



扫码添加小助手，发送“KubeEdge”加群

# CloudNativeLives

KubeEdge技术详解与实战

## KubeEdge快速上手与社区贡献实践

华为云原生团队核心成员 & CNCF社区主要贡献者倾力打造

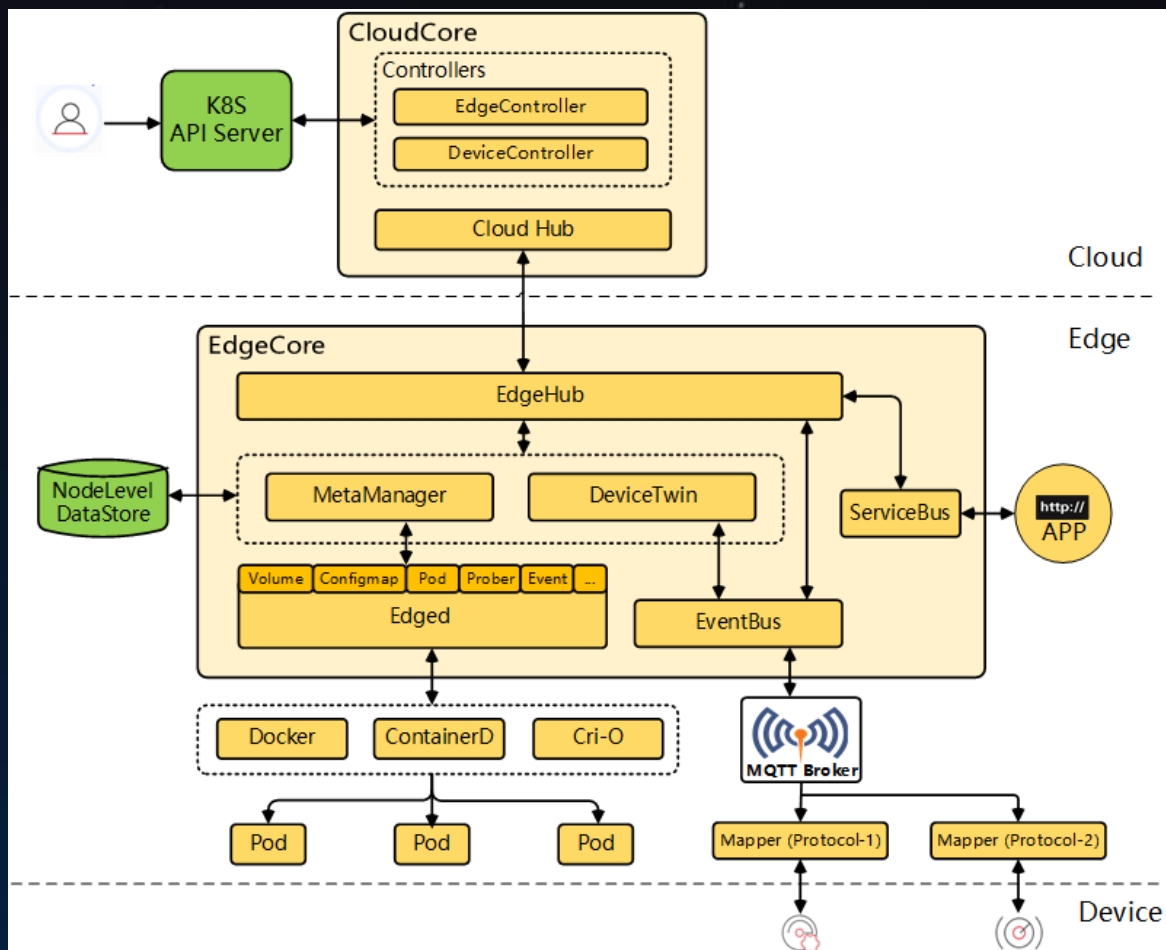


# 大纲

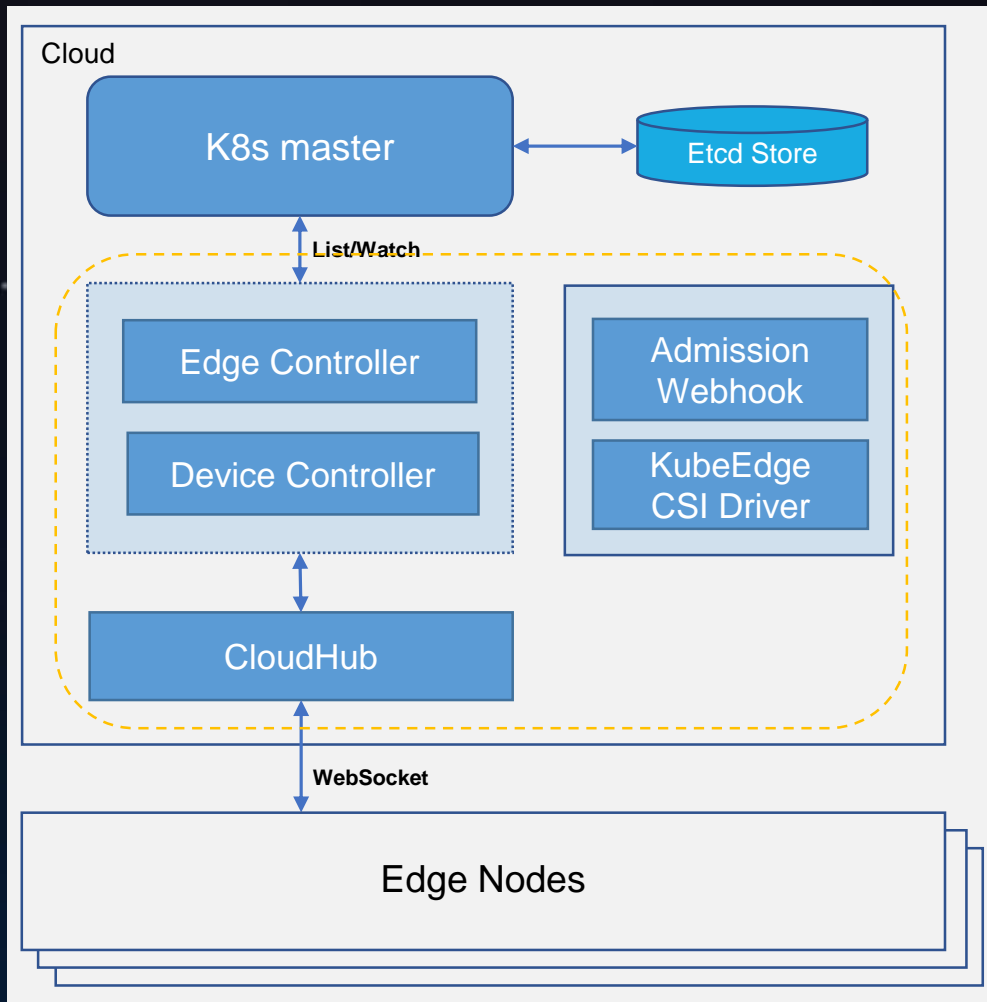
- KubeEdge的架构特点与优势
- KubeEdge 源码分析
- 运行一个KubeEdge集群
- 使用KubeEdge管理边缘应用与设备
- KubeEdge 社区开发快速上手

# KubeEdge架构

- 云边可靠协同
- 边缘极致轻量 ( ~70M )
- 边缘自治
- 边缘应用管理
- 边缘设备管理

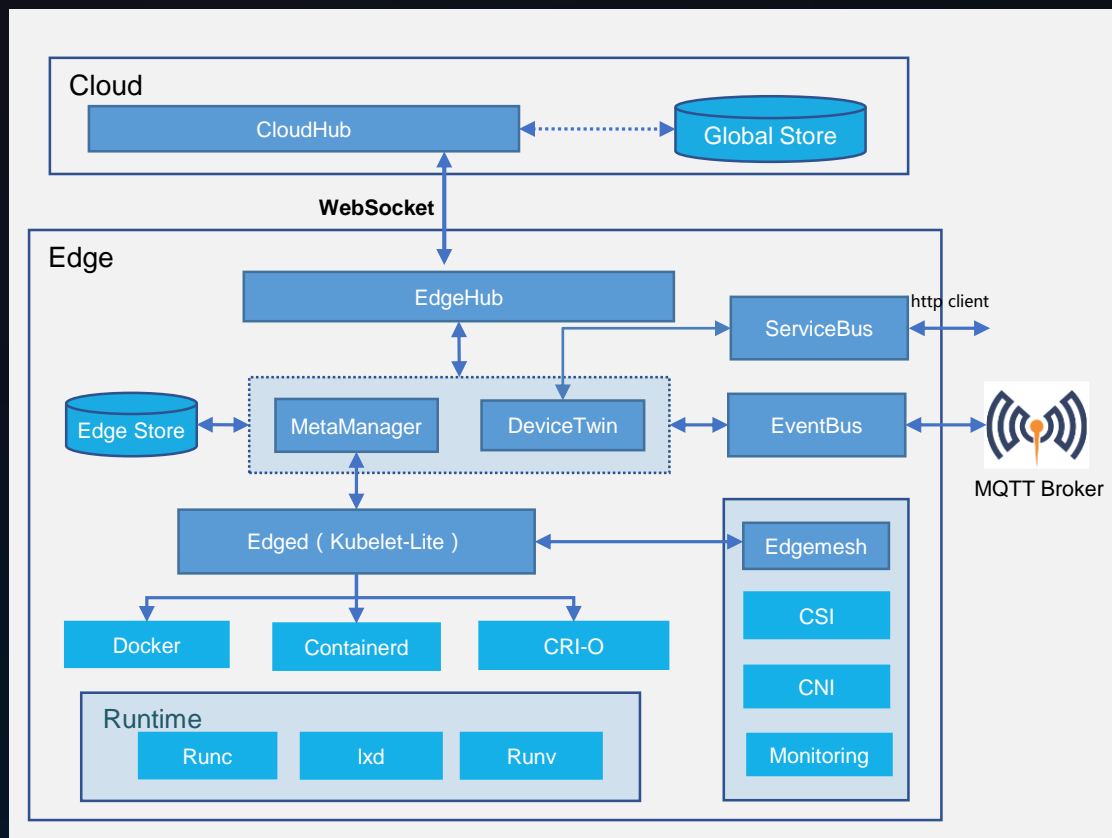


# KubeEdge云端组件



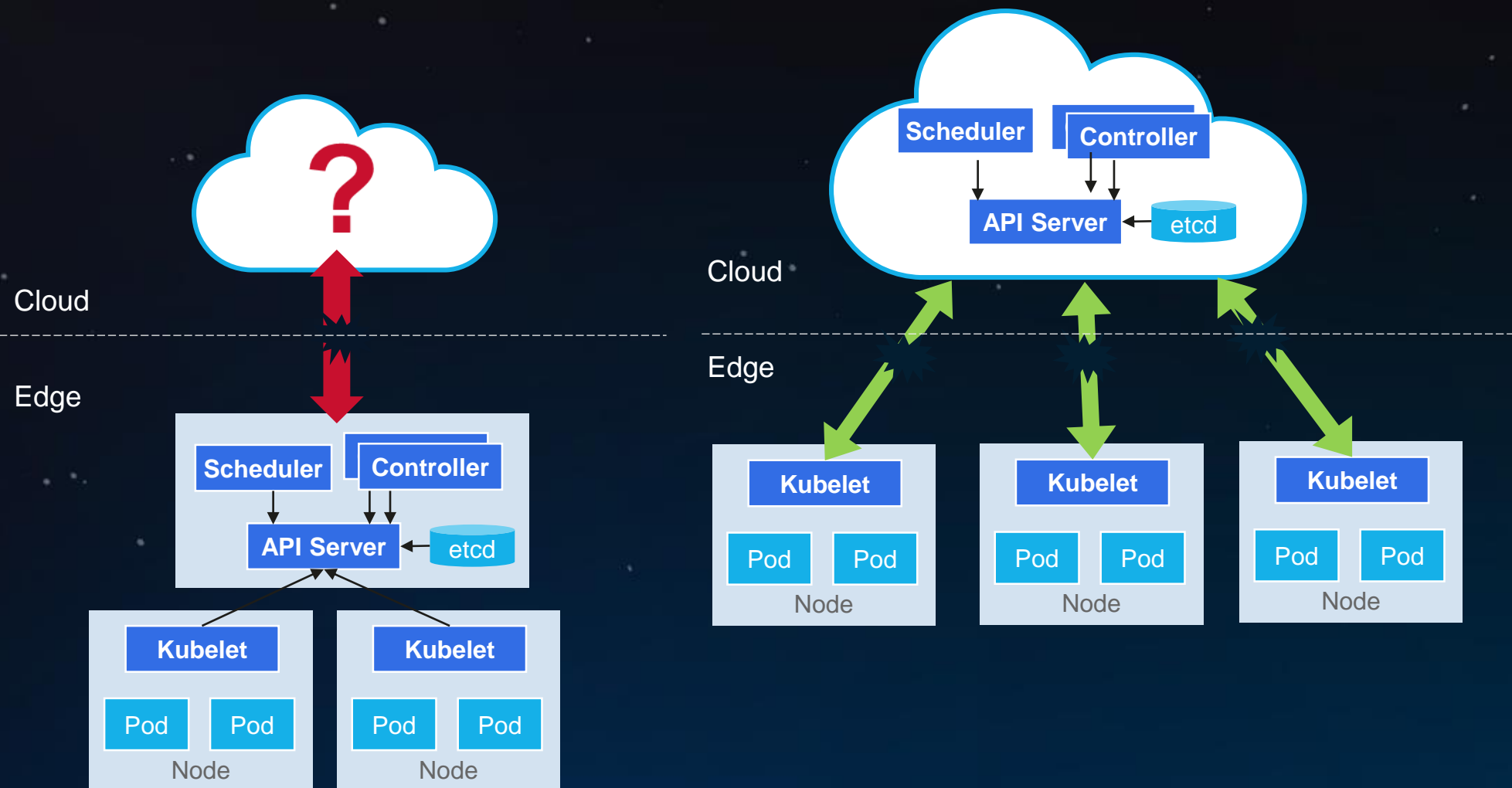
- EdgeController
  - 边缘节点管理
  - 应用状态元数据云边协同
- 设备抽象API/DeviceController
  - 接入和管理边缘设备
  - 设备元数据云边协同
- CSI Driver
  - 同步存储数据到边缘
- Admission Webhook
  - 扩展API合法性校验
  - Best Practice特性开关

# KubeEdge边缘组件



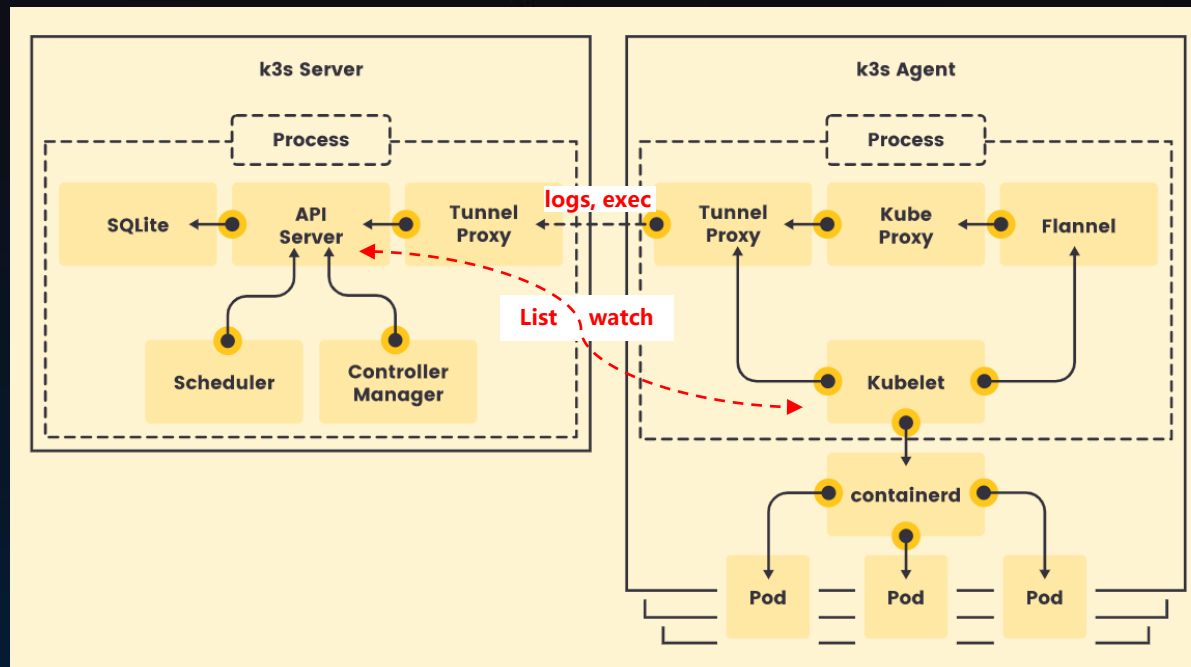
- EdgeHub
  - Messaging over WebSocket 提供可靠的云边信息同步
- MetaManager
  - 元数据本地持久化
- Edged
  - Kubelet-lite
  - 轻量化实现Pod生命周期
- DeviceTwin
  - 同步设备信息到云端
- EventBus
  - MQTT client
- ServiceBus
  - HTTP client

# 选择什么模式？



# 与K3S对比

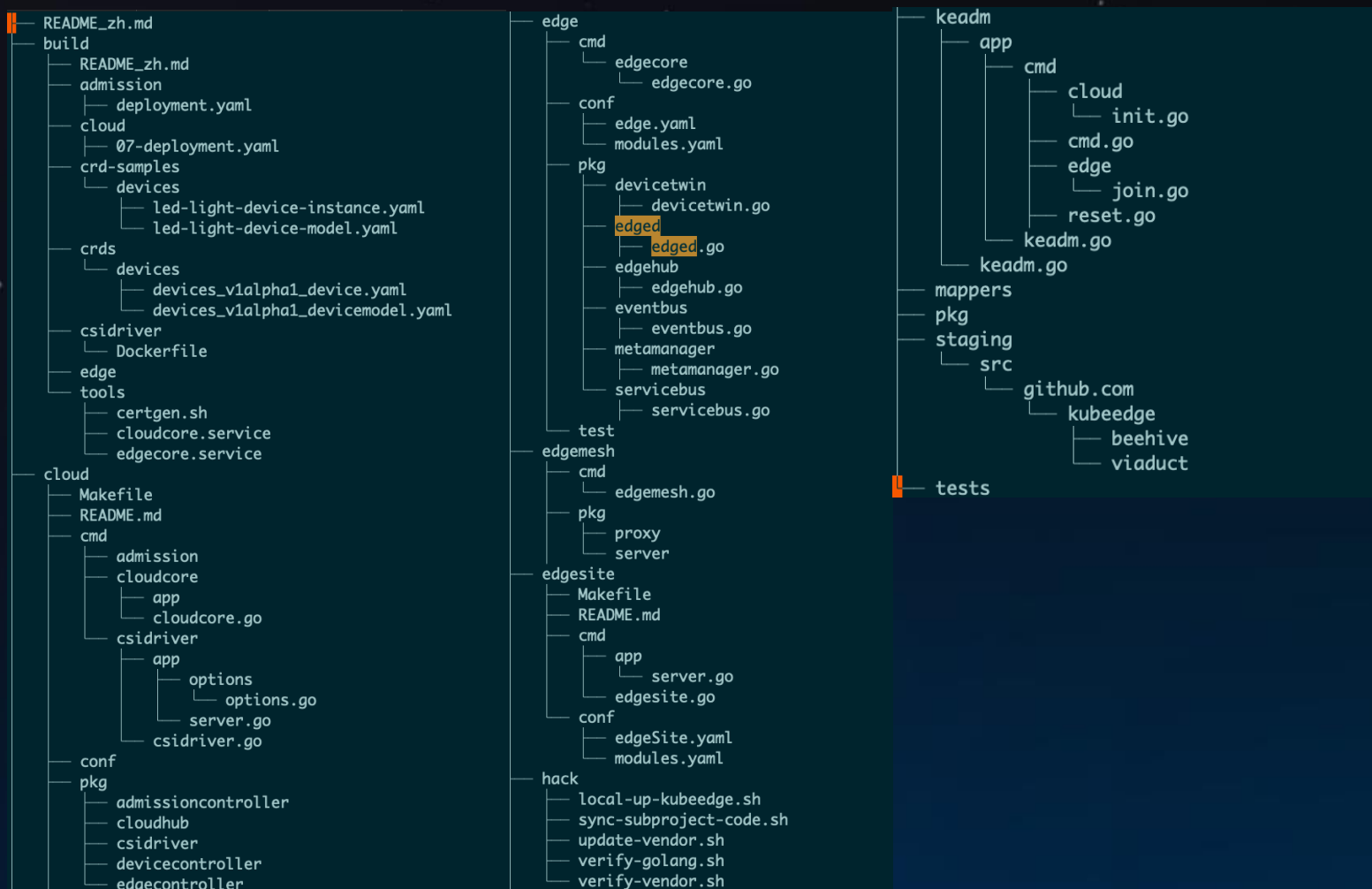
- 边缘端轻量化的K8s 集群
  - 删除 alpha features, in-tree storage plugins
  - Release with ARM build
  - Tunnel proxy for exec and logs APIs
  - Server 和 Agent 还是主要通过 List-watch 通信
- 资源占用率
  - K3s Server: 480M
  - K3s Agent: 120M
- 不具备云边可靠协同能力
- 只能运行在边缘端
- 社区
  - 由Rancher 开发和管理
  - 70+ 贡献者





- KubeEdge的架构特点与优势
- KubeEdge 源码分析
- 运行一个KubeEdge集群
- 使用KubeEdge管理边缘应用与设备
- KubeEdge 社区开发快速上手

# KubeEdge源码分析



- KubeEdge的架构特点与优势
- KubeEdge 源码分析
- 运行一个KubeEdge集群
- 使用KubeEdge管理边缘应用与设备
- KubeEdge 社区开发快速上手

# 一键部署单机版KubeEdge

```
#: cd kubeedge/kubeedge
```

```
#: hack/local-up-kubeedge.sh
```

```
#: export KUBECONFIG=/root/.kube/kind-config-test
```

```
#: kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
edge-node	Ready	edge	7m26s	v1.15.3-kubeedge-v1.1.0
test-control-plane	Ready	master	8m32s	v1.15.3

```
kubeedge git:(master) X hack/local-up-kubeedge.sh
checking kubect1
found kubect1, Client Version: v1.16.0-alpha.0.922+56b40066d55d08
checking kind
found kind, version: v0.5.1
building the cloudcore...
make: Entering directory '/root/codes/src/github.com/kubeedge/kubeedge'
cd cloud && make cloudcore
make[1]: Entering directory
'/root/codes/src/github.com/kubeedge/kubeedge/cloud'
bash ../hack/verify-golang.sh
go detail version: go version go1.12.9 linux/amd64
go version: 1.12.9
...
Local KubeEdge cluster is running. Press Ctrl-C to shut it down.
Logs:
/tmp/cloudcore.log
/tmp/edgecore.log

To start using your kubeedge, you can run:

export KUBECONFIG=/root/.kube/kind-config-test
kubectl get nodes
```

# KubeEdge 依赖关系

与K8s的兼容性列表：

	kubernetes 1.10	kubernetes 1.11	kubernetes 1.12	kubernetes 1.13	kubernetes 1.14	kubernetes 1.15	kubernetes 1.16
KubeEdge 1.0	✓	✓	✓	✓	✓	✓	—
KubeEdge 1.1	+	✓	✓	✓	✓	✓	✓
HEAD	+	✓	✓	✓	✓	✓	✓

与Golang的兼容性列表：

	Golang 1.10	Golang 1.11	Golang 1.12	Golang 1.13
KubeEdge 1.0	✓	✓	✓	✗
KubeEdge 1.1	✗	✗	✓	✗
HEAD	✗	✗	✓	TBD

- ✓ ：支持  
+ ： KubeEdge具有Kubernetes版本中可能不存在的功能或API对象。  
- ： Kubernetes 有 features or API 对象但是KubeEdge 没有  
TBD ： 待支持

# 源码编译安装KubeEdge

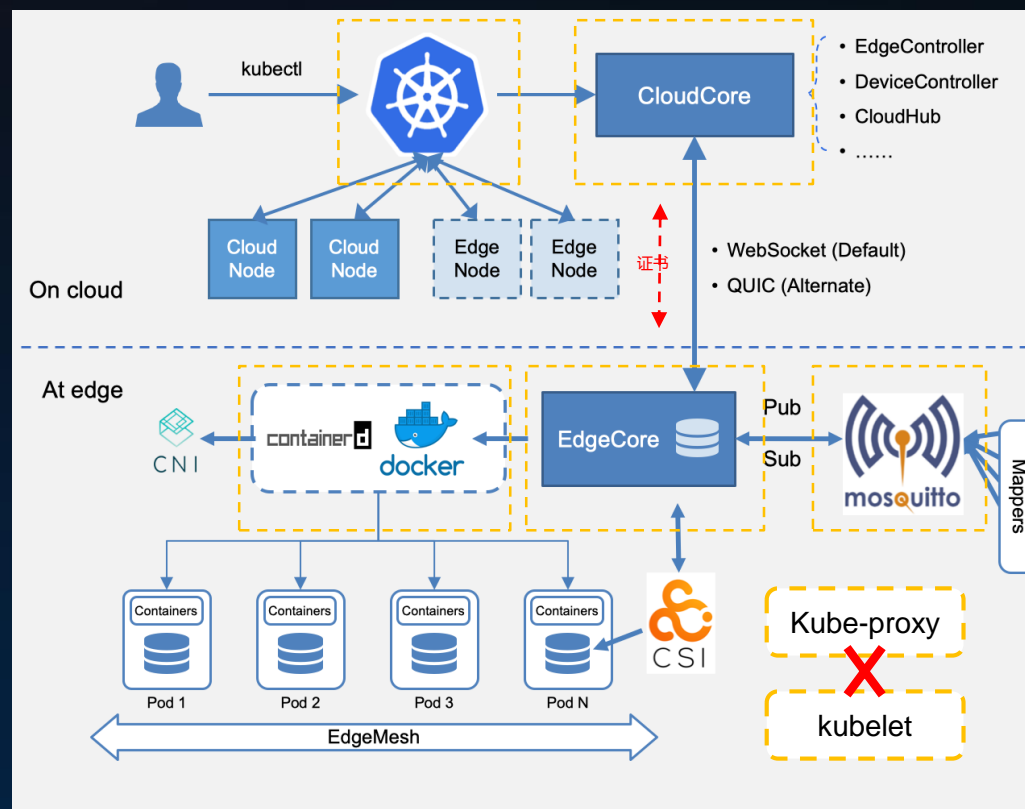
文档地址：<https://github.com/kubeedge/kubeedge/blob/master/docs/setup/setup.md>

云端

- golang 1.12
- Kubernetes
- Docker (Containerd)
- 证书
- CloudCore(配置文件)
- 部署CRD

边缘

- golang 1.12
- Docker (Containerd)
- 证书
- EdgeCore(配置文件)
- Mosquitto
- mappers



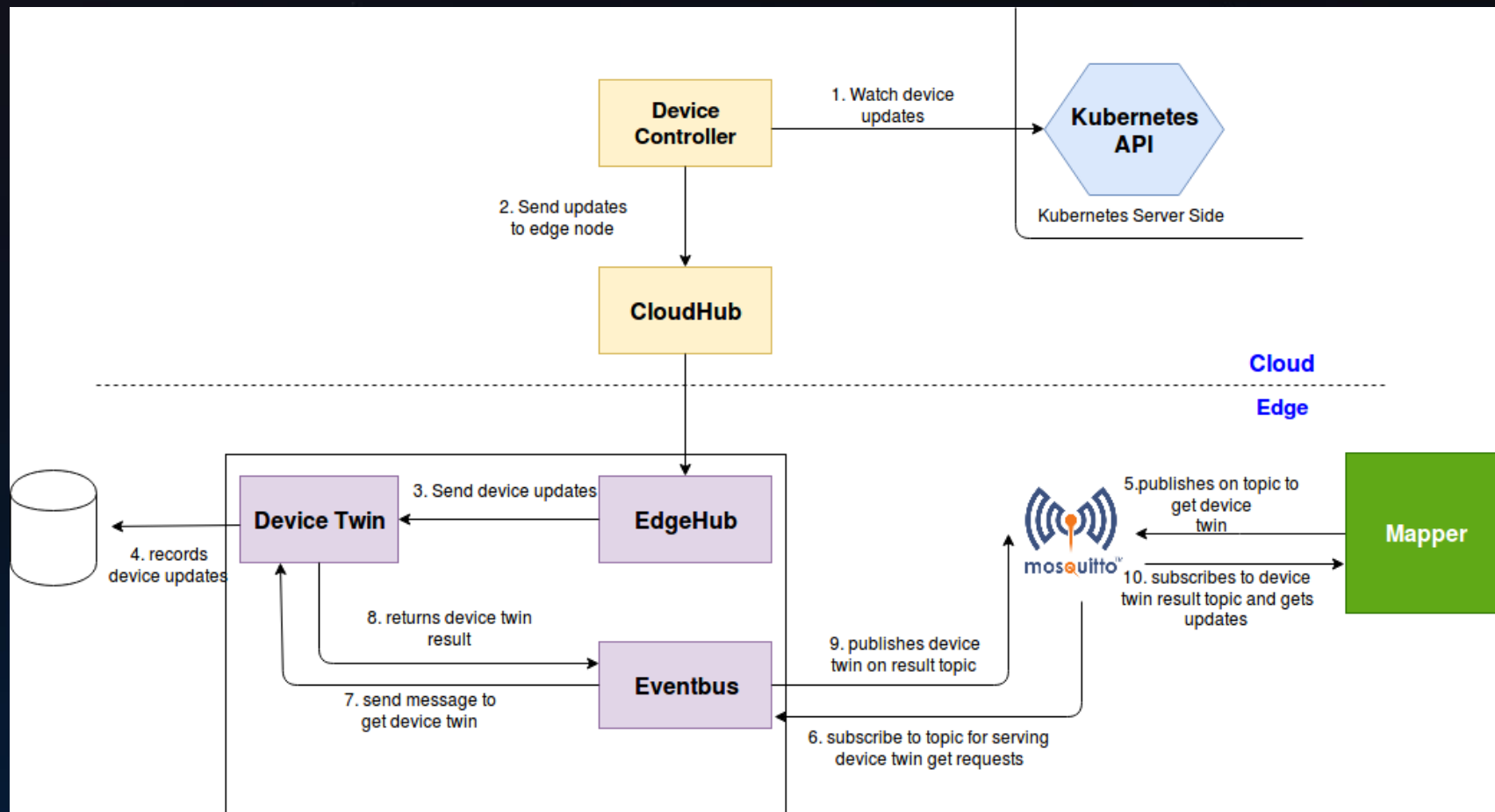
# Keadm部署KubeEdge

- 云端和边缘都安装Docker
- 云端安装Kubernetes
- 编译安装KubeEdge云端组件
  - keadm init
- 编译安装KubeEdge边缘侧组件
  - keadm join
- 社区正讨论重新定义keadm 的工作职责
- kubeadm 负责安装kubernetes 集群
- keadm 只负责KubeEdge组件的安装和环境检查
- 欢迎感兴趣的同学参与贡献

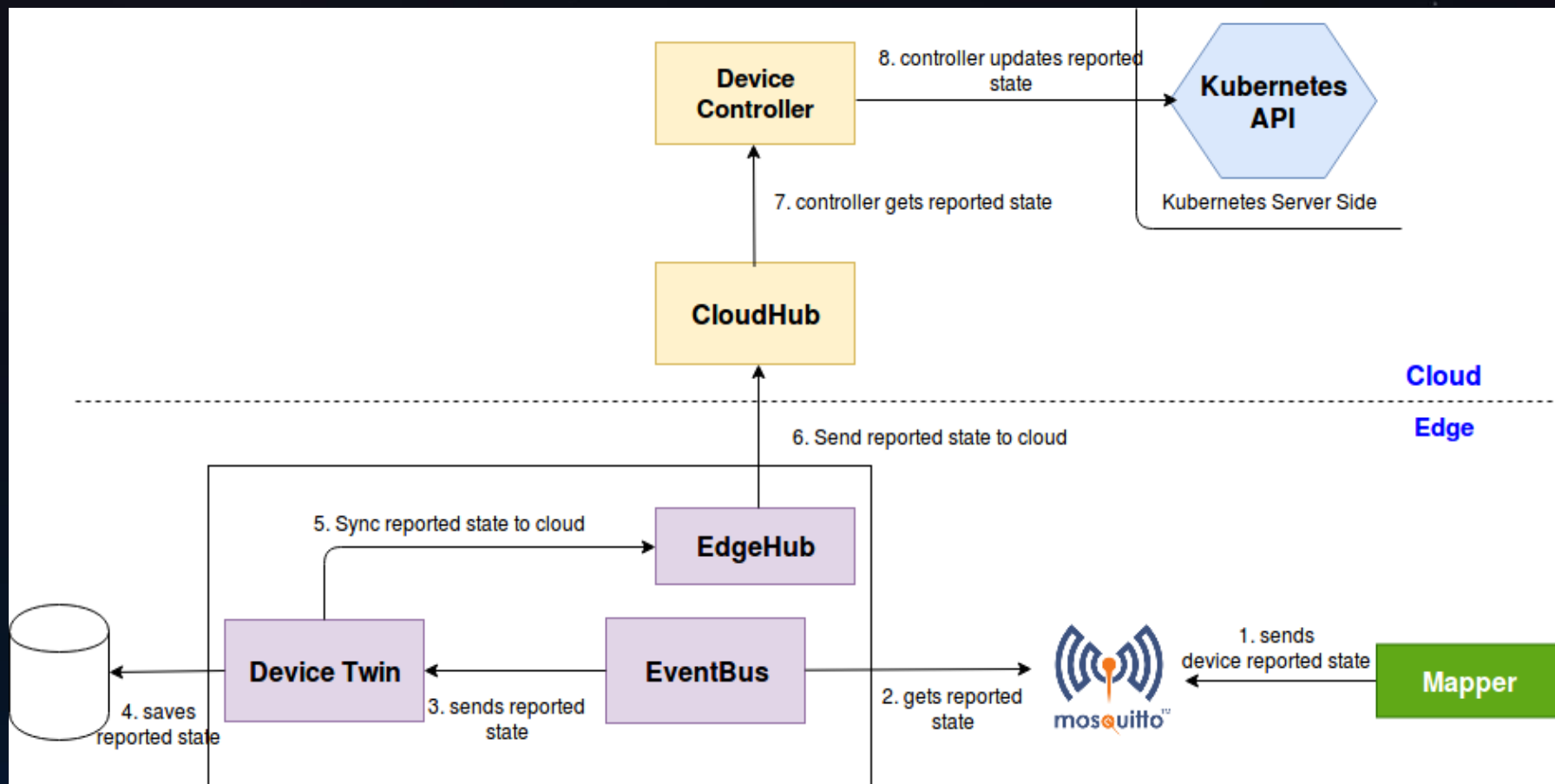
- KubeEdge的架构特点与优势
- KubeEdge 源码分析
- 运行一个KubeEdge集群
- 使用KubeEdge管理边缘应用与设备
- KubeEdge 社区开发快速上手



# 云端同步期望状态到边缘



# 边缘上报设备状态到云端



# KubeEdge管理树莓派交通灯



```
root@192.168.4.22-pi2:-# pinout
```

```
Revision          : b03111
SoC               : BCM2711
RAM              : 2048Mb
Storage          : MicroSD
USB ports        : 4 (excluding power)
Ethernet ports   : 1
Wi-fi            : True
Bluetooth        : True
Camera ports (CSI) : 1
Display ports (DSI) : 1
```

J8:

	3V3	(1)	(2)	SV
GPI02	(3)	(4)	SV	
GPI03	(5)	(6)	GND	
GPI04	(7)	(8)	GPI014	
GND	(9)	(10)	GPI015	
GPI017	(11)	(12)	GPI018	
GPI027	(13)	(14)	GND	
GPI022	(15)	(16)	GPI023	
3V3	(17)	(18)	GPI024	
GPI010	(19)	(20)	GND	
GPI09	(21)	(22)	GPI025	
GPI011	(23)	(24)	GPI08	
GND	(25)	(26)	GPI07	
GPI00	(27)	(28)	GPI01	
GPI05	(29)	(30)	GND	
GPI06	(31)	(32)	GPI012	
GPI013	(33)	(34)	GND	
GPI019	(35)	(36)	GPI016	
GPI026	(37)	(38)	GPI020	
GND	(39)	(40)	GPI021	

```

root@192.168.4.22-pi2:~# gpio readall
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| BCM | wPi |   Name   | Mode | V | Physical | V | Mode |   Name   | wPi | BCM |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|      |      | 3.3v     |      |   | 1 || 2 |      | 5v       |      |      | |
|  2   |  8   | SDA.1     | IN    | 1 | 3 || 4 |      | 5v       |      |      |
|  3   |  9   | SCL.1     | IN    | 1 | 5 || 6 |      | 0v       |      |      |
|  4   |  7   | GPIO. 7   | IN    | 1 | 7 || 8 |  1   | IN       |      |      |
|      |      | 0v        |      |   | 9 ||10 |  1   | IN       | RxD      | 15   | 14   |
| 17   |  0   | GPIO. 0   | IN    | 0 |11 ||12 |  0   | IN       | GPIO. 1  | 16   | 15   |
| 27   |  2   | GPIO. 2   | IN    | 0 |13 ||14 |      | 0v       | GPIO. 1  | 1     | 18   |
| 22   |  3   | GPIO. 3   | IN    | 0 |15 ||16 |  0   | IN       | GPIO. 4  | 4     | 23   |
|      |      | 3.3v     |      |   |17 ||18 |  0   | IN       | GPIO. 5  | 5     | 24   |
| 10   | 12   | MOSI      | IN    | 0 |19 ||20 |      | 0v       |          |      |      |
|  9   | 13   | MISO      | IN    | 0 |21 ||22 |  0   | IN       | GPIO. 6  | 6     | 25   |
| 11   | 14   | SCLK      | IN    | 0 |23 ||24 |  1   | IN       | CE0      |10     | 8     |
|      |      | 0v        |      |   |25 ||26 |  1   | IN       | CE1      |11     | 7     |
|  0   | 30   | SDA.0     | IN    | 1 |27 ||28 |  1   | IN       | SCL.0    |31     | 1     |
|  5   | 21   | GPIO.21   | IN    | 1 |29 ||30 |      | 0v       |          |      |      |
|  6   | 22   | GPIO.22   | IN    | 1 |31 ||32 |  0   | IN       | GPIO.26  |26     | 12   |
| 13   | 23   | GPIO.23   | OUT   | 0 |33 ||34 |      | 0v       |          |      |      |
| 19   | 24   | GPIO.24   | OUT   | 0 |35 ||36 |  0   | IN       | GPIO.27  |27     | 16   |
| 26   | 25   | GPIO.25   | OUT   | 0 |37 ||38 |  0   | IN       | GPIO.28  |28     | 20   |
|      |      | 0v        |      |   |39 ||40 |  0   | IN       | GPIO.29  |29     | 21   |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| BCM | wPi |   Name   | Mode | V | Physical | V | Mode |   Name   | wPi | BCM |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

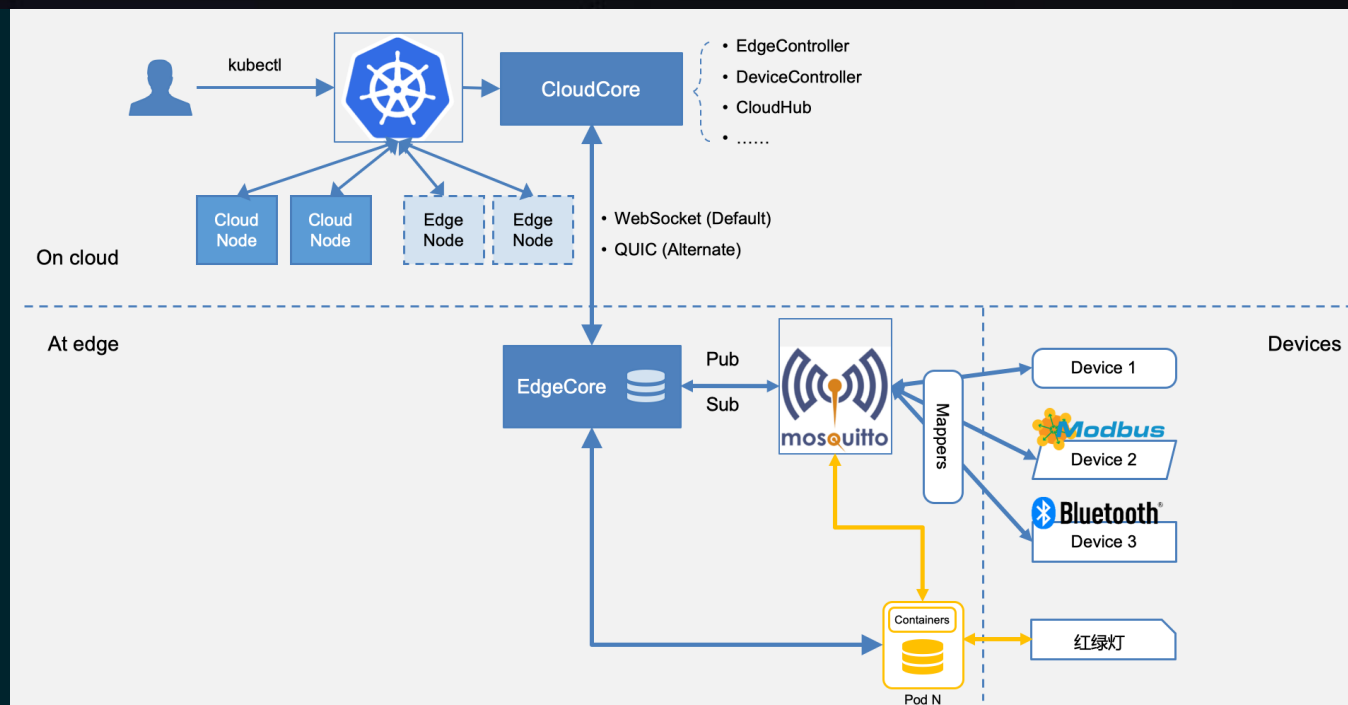
# 建立交通灯模型

```
apiVersion: devices.kubeedge.io/v1alpha1
kind: DeviceModel
metadata:
  name: traffic-light
  namespace: default
spec:
  properties:
    - description: Indicates whether the led light is ON/OFF
      name: red
      type:
        string:
          accessMode: ReadWrite
          defaultValue: "OFF"
    - description: Indicates whether the yellow light is ON/OFF
      name: yellow
      type:
        string:
          accessMode: ReadWrite
          defaultValue: "OFF"
    - description: Indicates whether the green light is ON/OFF
      name: green
      type:
        string:
          accessMode: ReadWrite
          defaultValue: "OFF"
    - description: Indicates red led pin munber
      name: red-pin-number
      type:
        int:
          accessMode: ReadOnly
          defaultValue: 25
    - description: Indicates green led pin munber
      name: green-pin-number
      type:
        int:
          accessMode: ReadOnly
          defaultValue: 23
    - description: Indicates rellow led pin munber
      name: yellow-pin-number
      type:
        int:
          accessMode: ReadOnly
          defaultValue: 24
```

```
apiVersion: devices.kubeedge.io/v1alpha1
kind: Device
metadata:
  labels:
    description: Light
    model: traffic-light
  name: traffic-light-instance-01
  namespace: default
spec:
  deviceModelRef:
    name: traffic-light
  nodeSelector:
    nodeSelectorTerms:
      - matchExpressions:
          - key: hostname
            operator: In
            values:
              - 192.168.4.22-pi2
  status:
    twins:
      - desired:
          metadata:
            type: string
            value: "OFF"
          propertyName: red
          reported:
            metadata:
              type: string
              value: "OFF"
        - desired:
            metadata:
              type: string
              value: "OFF"
            propertyName: yellow
            reported:
              metadata:
                type: string
                value: "OFF"
        - desired:
            metadata:
              type: string
              value: "OFF"
            propertyName: green
            reported:
              metadata:
                type: string
                value: "OFF"
```

# 部署应用

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: traffic-light
spec:
  replicas: 1
  selector:
    matchLabels:
      app: traffic-light
  template:
    metadata:
      labels:
        app: traffic-light
    spec:
      nodeSelector:
        hostname: 192.168.4.22-pi2
      hostNetwork: true
      containers:
        - name: light
          image: jiulongzaitian/traffic-light:v1.1
          command: ["/traffic/light"]
          args: ["--device=traffic-light-instance-01"]
          imagePullPolicy: Always
          securityContext:
            privileged: true
          volumeMounts:
            - name: config-volume
              mountPath: /opt/kubeedge/
      volumes:
        - name: config-volume
          configMap:
            name: device-profile-config-192.168.4.22-pi2
      restartPolicy: Always
```



# 开始演示

# KubeEdge管理空气净化器



- 内置MQTT
- 夜间模式，摇头，风速，定时功能
- 查看滤芯寿命，空气质量，湿度，温度，粉尘等功能

# 建立空气净化器设备模型

```
apiVersion: devices.kubedge.io/v1alpha1
kind: DeviceModel
metadata:
  name: dysonfan
  namespace: default
spec:
  properties:
    - name: NightMode
      description: Night Model ON/OFF
      type:
        string:
          accessMode: ReadWrite
          defaultValue: 'OFF'
    - name: Speed
      description: Fan Speed
      type:
        string:
          accessMode: ReadWrite
          defaultValue: '0002'
    - name: Oscillation
      description: oscillation ON/OFF
      type:
        string:
          accessMode: ReadWrite
          defaultValue: 'OFF'
    - name: FilterLife
      description: Filter lift
      type:
        string:
          accessMode: ReadOnly
          defaultValue: '1685'
    - name: Humidity
      description: humidity
      type:
        string:
          accessMode: ReadOnly
          defaultValue: '30'
    - name: AQI
      description: Air quality index
      type:
        string:
          accessMode: ReadOnly
          defaultValue: '0'
    - name: Dust
      description: dust
      type:
        string:
          accessMode: ReadOnly
          defaultValue: '5'
    - name: Temperature
      description: temperature
      type:
        string:
          accessMode: ReadOnly
          defaultValue: '24'
```

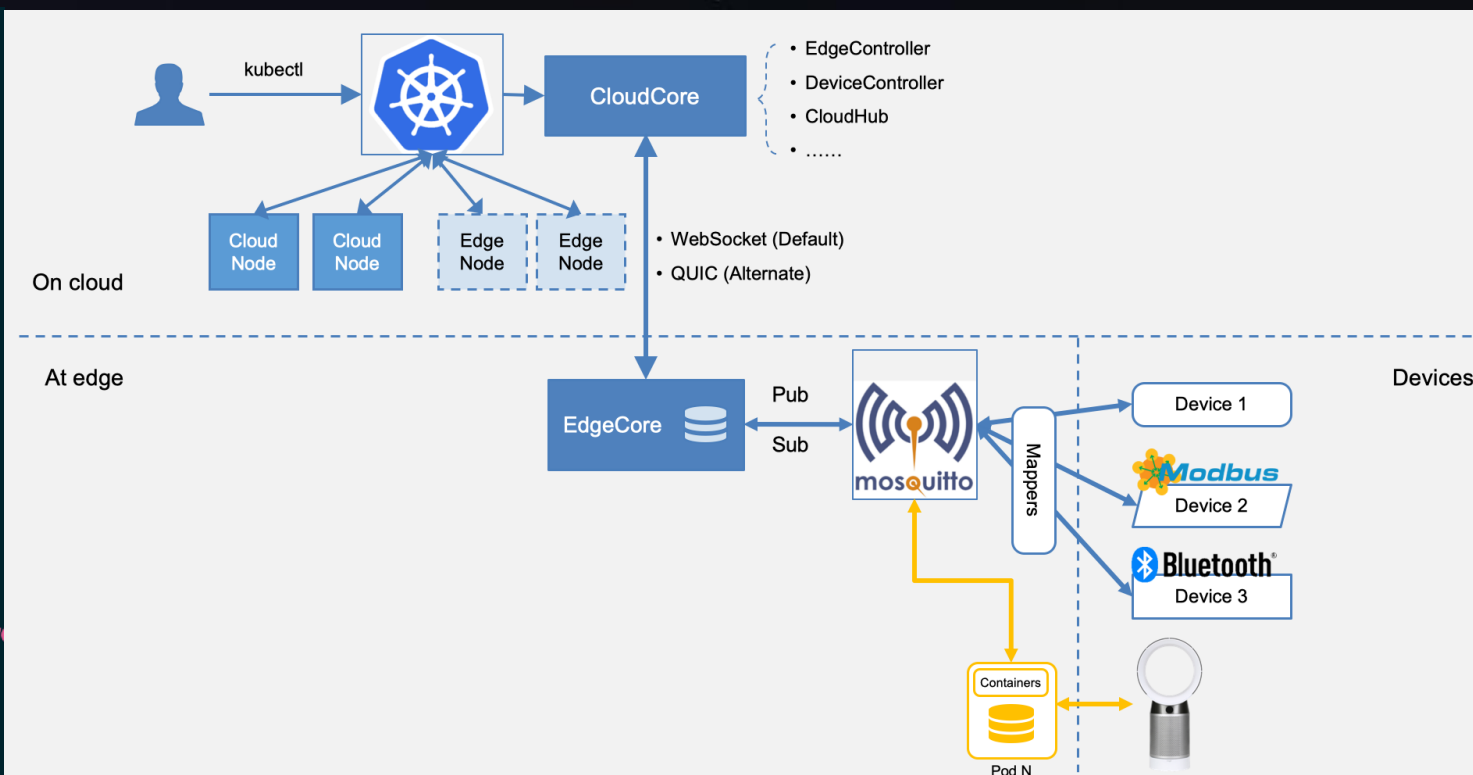
```
apiVersion: devices.kubedge.io/v1alpha1
kind: Device
metadata:
  labels:
    description: fan
    model: dysonfan
    name: dysonfan-instance-01
    namespace: default
spec:
  deviceModelRef:
    name: dysonfan
  nodeSelector:
    nodeSelectorTerms:
      - matchExpressions:
          - key: hostname
            operator: In
            values:
              - 192.168.4.22-pi2
status:
  twins:
    - propertyName: NightMode
      desired:
        metadata:
          type: string
          value: 'OFF'
      reported:
        metadata:
          type: string
          value: 'OFF'
    - propertyName: Speed
      desired:
        metadata:
          type: string
          value: '0002'
      reported:
        metadata:
          type: string
          value: '0002'
    - propertyName: Oscillation
      desired:
        metadata:
          type: string
          value: 'OFF'
      reported:
        metadata:
          type: string
          value: 'OFF'
    - propertyName: FilterLife
      reported:
        metadata:
          type: string
          value: '1685'
    - propertyName: Humidity
      reported:
        metadata:
```

```
type: string
value: '30'
- propertyName: AQI
  reported:
    metadata:
      type: string
      value: '0 '
- propertyName: Dust
  reported:
    metadata:
      type: string
      value: '5'
- propertyName: Temperature
  reported:
    metadata:
      type: string
      value: '24'
```



# 部署应用

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: dyson
spec:
  replicas: 1
  selector:
    matchLabels:
      app: dyson
  template:
    metadata:
      labels:
        app: dyson
    spec:
      hostNetwork: true
      containers:
        - name: light
          image: jiulongzaitian/dysonfan:v1.1
          command: ["/dyson/dysonFan"]
          args: ["-u", "your_account", "-p", "y
          imagePullPolicy: Always
          securityContext:
            privileged: true
          restartPolicy: Always
```



# 开始演示

- KubeEdge的架构特点与优势
- KubeEdge 源码分析
- 运行一个KubeEdge集群
- 使用KubeEdge管理边缘应用与设备
- KubeEdge 社区开发快速上手

# 社区开发快速上手 —— 参与社区

## 参与方式：

- 补充社区文档
- 补充测试用例
- 发现问题并提一个Issue
- 修复Issue
  - Help wanted
  - Good first issue
  - Bug
- 设计新特性
  - 写Proposal
  - 写特性代码

## 社区交流：

- Slack
- Mailing list
- 微信群

## 社区例会：

- 每双周举行社区例会

<https://github.com/kubeedge/kubeedge>

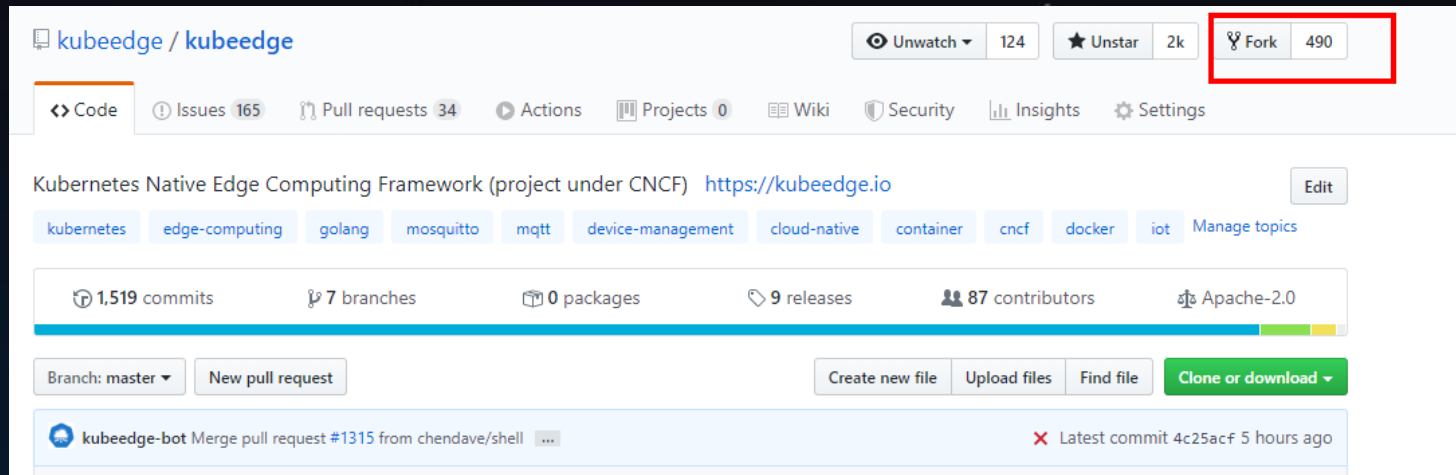
# 社区开发快速上手 —— 社区角色

- Member :
  - 2个reviewer同意，个人主页显示KubeEdge徽章（5+个Pull request即可申请）
  - 申请指南：<https://github.com/kubeedge/community - membership>
- Reviewer :
  - 2个Approver同意，负责review一块代码
- Approver :
  - 2个Maintainer同意，负责维护相应模块的代码，拥有merge代码权限
- Maintainer:
  - 2个Owner同意，负责整个KubeEdge的特性、版本管理
- Owner :
  - 3个Owner同意，负责整个项目的技术演进与发展方向

# 社区开发快速上手 —— 代码贡献

## 提交流程：

- Fork KubeEdge项目
- 新建个人开发分支
- 编写开发的Commit
- 提交一个Pull Request到KubeEdge项目



## 合入流程：

- Member、Reviewer、Approver检视代码
  - 修改意见
  - 添加 “lgtn” 标签
- Approver合入代码
  - 添加 “approve” 标签

社区欢迎每一位贡献者！

# 参与KubeEdge社区贡献 —— 赢HDC大会门票



## 活动规则

鉴于本次活动接受多种贡献方式，下面是我们对贡献的一些要求以及如何产生获胜者：

### 代码贡献：

任何代码贡献都应该遵循贡献流程。我们会评估贡献者提交的关于特性开发，测试用例开发和 Bug Fix的 Pull Request。

### Issue提交：

我们将考虑提交的Issue的严重性、复现步骤和数据的充分性。

### 需求反馈：

需求的唯一性，以及与其他边缘计算平台相比它可以带来的价值会作为评选的重要因素。

### 项目推广：

任何关于KubeEdge的博客、微信、Twitter、文章、白皮书等，我们会考虑它们的内容和受欢迎程度。

### 使用样例：

我们将考虑使用样例包含的代码、描述文档和用户指南等。

### 奖励机制：

1. 参与社区贡献，合入5个及以上 Pull Request即可申请为KubeEdge、Volcano社区 member - Github个人profile页面显示社区徽章。
2. 根据贡献积分排名，华为云将邀请20名开发者现场参加2020年华为开发者大会（HDC），获得与开源大咖面对面交流的机会，并且参与现场开发者活动，有机会赢取大奖。我们将在2020年1月20日通过邮件、Slack、微信、Twitter等多个渠道发布获胜者名单。

### 活动时间：

2019年12月16日 - 2020年1月20日



## 社区欢迎每一位贡献者！

<https://github.com/kubeedge/kubeedge>

## 公众号容器魔方



每日推送图文  
社区最新动态、直播课程、技术干货

## KubeEdge技术交流群



添加小助手，发送KubeEdge加群  
社区专家入驻，技术问题随时答疑



# Thank You

<https://bbs.huaweicloud.com/webinar/91fadcd3ea2a435f91771fc13d5136b8>

直播 每周四 晚20:00