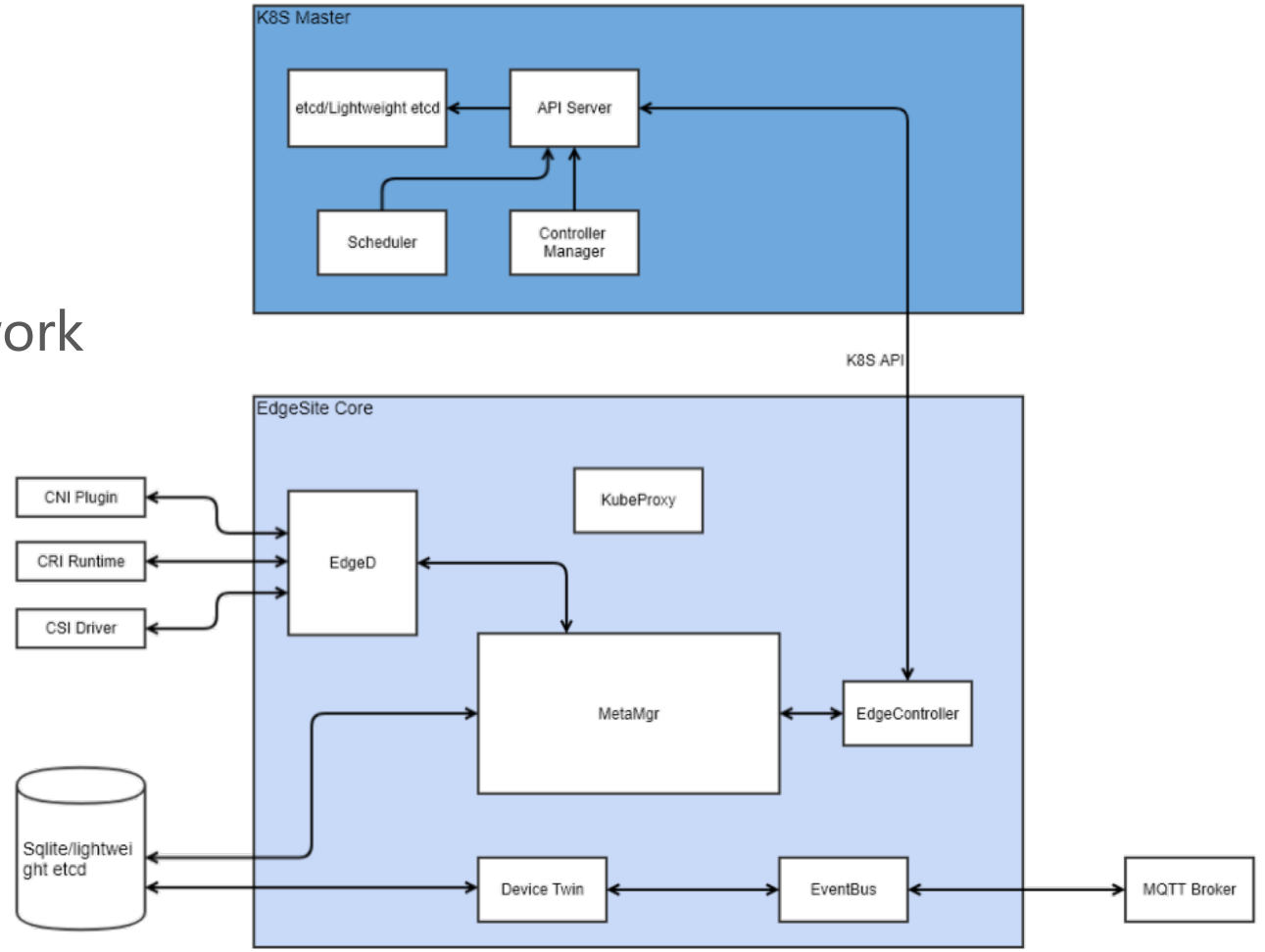
New in 1.0 – EdgeMesh

- K8s service discovery, routing and lifetime management
  - Support service and endpoint
  - Service discovery
  - Support north-south and east-west network routing
  - Mesh for services at edge and cloud



New in 1.0 – EdgeSite

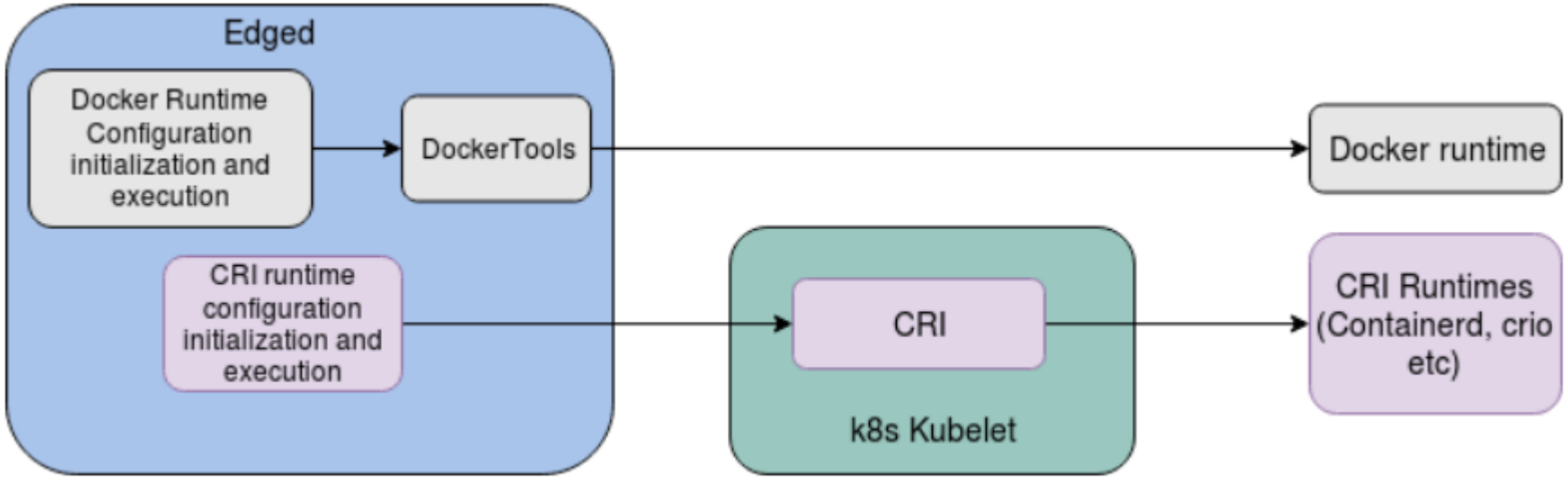
- Enable customer to run a lightweight K8s cluster at edge where control plan can support HA
- The KubeEdge pluggable module framework
- Conformed K8s APIs



KubeEdge

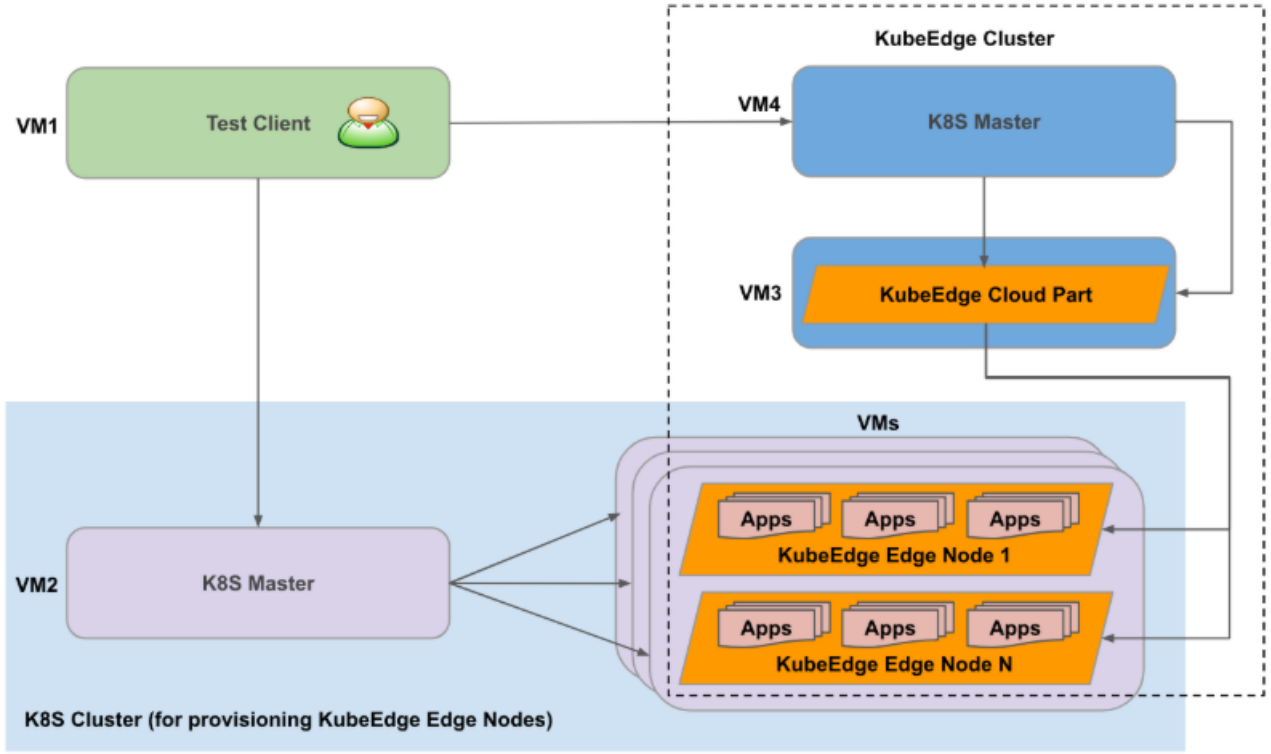
New in 1.0 – CRI for Edged

- Support multiple container runtime e.g. docker, containerd, cri- o etc. on the edge node
- Support light weight container runtime on edge node

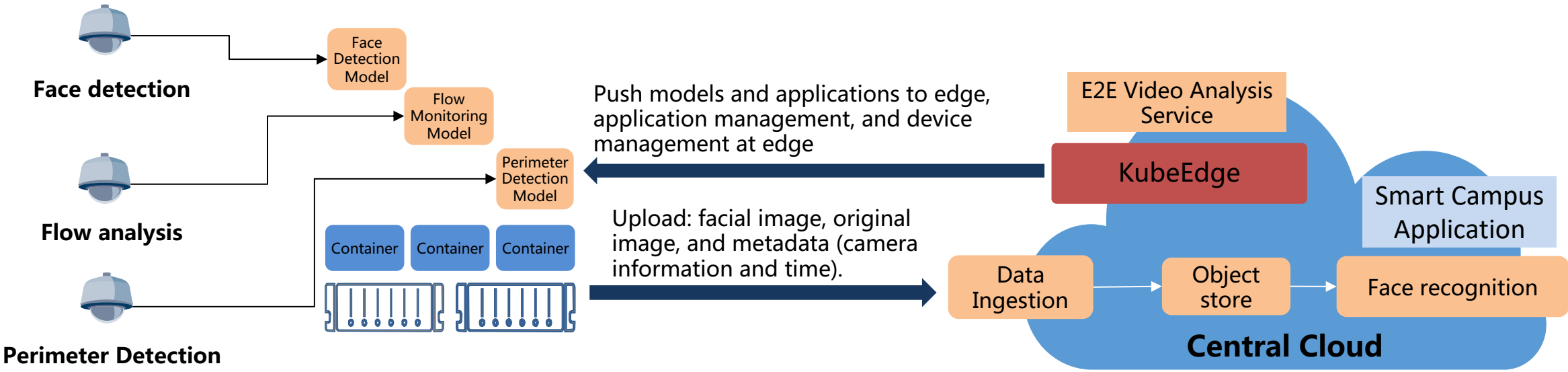


New in 1.0 – Perf and Scale Evaluation

- Evaluate the latency and throughput between edge and cloud
- Scalability
  - How many edge nodes can be managed in a cluster
  - How many pods can run on an edge node.
    E.g. Raspberry-pi



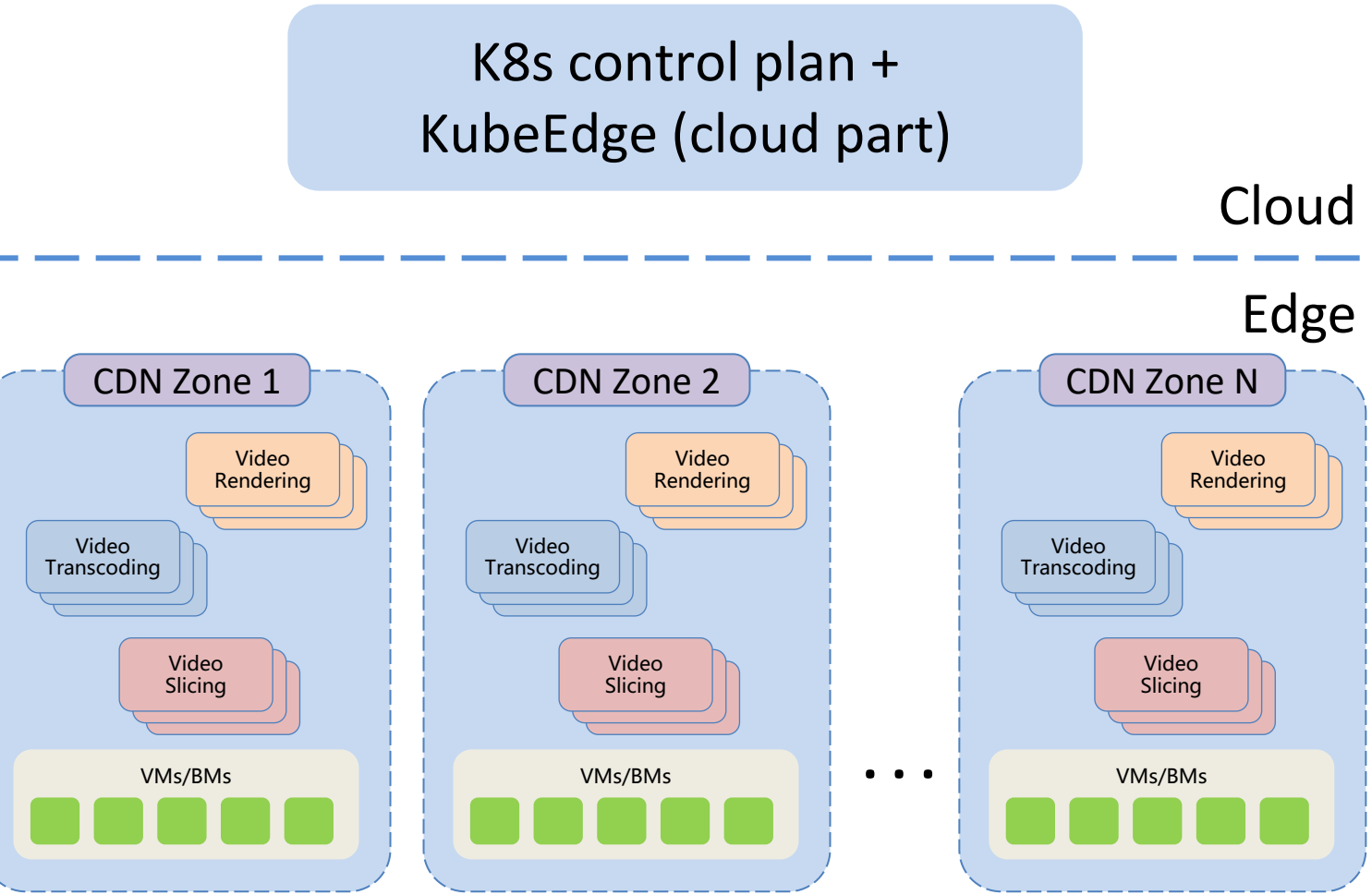
Smart Campus with KubeEdge



Learnings:

1. **Low latency:** video flow analyzed at edge, cutout useful pictures to upload
2. **Business Value:** Smart analysis on surveillance video, real time detect trespass, flow bursting etc., optimize labour costs
3. **Edge-cloud coordination:** edge application lifecycle management and rolling update
4. **Training on cloud:** Automatic training, easy to scale in/out and update
5. **Camera compatibility:** able to work with legacy IPC cameras, act like smart cameras with cloud-edge coordination

Building edge CDN with KubeEdge



Typical requirements:

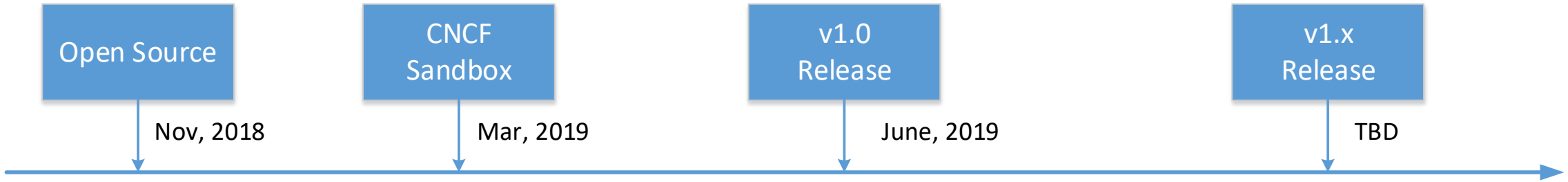
- Many servers at the edge
- CDN sites controlled by the central cloud
- Workloads dispatched by central cloud
- Mainly run video transcoding, rendering, slicing etc. at edge, managed as Job and Deployment
- Need auto-scaling and pod priority
- No need on services and shared storage

Learnings:

- Unified application management
- Autonomy for CDN sites at edge
- Low system overhead at edge
- Cons:
  - Auto-scaling can’t work when disconnected
  - No able to “replace” a pod when disconnected

KubeEdge

Roadmap



1. Lightweight agent at edge
2. Kubernetes core primitive support, e.g. Node, Pod, ConfigMap, Secret etc.
3. Device twin and MQTT protocol for IoT devices talking to Edge node
4. Loosely coupled Edge/Cloud communication and data sync channel
5. Run native containerized applications

1. Service mesh enabling network for data plane
2. Performance and scalability evaluation framework
3. Kubernetes based Device Management through CRD
4. Security integration with SPIFFE/SPIRE
5. Support CRI for Edged

1. Enable monitoring and telemetry
2. Standardization of Edge-Cloud Communication
3. Support more device protocols
4. Evaluate and enhance performance and scalability
5. Support CSI for Edged
6. Contributor experience

Resources

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

WeChat Public Account