GOTC

全球开源技术峰会

THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE

OPEN SOURCE , OPEN WORLD

「边缘计算」专场

本期议题：基于 Kuiper 和 KubeEdge 的边缘流数据分析

黄济泳 2021年08月01日

Agenda

- 开源 IOT 边缘流数据分析：LF Edge eKuiper
- 使用 eKuiper 进行边缘流数据分析
- KubeEdge + eKuiper: 云边协同的流处理方案
- eKuiper 开源社区现状与未来规划

流式分析

流式分析在边缘端

- 流式分析
  - 对无限的数据流进行状态分析的软件或者框架
  - 对流数据进行管理、监控和实时分析
  - 典型运行在云端

- Apache Flink & Spark 不适合边缘端流式分析
  - 时延
  - 数据安全
  - 带宽成本

- 边缘流式分析的挑战
  - 轻量级和高效：边缘端 CPU 和内存受限
  - 敏捷与灵活：需要用更加敏捷的方式实现业务支持，避免复杂的编码和编译等工作
  - 部署和管理：非集中，在弱网情况下的分布式部署

LF Edge eKuiper

GOTC

开源 IoT 边缘分析

2021 年 6 月份加入 LF Edge
- 首个由国内初创公司贡献的项目

eKuiper 项目主要里程碑
- June 2021: LF Edge Stage 1 At Large
- May 2021: 发布 1.2.0，支持ML/AI 函数调用
- Feb, 2021: 发布 1.1.1，支持二进制数据处理
- Oct, 2020：发布 1.0.0，第一个稳定的主版本
- Jul, 2020：发布 0.5.1，与 KubeEdge 集成
- Apr, 2020：发布 0.3.2，与 EdgeX Foundry 集成
- Oct, 2019: 开源并发布第一个版本

LF EDGE

eKuiper

eKuiper 概述

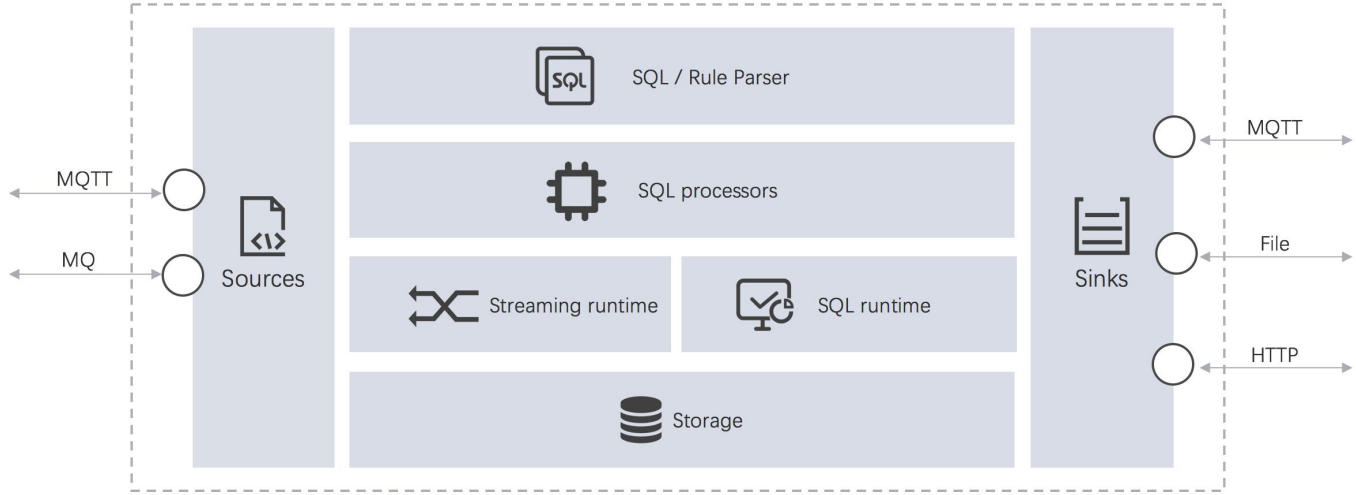
GOTC

为边缘流处理而生

- 二进制可安装包和 Docker 镜像
  - 8MB 安装包; 10MB 初始内存使用
  - X86 AMD * 32, X86 AMD * 64; ARM * 32, ARM * 64; PPC
  - Linux 发行版, OpenWrt Linux, MacOS 与 Docker

- 性能
  - 树莓派 3B+
  - TPS: 12k
  - CPU: sys+user 70%
  - 内存: 20M

  * 详情参考 https://github.com/emqx/kuiper#performance-test-res



- eKuiper – 数据 ETL
  - 数据抽取 (extract) : sources
  - 数据转换 (transformation): 使用 SQL 进行分析 + 转换
  - 数据存储 (load): sinks

快速上手

GOTC

使用 eKuiper 3步骤

- 创建流

```
create stream demo '() WITH (FORMAT="JSON", DATASOURCE="$hw/events/device/+/twin/update")'
```

- 创建规则

```
{
  "sql": "SELECT data->tag1->value AS temperature, data->tag2->value AS humidity FROM demo",
  "actions": [
    {
      "log": {}
    },
    {
      "mqtt": {
        "server": "tcp://broker.emqx.io:1883",
        "topic": "devices/result",
        "qos": 1,
        "clientId": "demo_001"
      }
    }
  ]
}
```

- 提交、运行规则

```
curl -X POST http://$kuiper_server:9081/rules -H 'Content-Type: application/json’
-d '$my_rule'
```

SQL分析

GOTC

内置支持

- 函数
  - 数学: sin, cos, abs, log, mod etc; 共计 25个函数
  - 字符串: concat, substring etc; 共计 19个函数
  - 汇聚: avg, count, max, min, sum, collect & deduplicate; 共计 7 个函数
  - Conversion/ Encoding & decoding / Hashing / JSON processing / 其它; 共计 18个函数
  - 二进制处理：resize, thumbnail; 共计 2 个函数
  - GeoHash函数，共计10个函数
- 过滤
  - WHERE / CASE WHEN
- Join (LEFT | RIGHT | FULL | CROSS JOIN)
  - Streams: 动态流动数据
  - Tables: 静态数据，经常用于关联更多别的数据。如提供用户 id，获取相关的名字
- Window
  - Tumbling / Hopping / Sliding / Session / Count
  - Group By & Order By

SQL分析能力扩展

GOTC

自定义函数

- 自定义函数
  - 与普通内置函数相同，都可以在SQL中使用
- 使用配置方式扩展，调度已有服务
  - 方便快速，推荐使用
  - 微服务架构，语言无关，崩溃不影响主程序
  - 配置 json 和 proto 文件
  - 支持 gRPC, REST, msgpack RPC

```
SELECT getFeature(base64Img) as feature FROM images
```

- 使用插件扩展
  - 仅可使用 go 编写，编译有门槛
  - 性能相对较好，支持状态 API

```
{
    ...
    "interfaces": {
      "tsrpc": {
        "address": "tcp://localhost:50051",
        "protocol": "grpc",
        "schemaType": "protobuf",
        "schemaFile": "hw.proto",
        "functions": [
          {
            "name": "getFeature",
            "serviceName": "getFeatureSrv"
          },
          ...
        ]
      },
      ...
    }
}
```

```
// Outputs
{
  "feature": [
    {
      "features": [...],
      "box": ...
    }
  ]
}
```

插件扩展

GOTC

自定义数据汇入与流出

- 扩展点
  - Source
  - Sink
  - Function

- 步骤
  - 开发 & 调试
  - 编译为 *.so 文件
  - 部署插件

- Native 插件开发
  - 优点: 更好的性能
  - 缺点: 来自于 Golang 的诸多限制, 比如必须为同一个 go 版本, 严格的库文件依赖和 GOPATH

```
//Called during initialization. Configure the source with the data source(e.g. topic for mqt
t)
//and the properties read from the yaml
Configure(datasource string, props map[string]interface{}) error

//Should be sync function for normal case. The container will run it in go func
Open(ctx StreamContext, consumer chan<- SourceTuple, errCh chan<- error)

Close(ctx StreamContext) error

function MySource() api.Source{
    return &mySource{}
}
```

Source 扩展：扩展 source 所需实现的接口

数据分析实例

GOTC

使用窗口进行有状态计算

- 计算最近 5s 平均值变化百分比
- SQL

```
SELECT (ts - windowStart() > 5000) as firstHalf, AVG(val) as avg FROM demo
GROUP BY HoppingWindow(ss, 10, 5), firstHalf ORDER BY firstHalf
```

- SQL输出样例

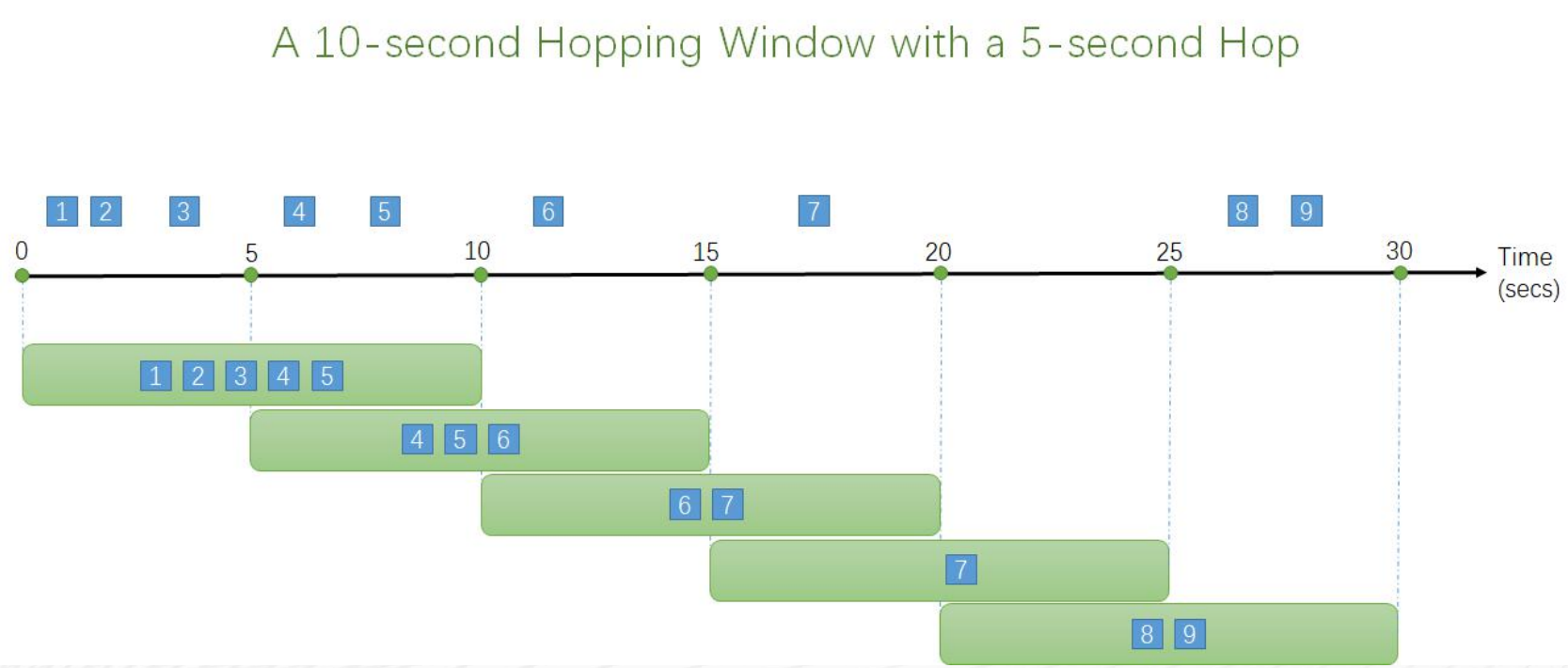
```
[{"firstHalf": false, "avg": 3},
{"firstMin":true, "avg":4.5}]
```

- 数据模板

```
"{\"percentage\":\"
{{mulf (round (divf (index . 0).avg
(index . 1).avg) 2) 100}}%\"}"
```

- 最终输出

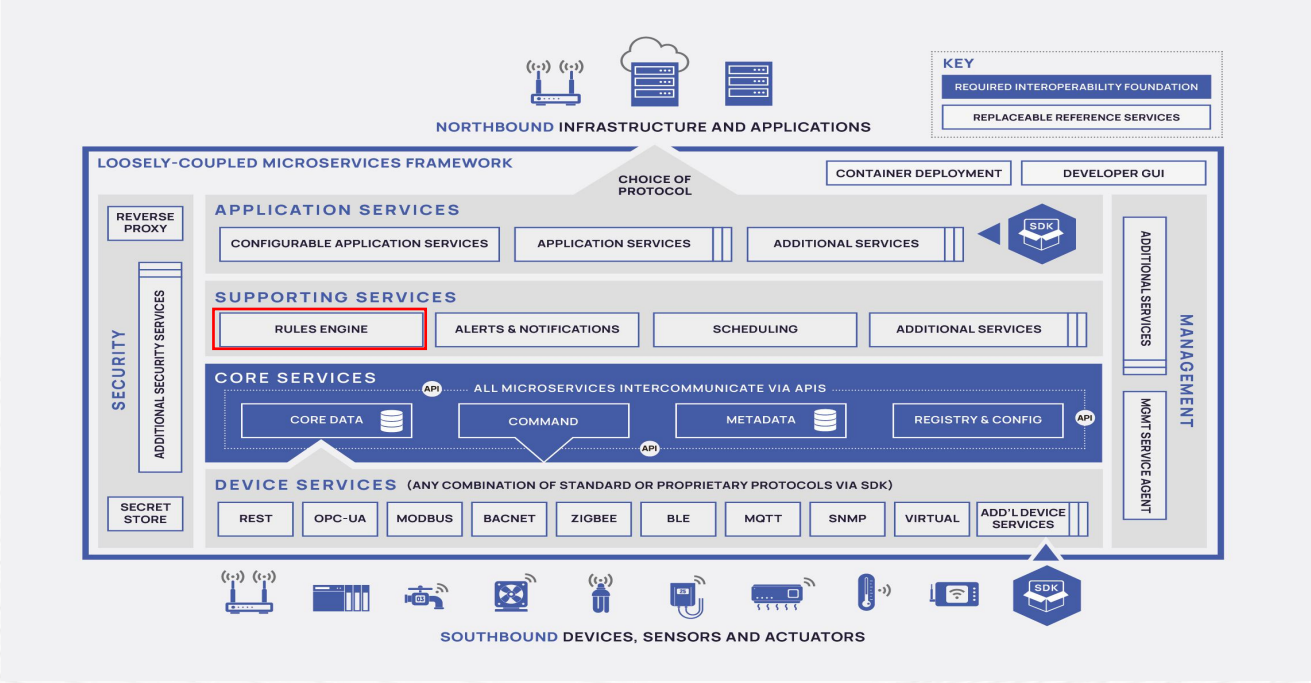
```
{"percentage":"150%"}
```

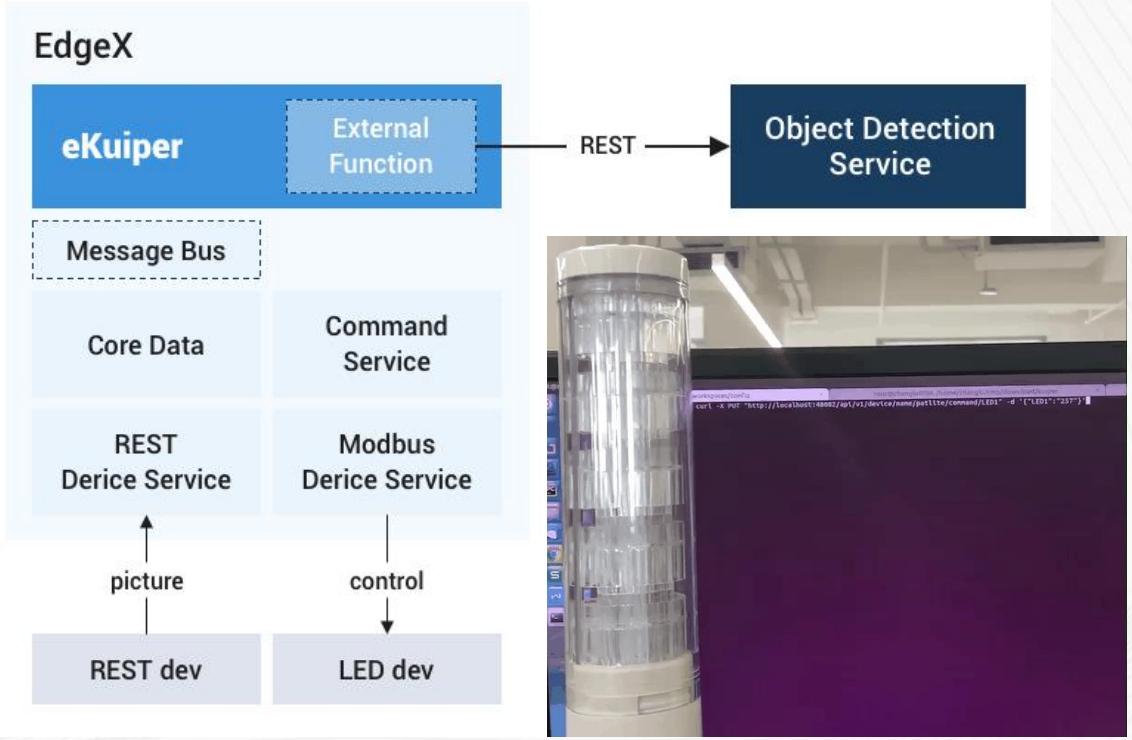


数据分析实例

与 EdgeX 结合，进行实时 AI 分析

- eKuiper 为 Edge X Foundry 内置规则引擎实现
- 通过 eKuiper 实时分析摄像头图片，并触发设备动作





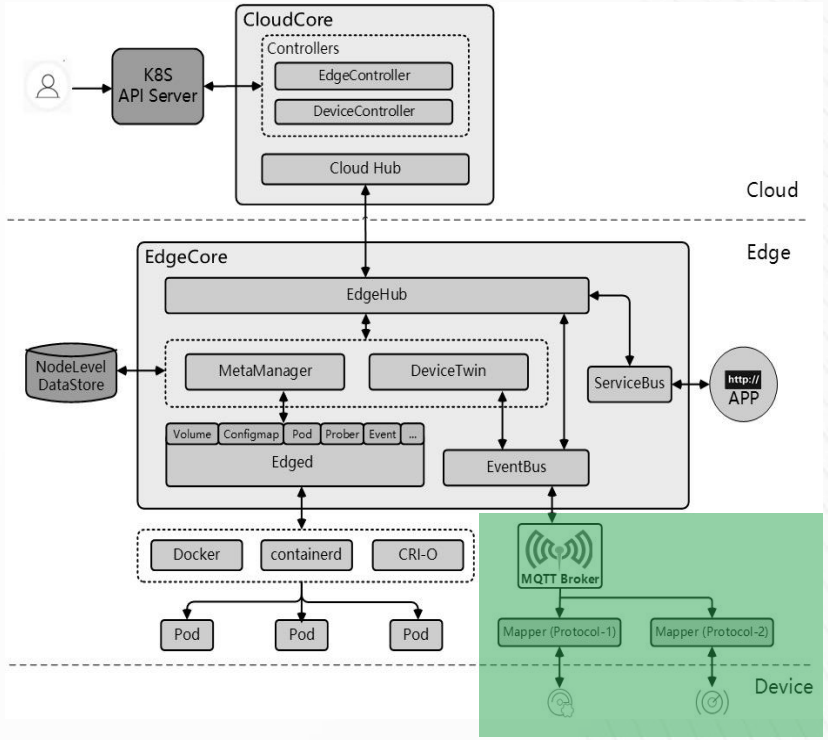
KubeEdge & eKuiper 集成

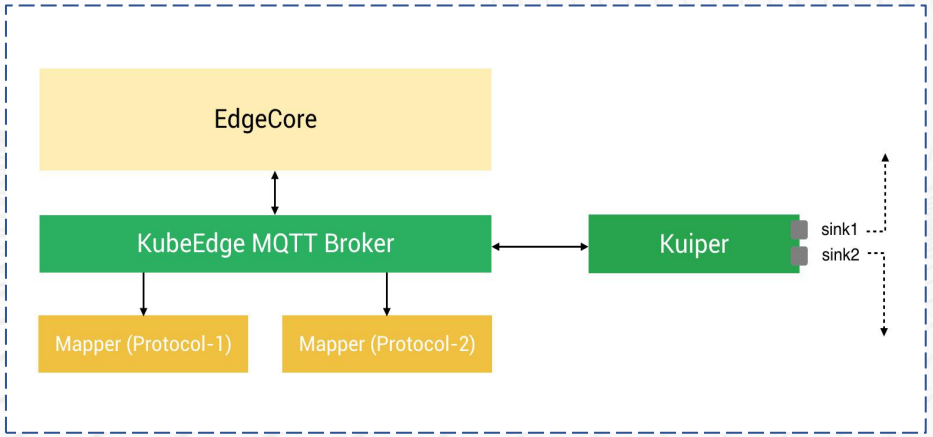
云边容器编排协同 + 边缘分析

- KubeEdge
  - 将原生容器化应用编排功能扩展到边缘节点的开源软件

- eKuiper 增强了边缘分析能力

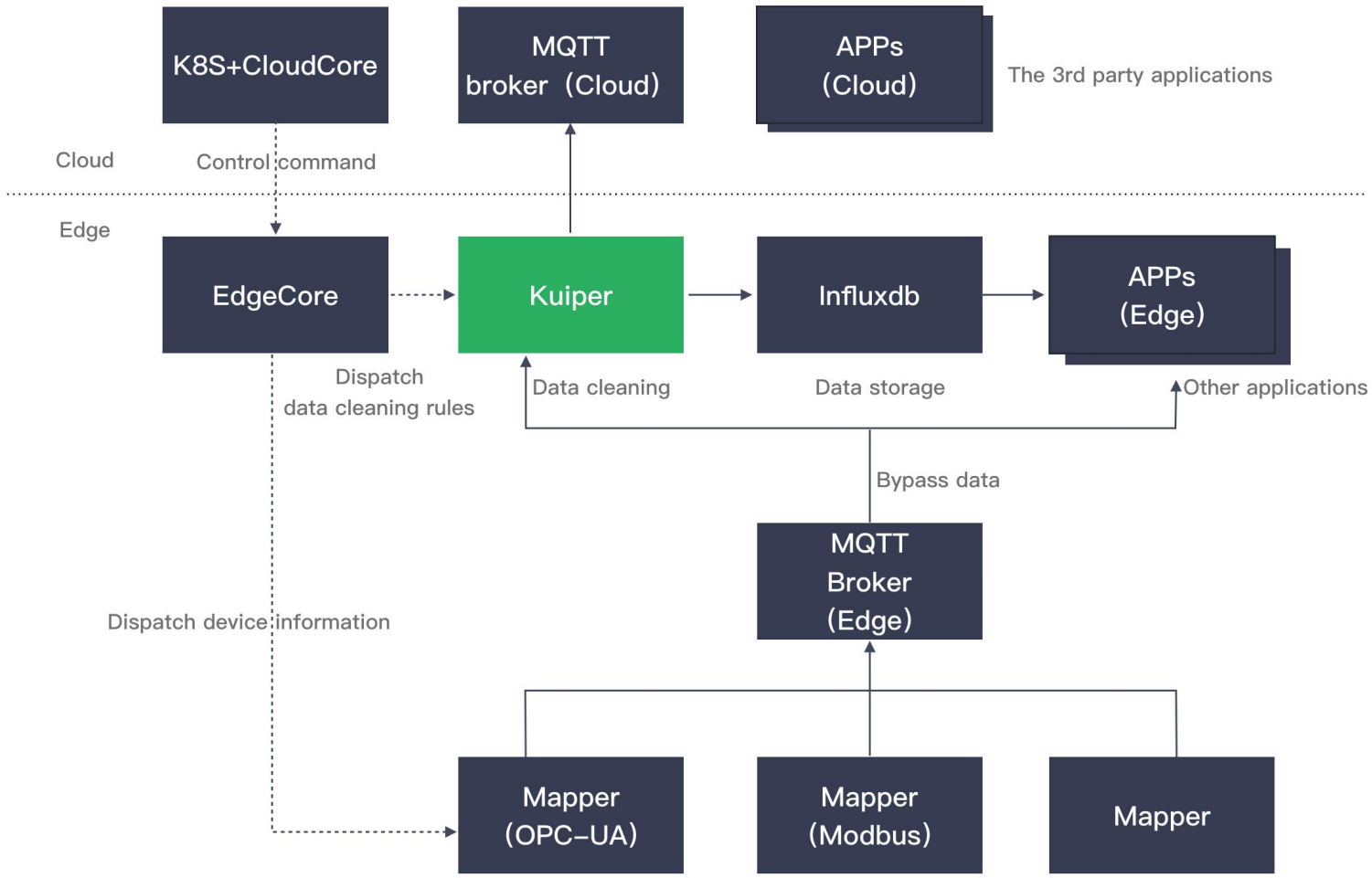
- 好处 – 解决了 IoT 边缘分析的挑战
  - 低时延，节省带宽
  - 用户易于实现业务逻辑
  - 从云端管理和部署 eKuiper、应用、AI 算法





客户案例：KubeEdge + eKuiper

中国移动 – 国家工业互联网大数据中心



全球开源
THE GLOBAL OPENSOURCE

eKuiper 开源社区

GOTC

社区与未来规划

- 社区相关链接
  - Github: https://github.com/lf-edge/ekuiper
  - 网址：https://www.lfedge.org/projects/ekuiper/
  - Wiki: https://wiki.lfedge.org/display/EKUIPER/eKuiper
- 社区参与: CONTRIBUTING
  - Code/Doc PR
  - Enhancement Request/Issues
  - Collaboration
- 定期会议：Calendar

- 下一步计划
  - 与更多开源项目合作
  - 2021 Roadmap

- **欢迎加入 eKuiper 社区**

GOTC

THANKS

全球开源技术峰会

THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE