

Kubernetes IoT Edge Working Group

Using Event Driven Architecture at Edge

Steven Wong

Staff Open Source Software Engineer
VMware

Dejan Bosanac

Senior Software Engineer
Red Hat

[November 20, 2020](#)

[November 20, 2020](#)



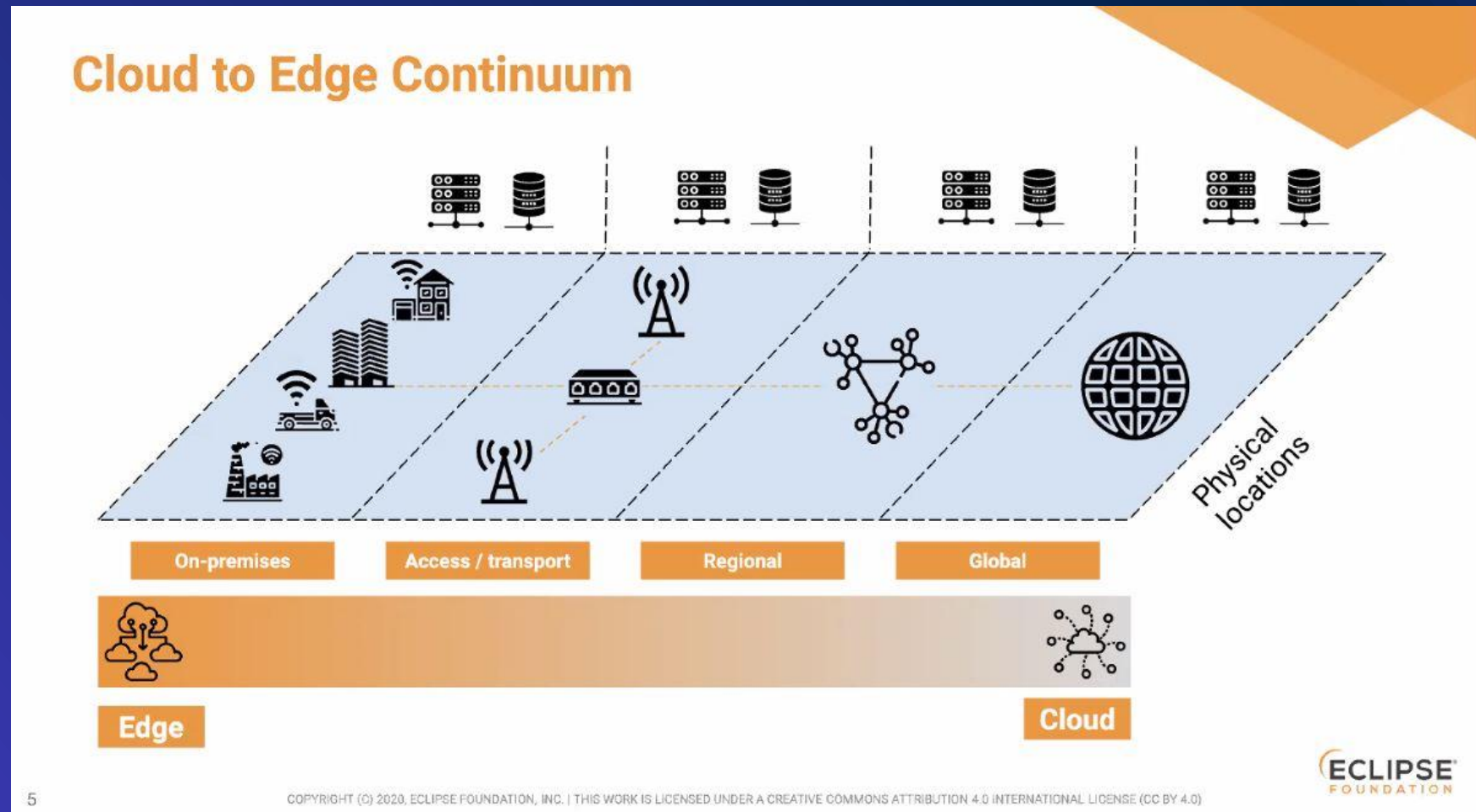
Agenda

- Characteristics and challenges of Edge Applications.
- Introduction to event driven architecture.
- What is it? How do you deploy and manage it at edge.
- Using open-source based event driven tools to host data collection and IoT processing on small devices feeding into K8s hosted applications and services.
- Demonstration using edge generated CloudEvents with a Kubernetes hosted application.
- How to get involved in the K8s IoT Edge Working Group.

Edge Applications

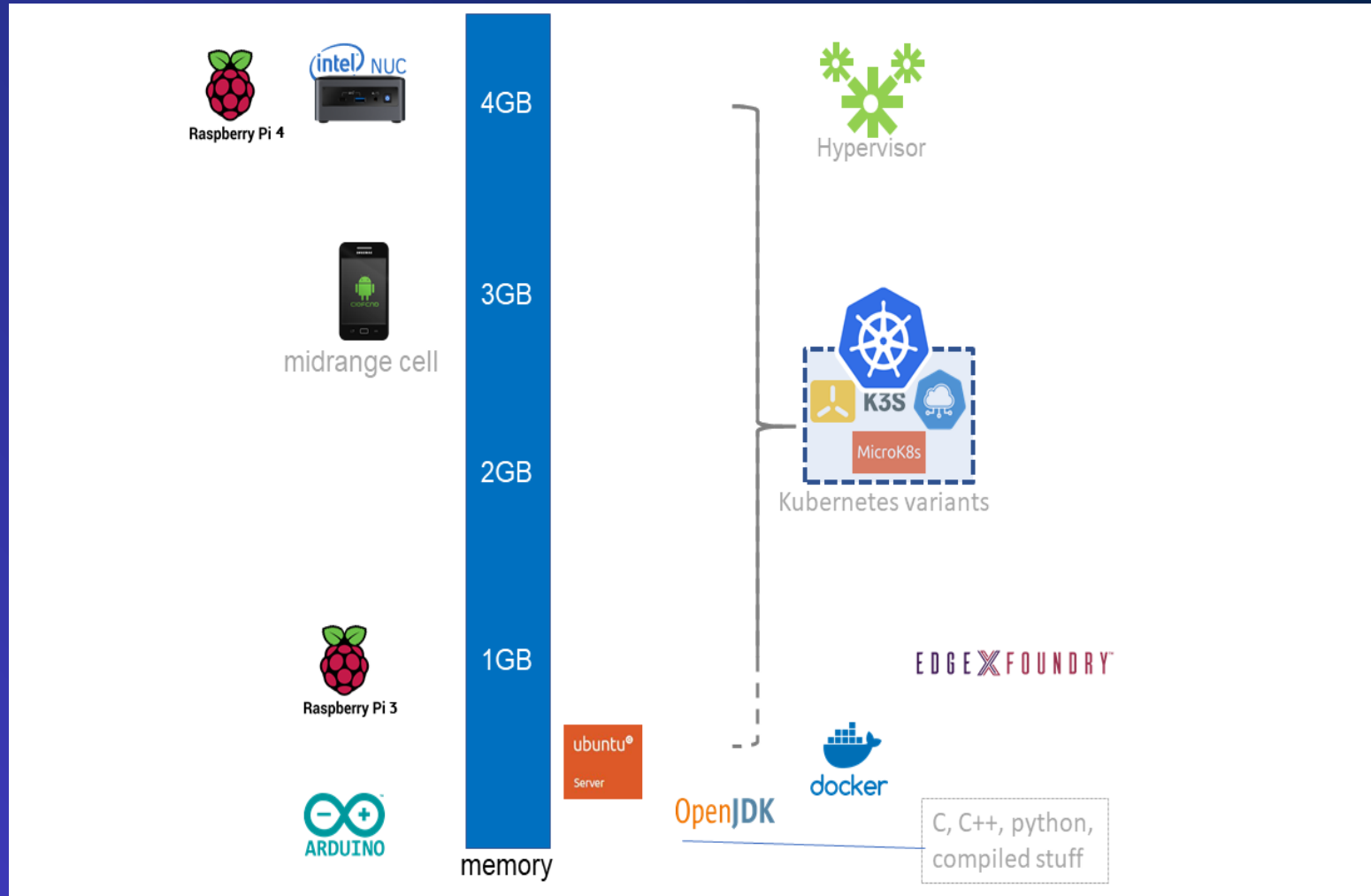
Characteristics and Challenges

Edge Applications might be isolated, but are usually going to be part of a “bigger picture” multi-tier architecture if Kubernetes is involved.



Edge Applications

Kubernetes can't run everywhere

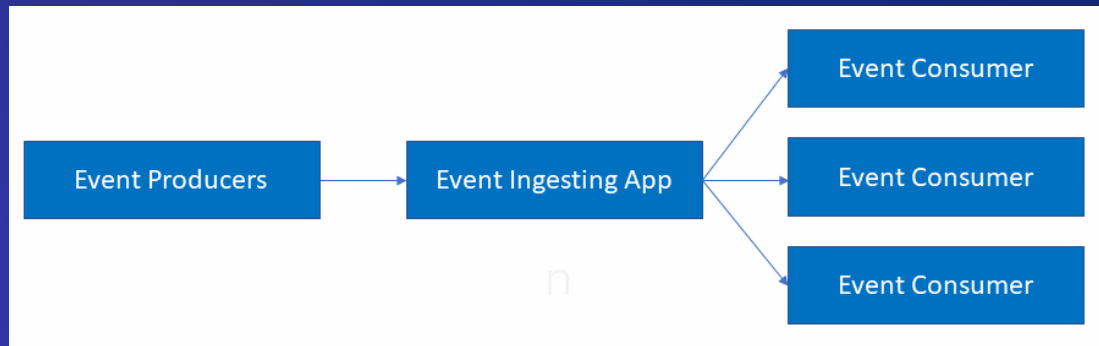


You can take steps to shrink K8s resource demands but there will always be a lower limit - some devices that can't even run Linux. Are you just going to declare the hard parts to be somebody else's problem?

Event Driven Architecture

Introduction

Event-driven architecture (EDA) is a software architecture paradigm promoting the production, detection, consumption of, and reaction to events. [Wikipedia](#)



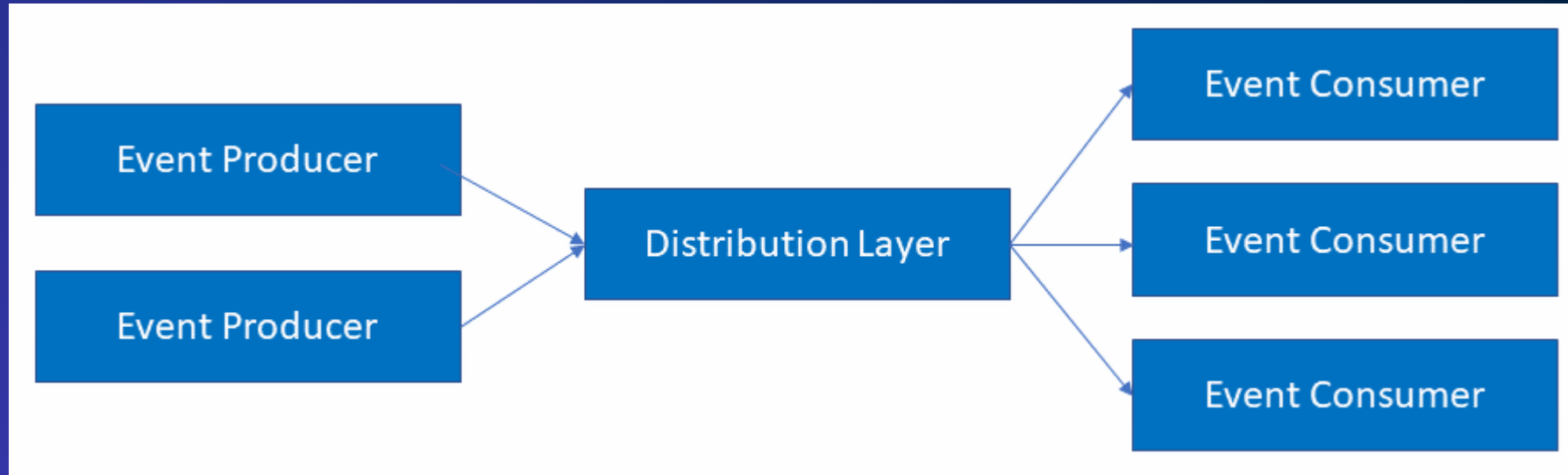
An event is a piece of information

It could be a fact - a measurement or a notice that something was detected

It could be a command - “I want the light turned on”

Serverless at edge

Optional distribution layer: router, queue, resilient storage, pub sub catalog

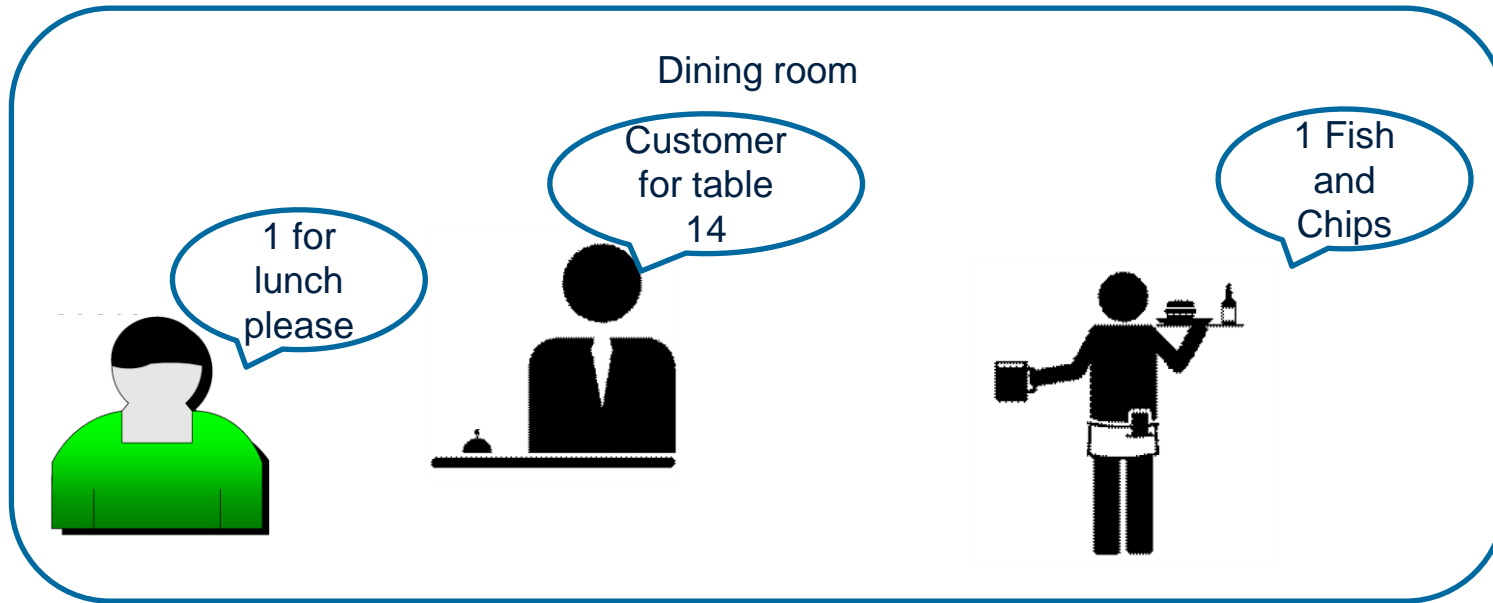
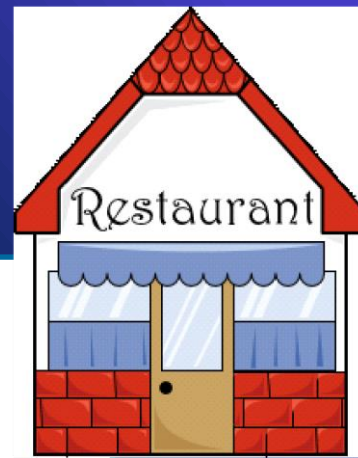


An event is a piece of information

It could be a fact - a measurement or a notice that something was detected

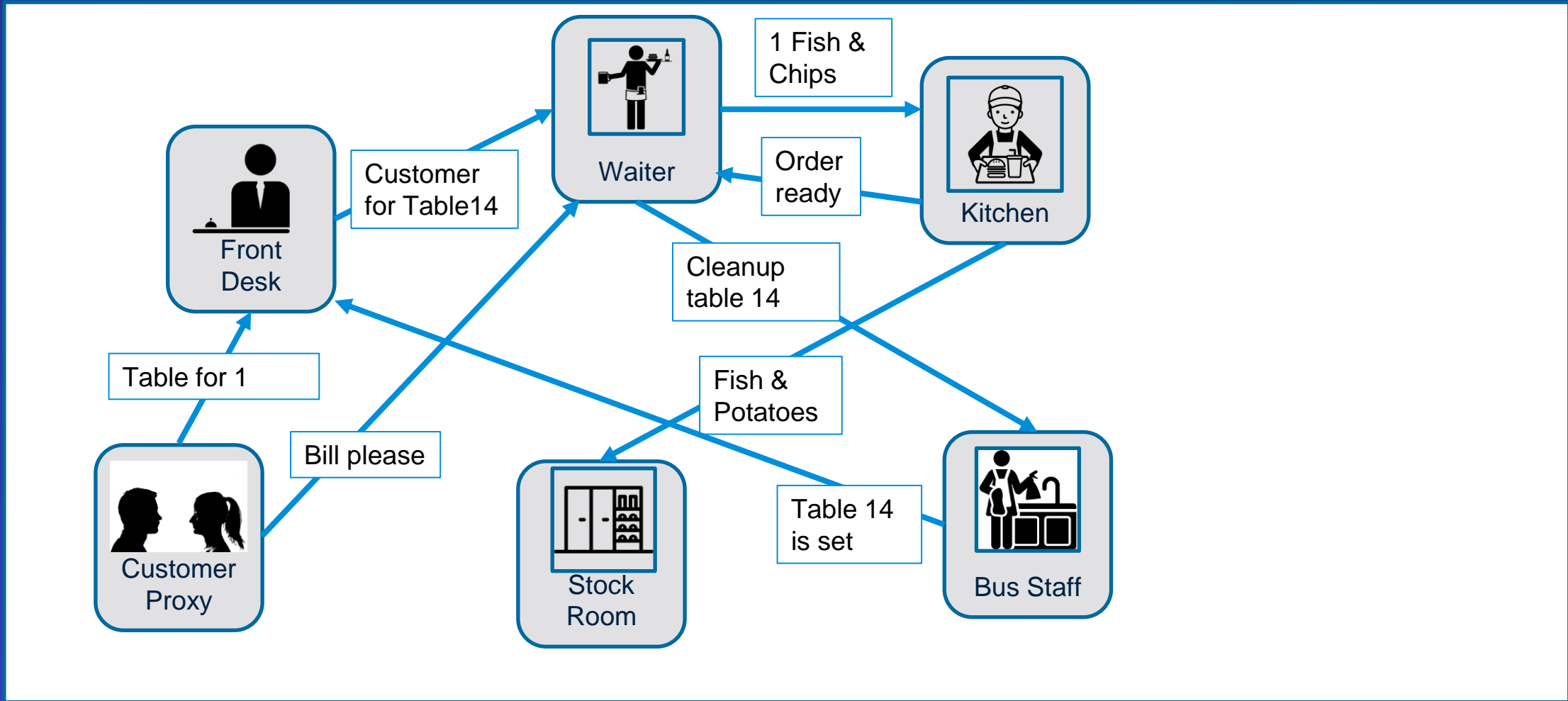
It could be a command - "I want the light turned on"

Event driven in a physical world context



Later:
"Order Up, Table 14"
"Clean up Table 14"

Is coding microservices a great way to handle this?



Event Driven can originate on devices below the “Kubernetes waterline”

Examples

Simple protocols like MQTT can be implemented on VERY low resource devices, particularly if you can afford to skip TLS.

NATS is also lighter weight.

AMQP came out of financial community and might have features you need at a resource cost you can still afford

Kafka, RabbitMQ and others might also be fits for your application – and you don’t have to pick just one. Higher level tools often has plugins from a variety of sources.

Event Driven advice

Frequently given patterns and anti-patterns

Event persistence: Events are persisted as a replayable stream history. Event consumers are not tied to the producer.

View an event is a record of something that has happened and so can't be changed.
(You can't change history.)

Messages on common delivery platforms can often deliver these characteristics - but can also be used in ways that are not events (messages targeted at specific recipients, request reply patterns, transient data deleted on read). Re-inventing RPC calls over a message bus sometimes doesn't end well on projects

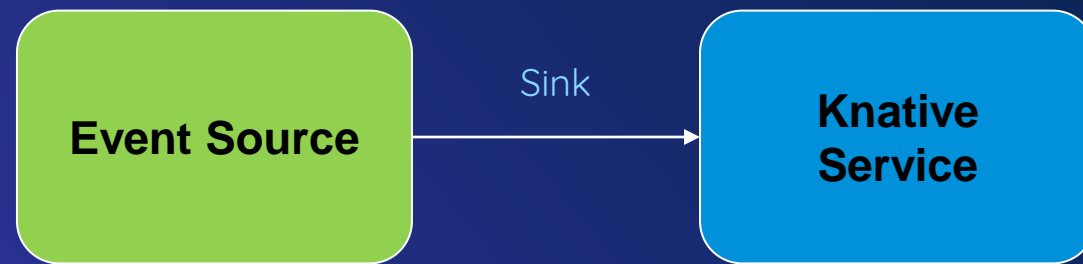
Rules are and should be different across latency, failure domain and context boundaries

favor asynchronous and eventual consistency across the boundaries

Synchronous might be OK within a boundary

Knative eventing

Sinks



CloudEvents

cloudevents.io

"A specification for describing event data in a common way"

```
Content-Type:
application/json
ce-specversion: 1.x-wip
ce-type: myevent
ce-id: 1234-1234-1234
ce-source: example.com
```

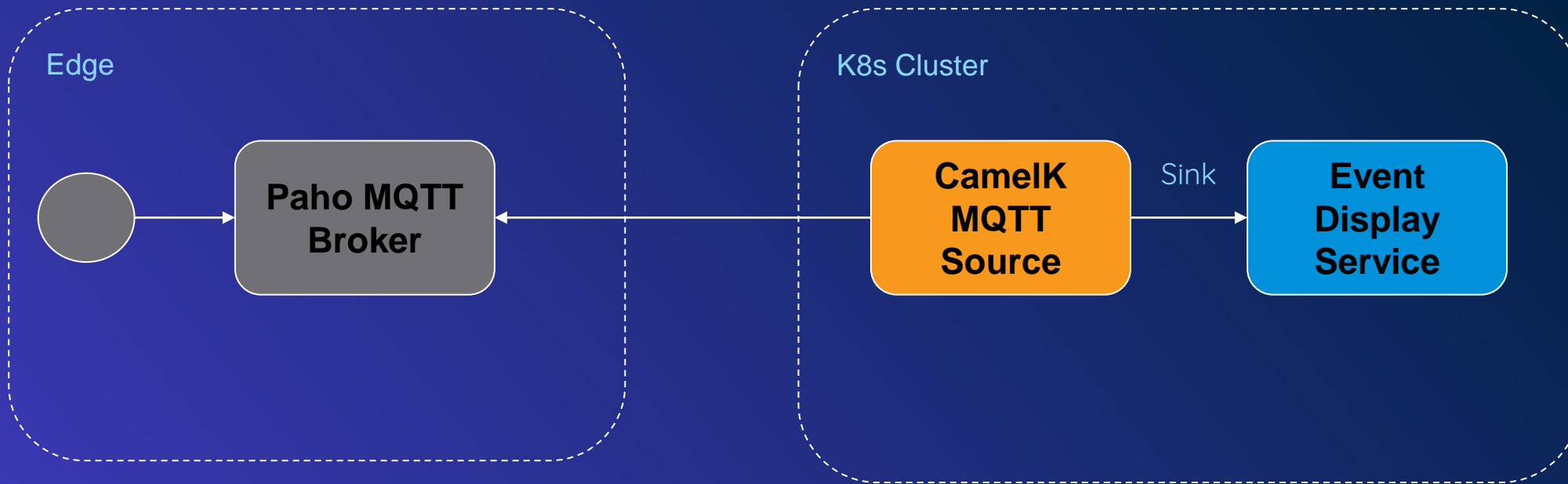
```
{
  "specversion": "1.x-wip",
  "type": "coolevent",
  "id": "xxxx-xxxx-xxxx",
  "source": "bigco.com",
  "data": { ... }
}
```

- Consistency
- Accessibility
- Portability

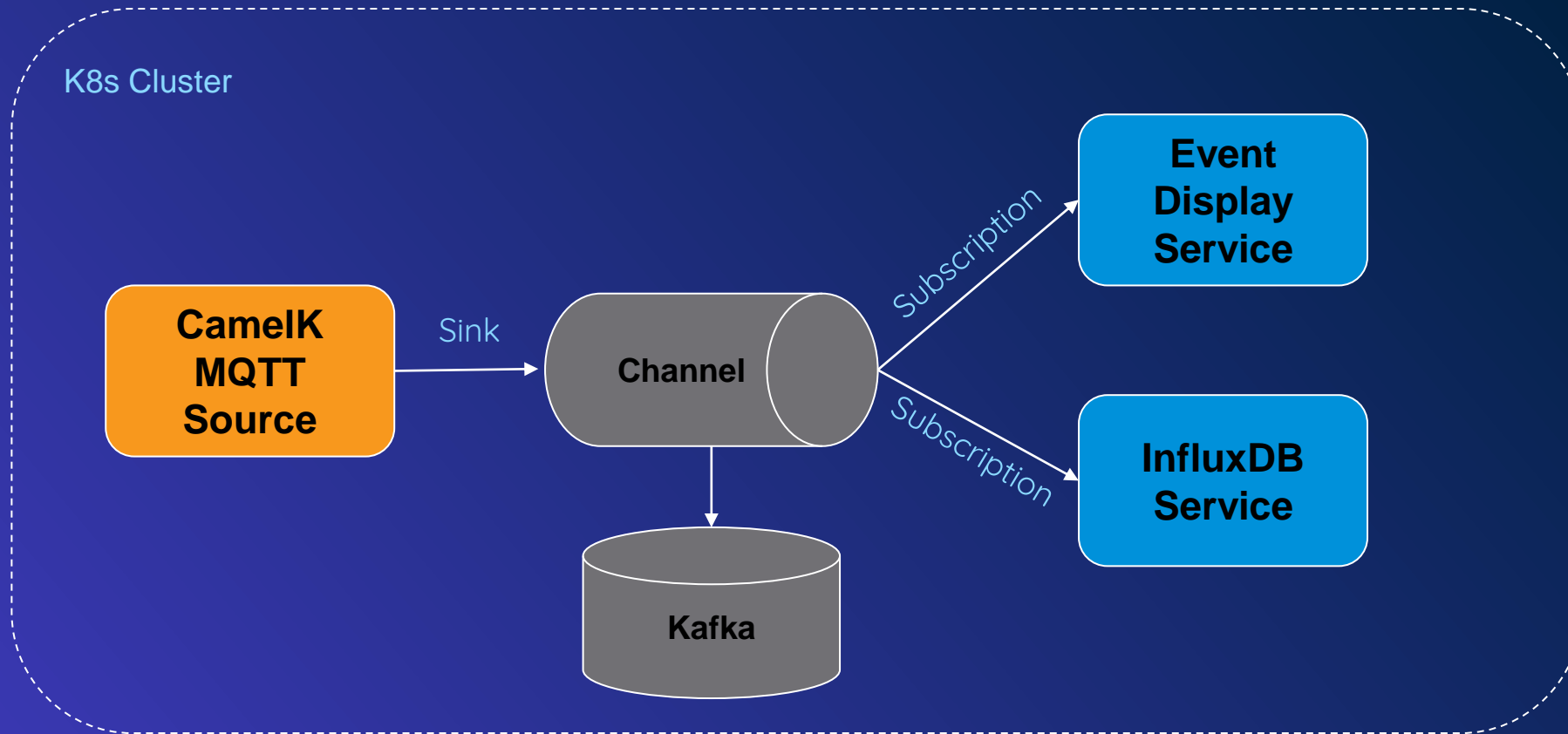


Demo

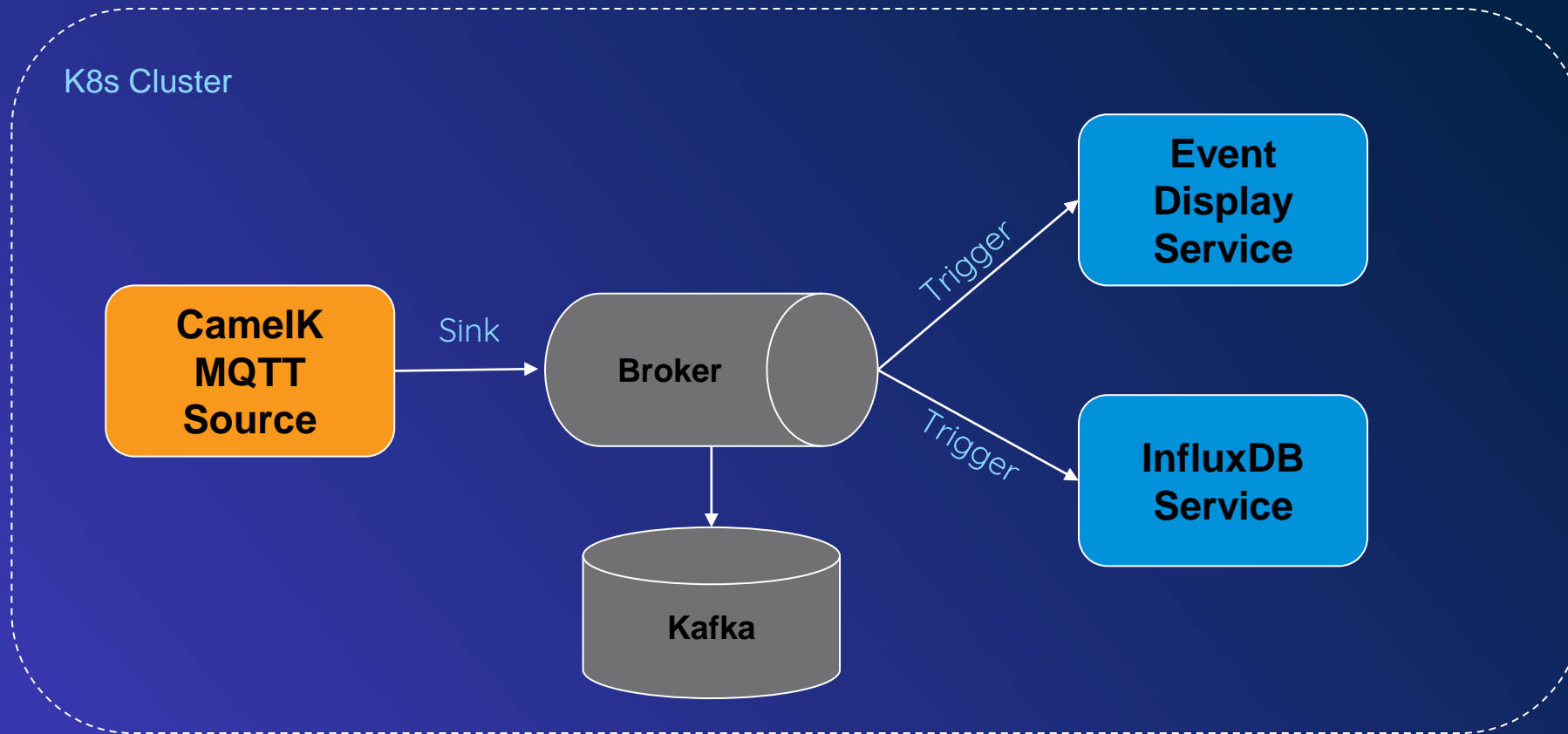
MQTT Source



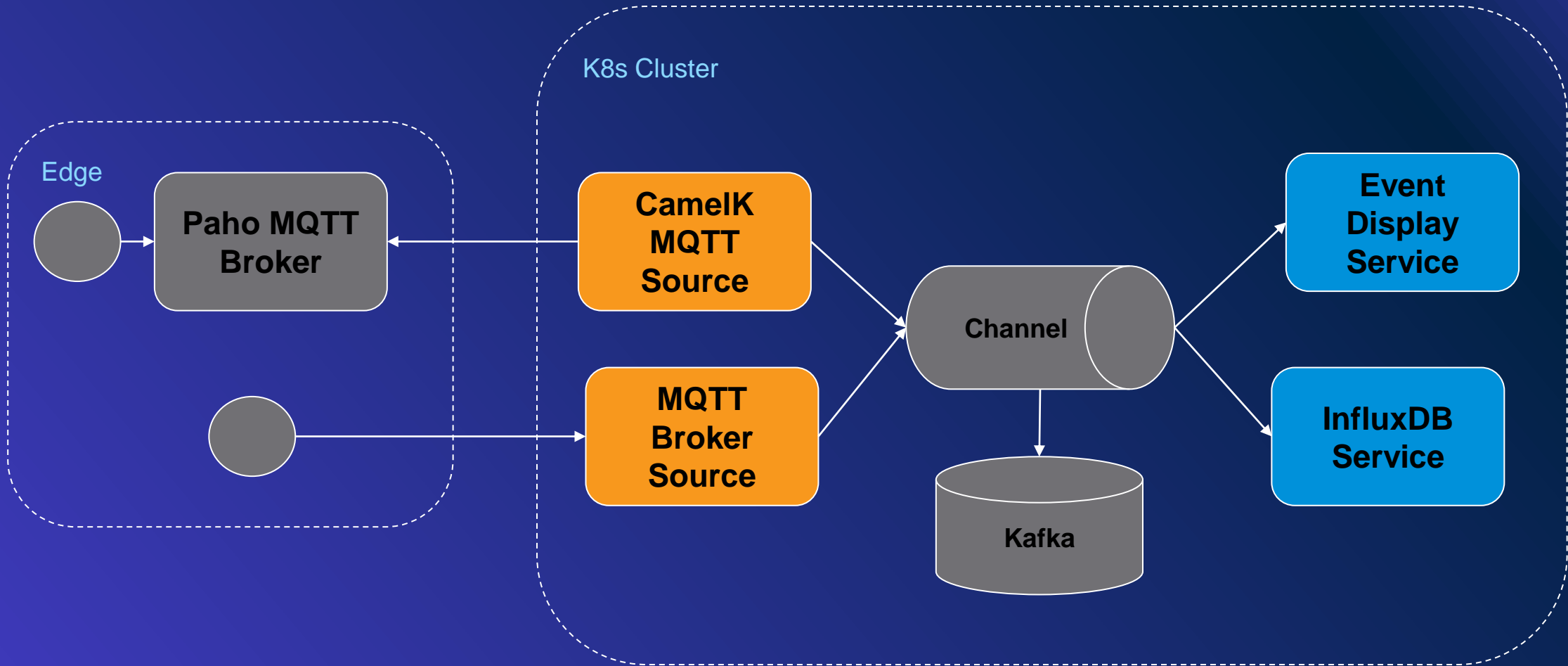
Knative Channels



Knative Brokers



Demo



To Do - get involved

Make MQTT first class citizen in the Knative world

- Figure out device security
- Figure out command and control

Integration with existing platforms, i.e. Eclipse Hono, Aws, Azure

Easy to way to install and manage components (CLIs, UIs, ...)

Commonly used services out of the box (Observability, Digital Twin, ...)

Extend clouds to the Edge using Skupper

- Virtual Application Networks for Cloud Native Applications - <https://sched.co/ekCc>

Open Source Communities – how to learn more + get involved

The Kubernetes IoT Edge Working Group + more

Regular Work Group Meeting:

USA WG Meeting Wednesday 9am PT, every 4 weeks, next on December 2

APAC WG meeting Wednesday 5 UTC every 4 weeks, next on December 16

- [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>

Speaker contact info

Deck link: <https://sched.co/ekH1>



Dejan Bosanac
Red Hat
@dejanb



Steve Wong
VMware
@cantbewong

Thank You

KEEP CLOUD NATIVE
EVERYWHERE



KubeCon



CloudNativeCon

North America 2020

Virtual

