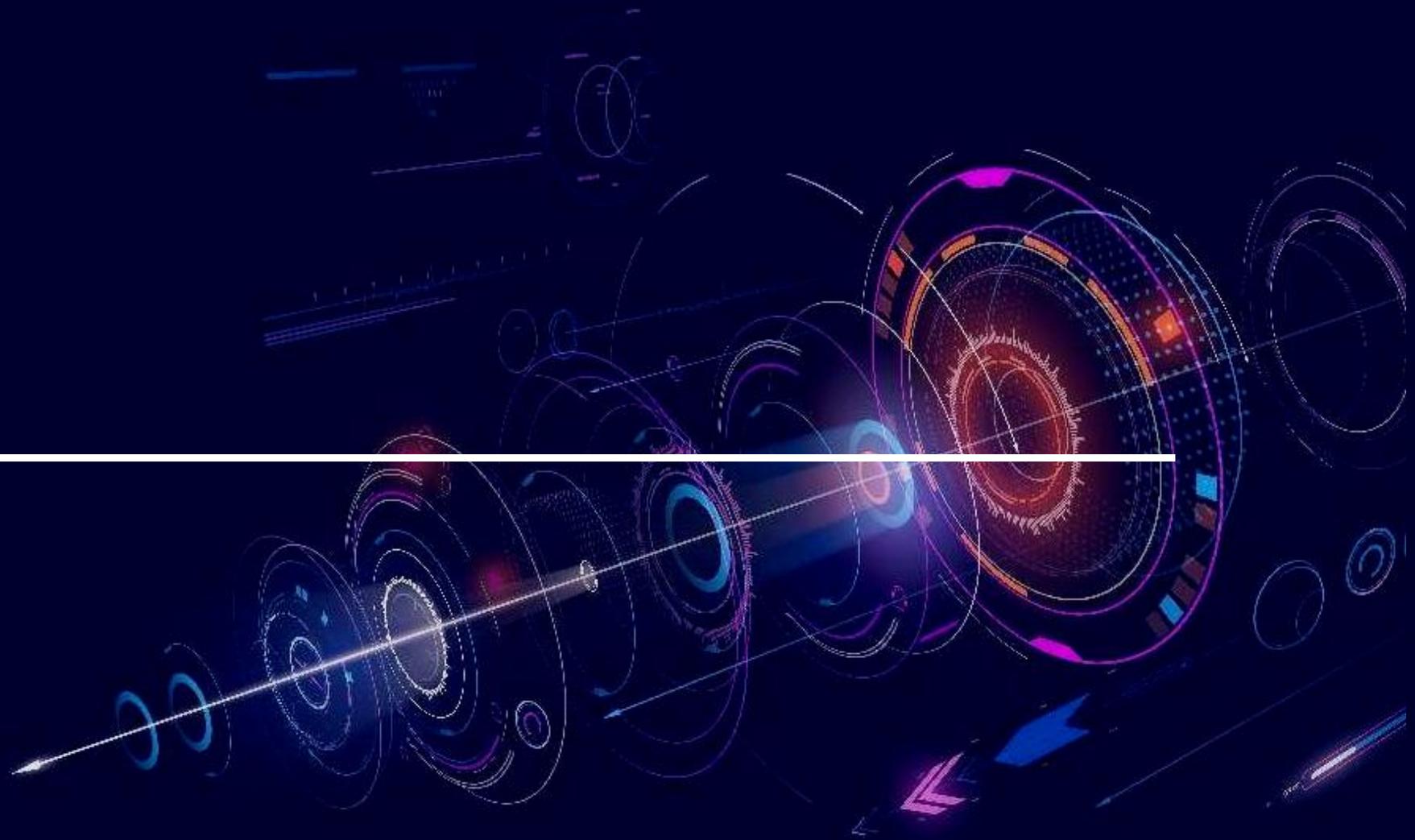


边



C

云

数

云原生技术发展

1

云原生技术

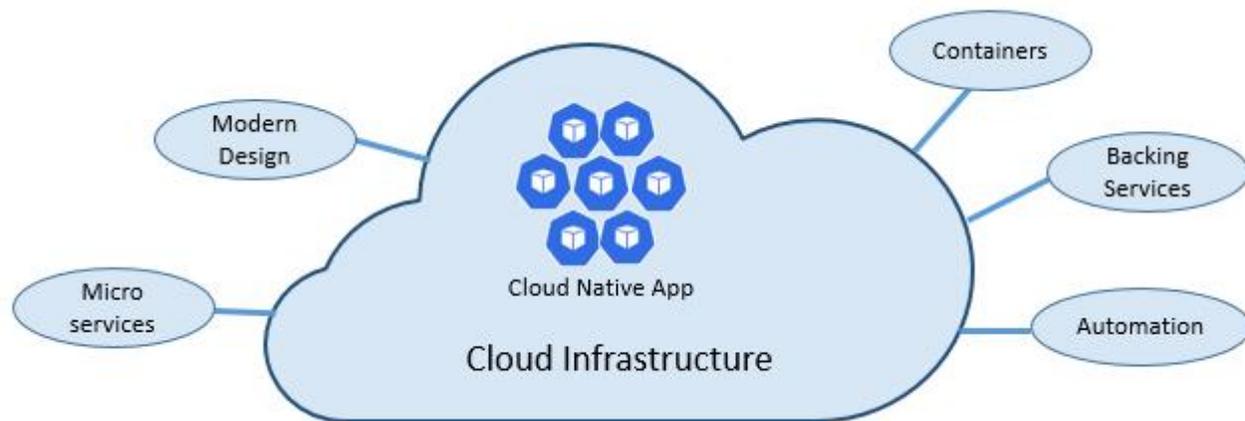
- Cloud-native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds. Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach. — From

CNCF

- 云原生技术有利于各组织在公有云、私有云和混合云等新型动态环境中，构建和运行可弹性扩展的应用。云原生的代表技术包括容器、服务网格、微服务、不可变基础设施和声明式 API。

- Why cloud native?

- speed
- agility
- scalability



— From MS docs

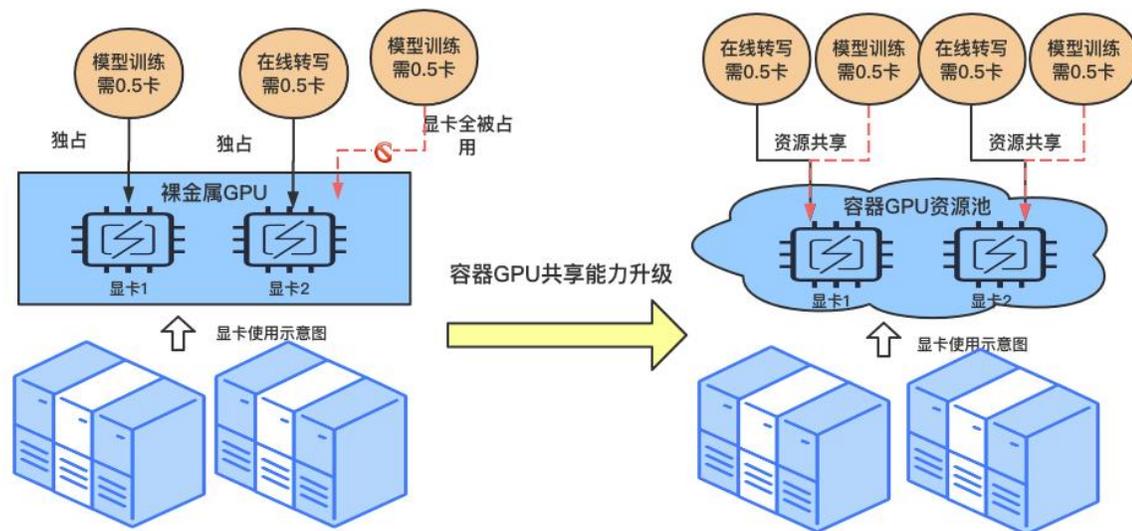
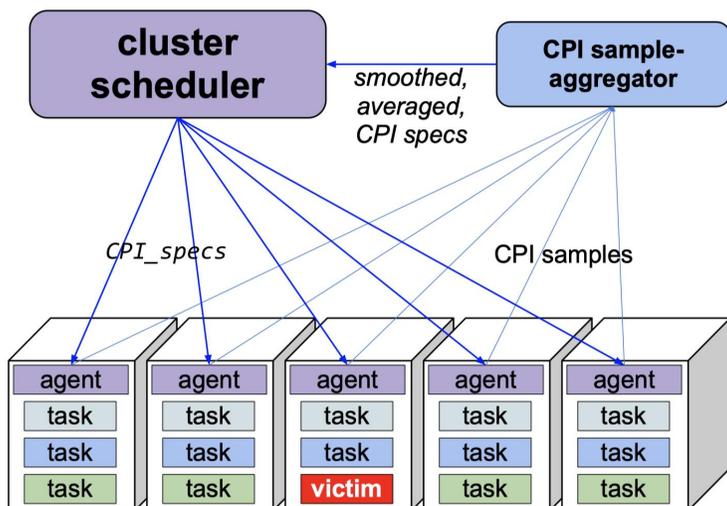
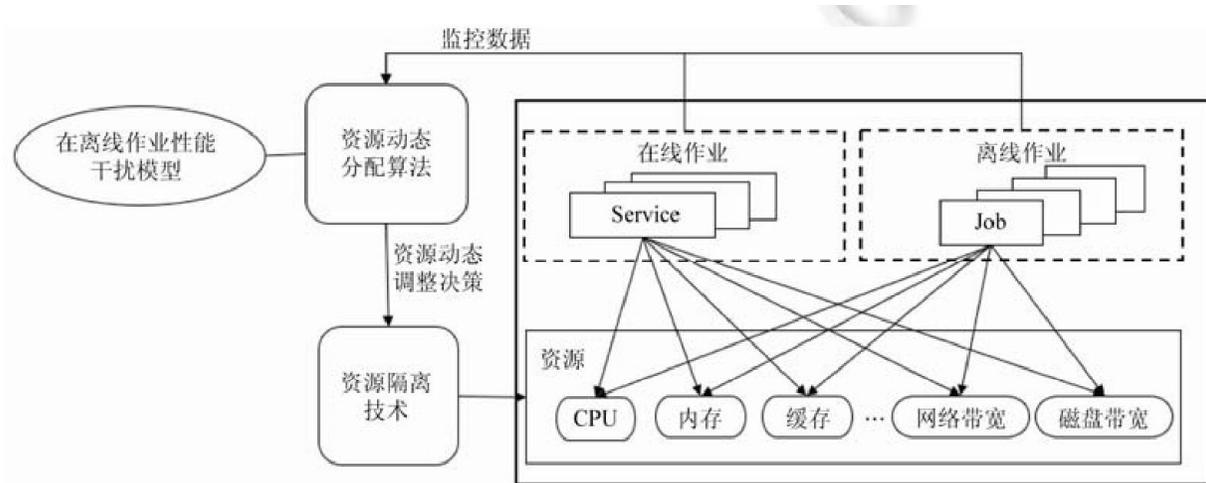
CNCF 蓝图

The image displays a grid of CNCF projects categorized into 14 functional areas. Each project tile includes a logo, name, and CNCF status (Graduated, Incubating, Sandbox). The categories are:

- Database:** KV, Vitess, CockroachDB, TiDB, etc.
- Streaming & Messaging:** cloudevents, NATS, etc.
- Application Definition & Image Build:** HELM, Backstage, Buildpacks.io, KubeVirt, etc.
- Continuous Integration & Delivery:** argo, flux, etc.
- Scheduling & Orchestration:** kubernetes, CoreDNS, etcd, etc.
- Coordination & Service Discovery:** CoreDNS, etcd, etc.
- Remote Procedure Call:** gRPC, etc.
- Service Proxy:** envoy, etc.
- API Gateway:** EMISARY INGRESS, etc.
- Service Mesh:** LINKERD, etc.
- Cloud Native Storage:** Rook, Longhorn, etc.
- Container Runtime:** cri-o, etc.
- Cloud Native Network:** cilium, etc.
- Automation & Configuration:** KubeEdge, etc.
- Container Registry:** Harbor, etc.
- Security & Compliance:** Falco, etc.
- Key Management:** spiffe, etc.

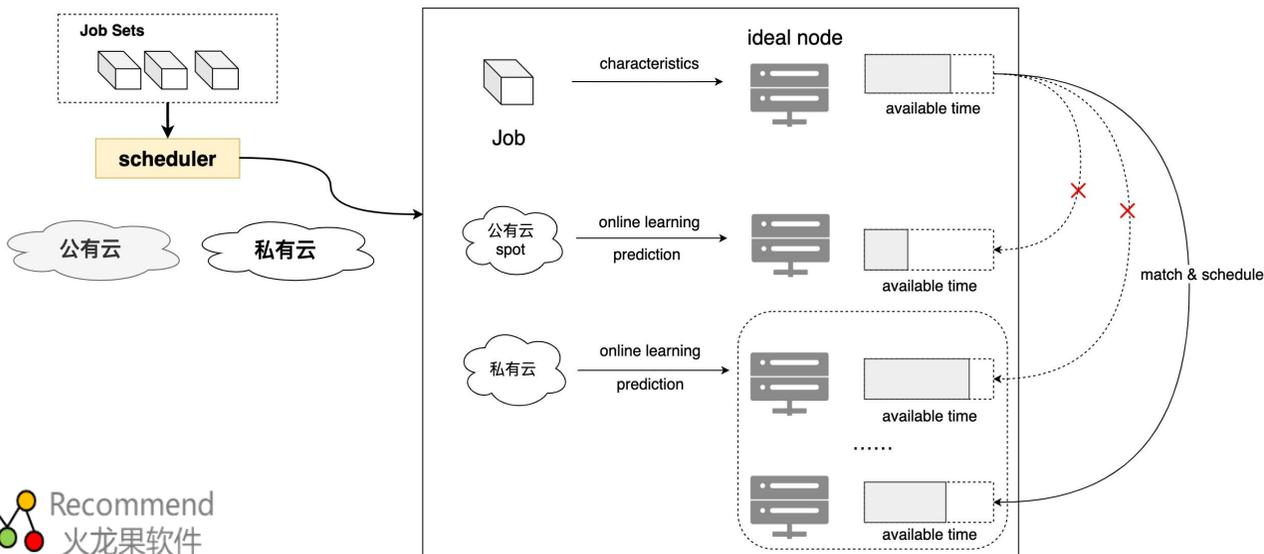
云原生技术发展 —— “栈”化

- 在离线业务一“栈”化。离线业务容器化，与在线应用混合部署，保障业务整体QoS
- GPU的容器化共享
- 服务整体干扰度量

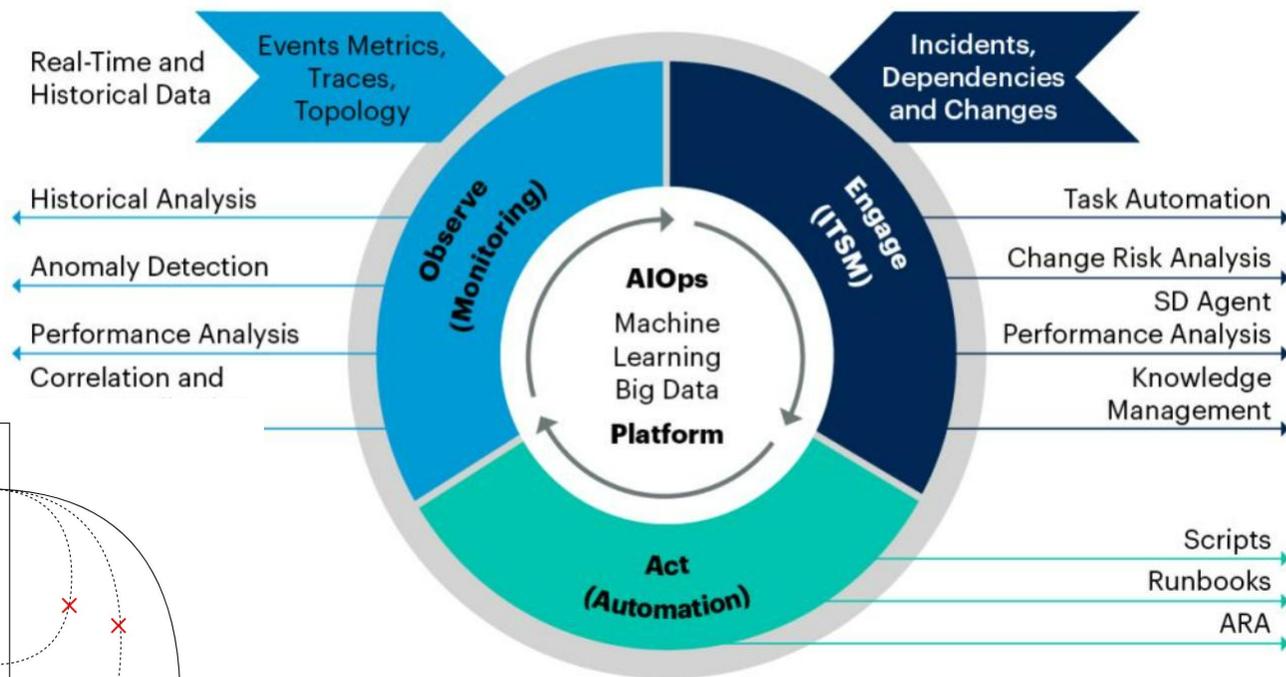


云原生技术发展 — AIOps

- 云原生AIOps, 提升云用户体验
 - 用云成本优化
 - 最佳配置实践
 - 故障定界分析
- FinOps?

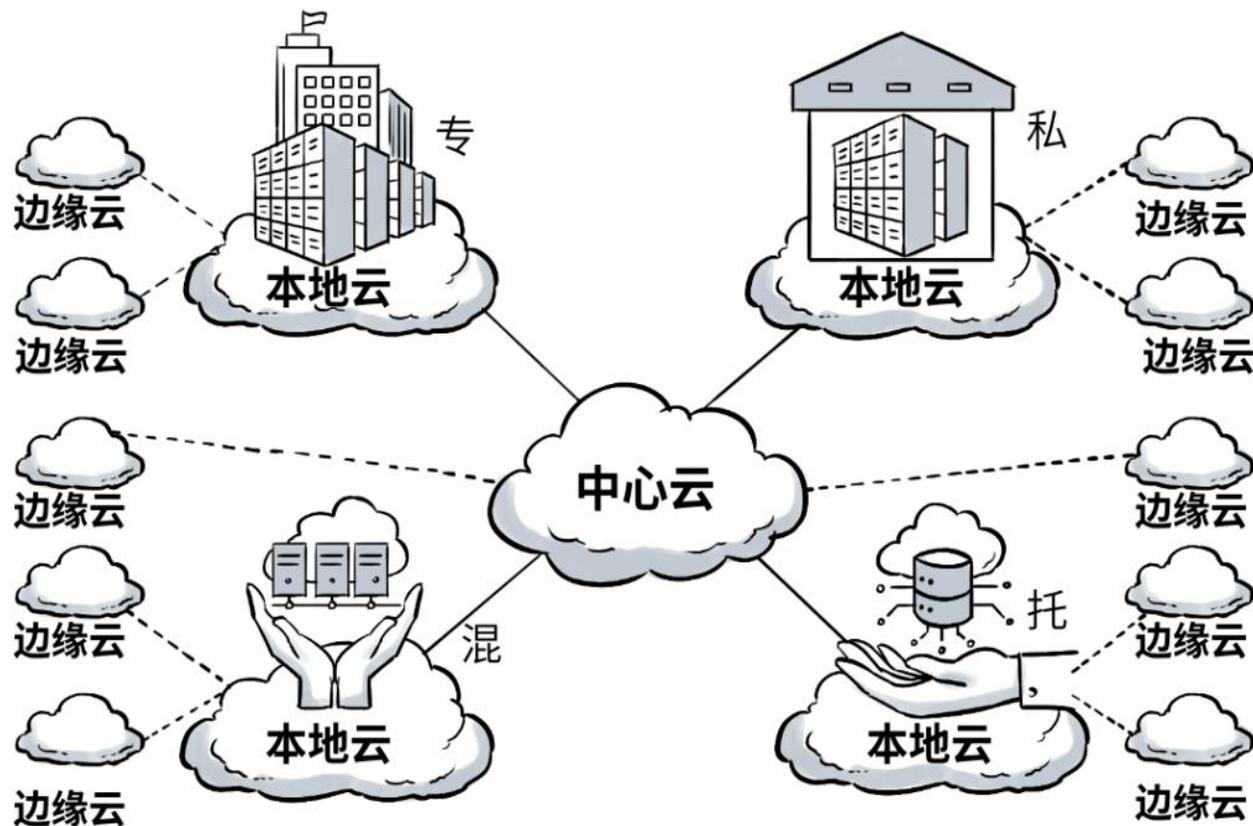


AIOps Platform Enabling Continuous Insights Across IT Operations Monitoring (ITOM)



云原生技术发展 — 分布式云

- 具有边缘计算能力的分布式云将是云计算的一个趋势
- 《浙江省数字基础设施发展“十四五”规划》提到，“打造云边协同的绿色数据中心集群”
- 分布式云特点：
 - 降低延时提高性能
 - 降低使用成本
 - 扩展分布式业务
 - 降低网络故障的风险



示意图来源于网络

云原生边缘计算

2

边缘计算

边缘计算

是在靠近物或数据源头的网络边缘侧，融合

- 网络
 - 计算
 - 存储
 - 应用编排能力
- 的**分布式开放平台**，就近提供边缘智能服务，满足行业数字化在
- 敏捷联接
 - 实时业务
 - 应用智能
 - 有限自治
 - 安全与隐私保护
- 等方面的关键需求。

(引自边缘计算产业联盟《边缘计算与云计算协同白皮书2018》)

截止2021年，边缘计算连续第四年被Gartner评为十大战略性技术趋势。

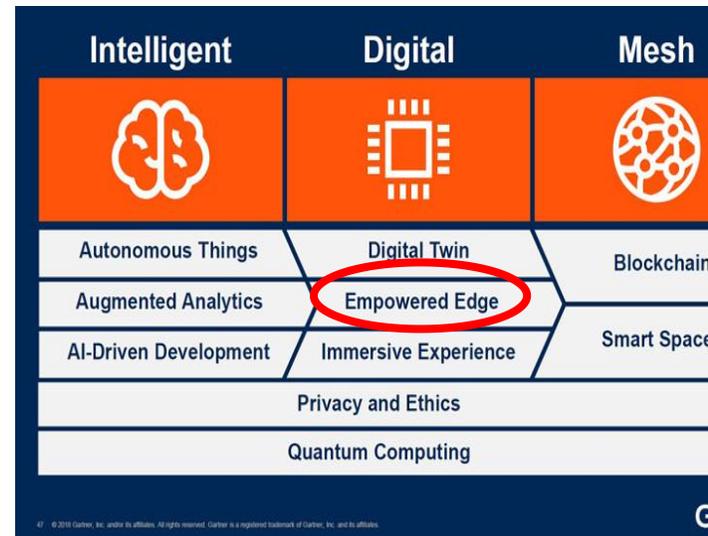
- 2018



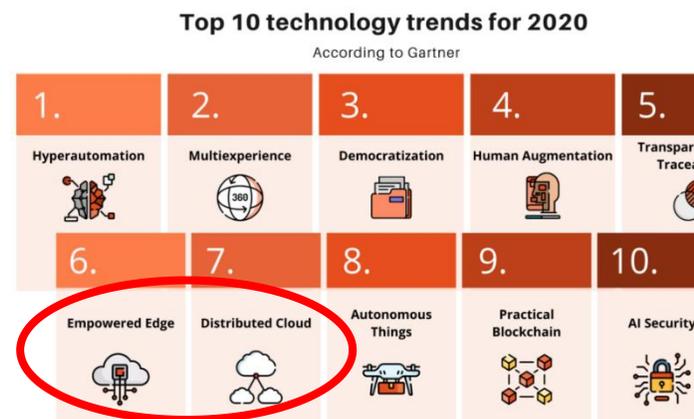
- 2021



- 2019



- 2020



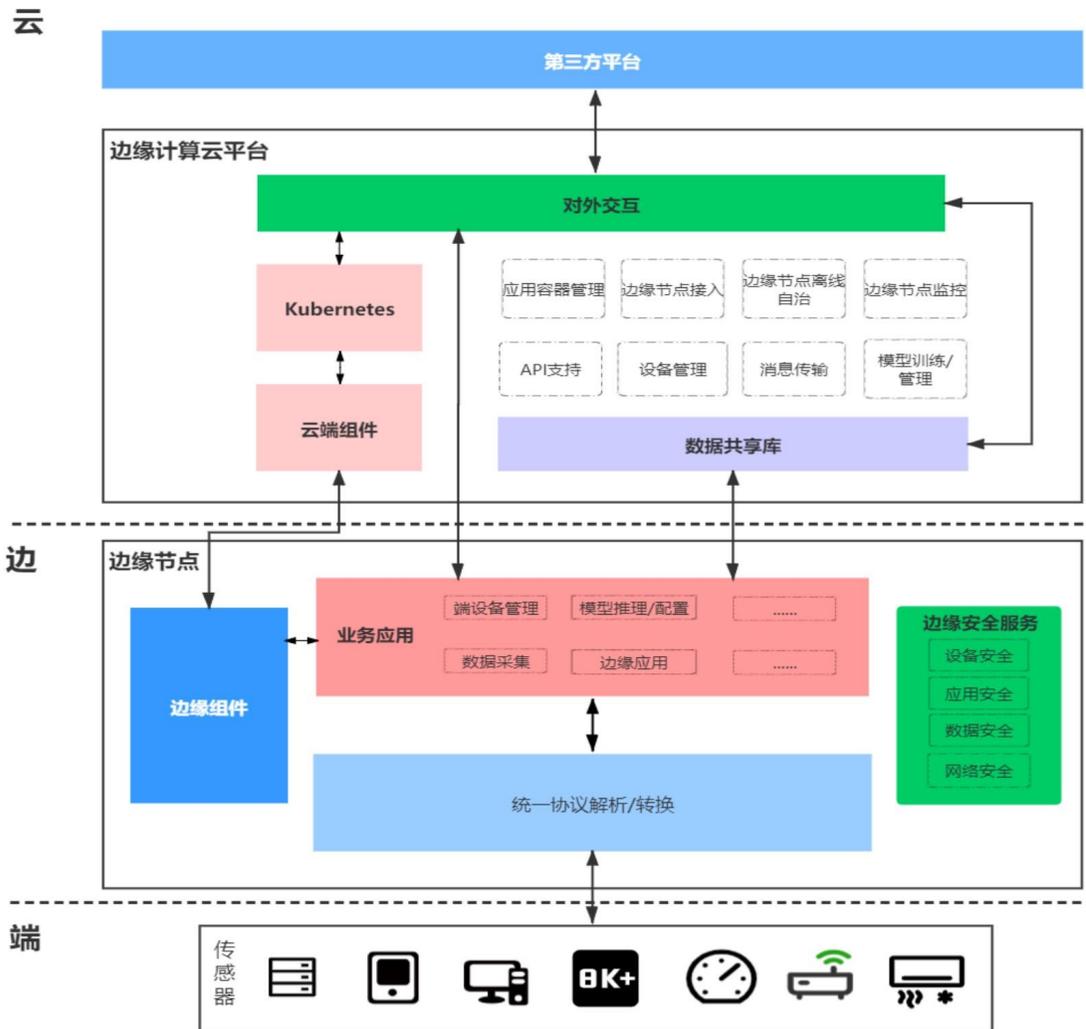
云原生边缘计算

- 主要特点:

- 海量边缘集群接入
- 云边集群协同
- 边缘自治
- 云边通讯

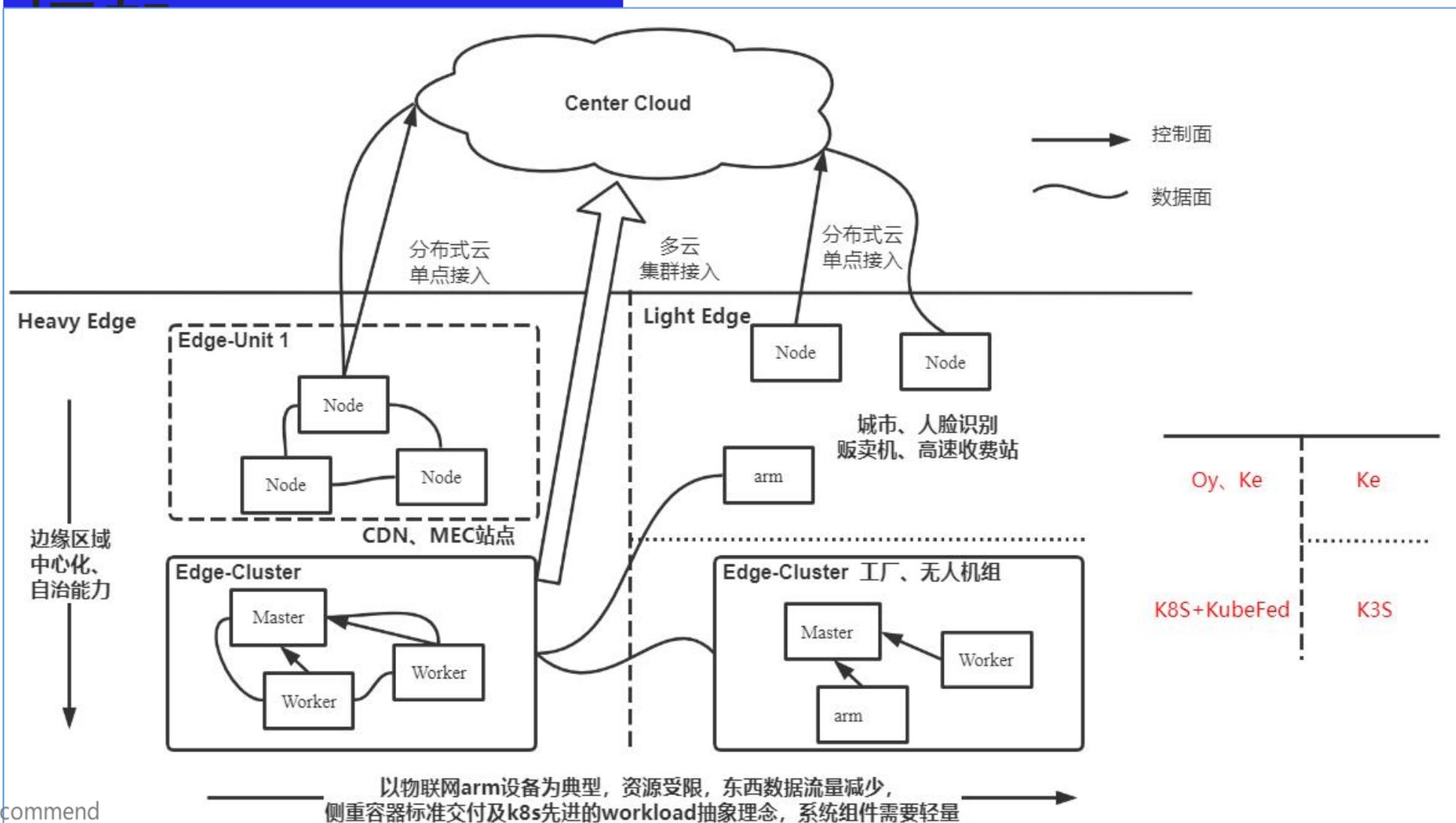
- 面临挑战

- 异构边缘资源的管理
- 时空配准与状态管理需求
- 边缘智能支持 (AI资源需求与边缘资源受限的矛盾)
- 边缘侧低代码



解耦、轻量、可定制

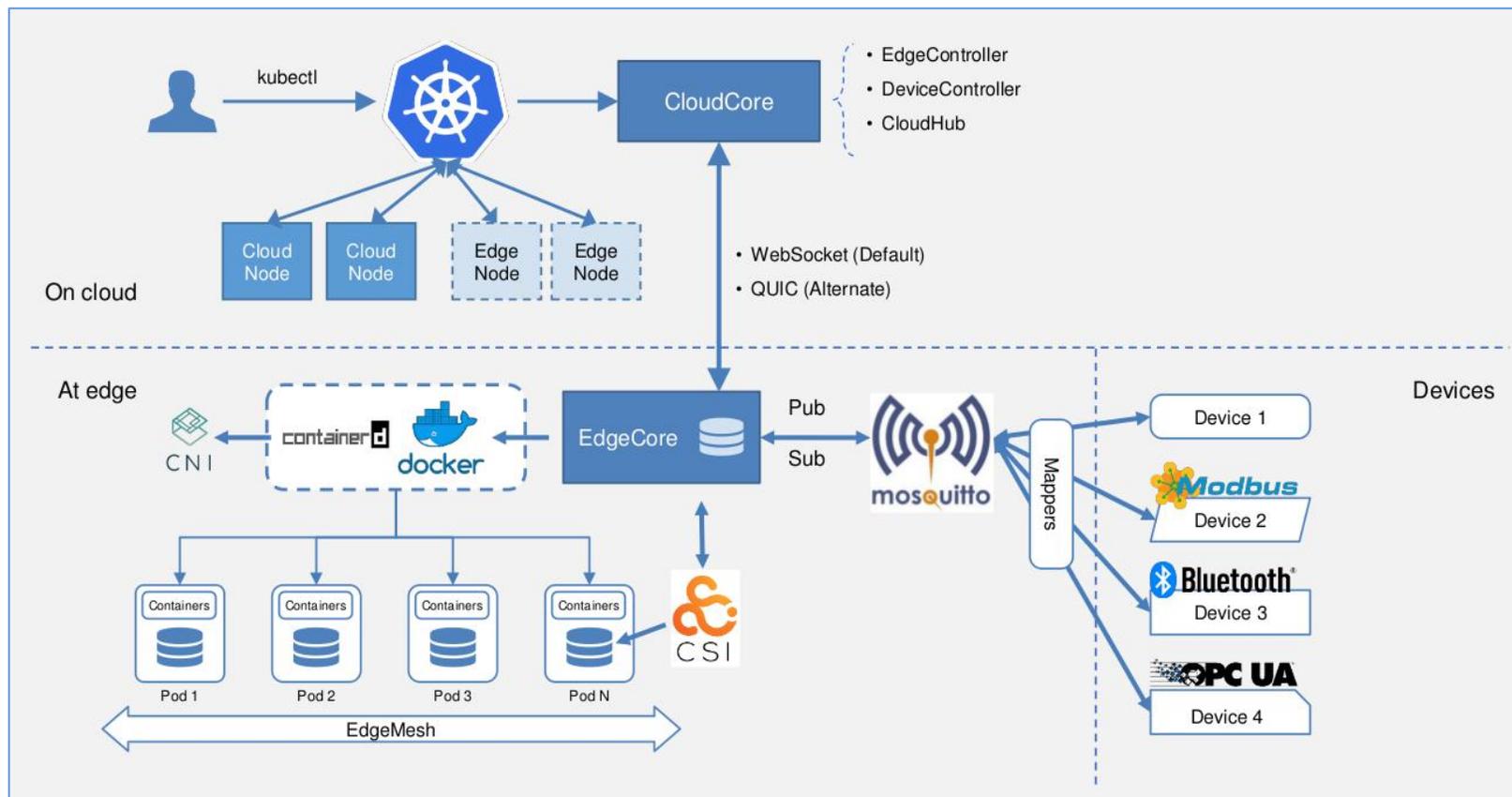
边缘云原生整体



主流平台

KubeEdge功能

- CloudCore接入代理
- 云边元数据协同
 - 可靠传输
 - 监听同步
 - 消息化封装
- 融合设备管理框架
- 轻量化Kubelet



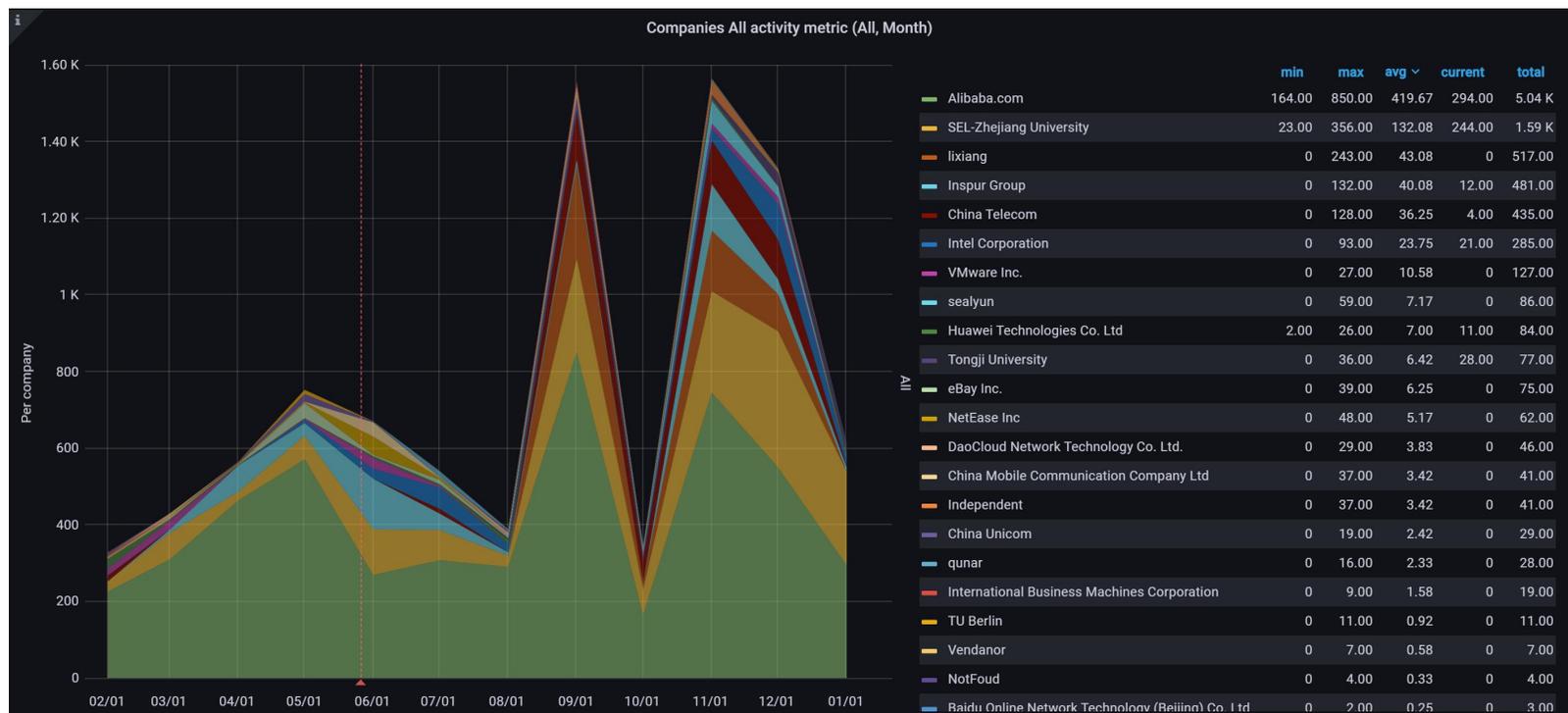
主流框架

浙大边缘计算小组, OpenYurt社区开源贡献统计 (2021年)

- 主库48commits, 2万+LOC
- 子项目32 commmits, 4万+LOC
- Device模块maintainer

主要参与feature:

- 边缘自治
- 边缘智能
- 设备管理



边缘场景下中间件

3

云原生中间件

分布式消息服务

- RocketMQ、Kafka、Pulsar、RabbitMQ

分布式事务系统

- TCC、两阶段提交、本地消息

分布式配置服务

- 权限控制、审计日志、环境管理、配置回滚、灰度发布、高可用

API网关

- 认证鉴权、流量控制、协议转换

分布式缓存

- 高性能地读取数据、能够动态地扩展缓存节点、能够自动发现和切换故障节点、能够自动均衡数据分区

链路跟踪服务

- 调用链路还原、调用请求量统计、链路拓扑、应用依赖分析等

—— From 云原生产业联盟《云原生中间件白皮书（2020年）》

为什么需要轻量化

边缘资源受限

- 资源空间有限，配置管理组件需要轻量化
- 计算资源受限，不能支持类似云端的计算任务

应用轻量化需求

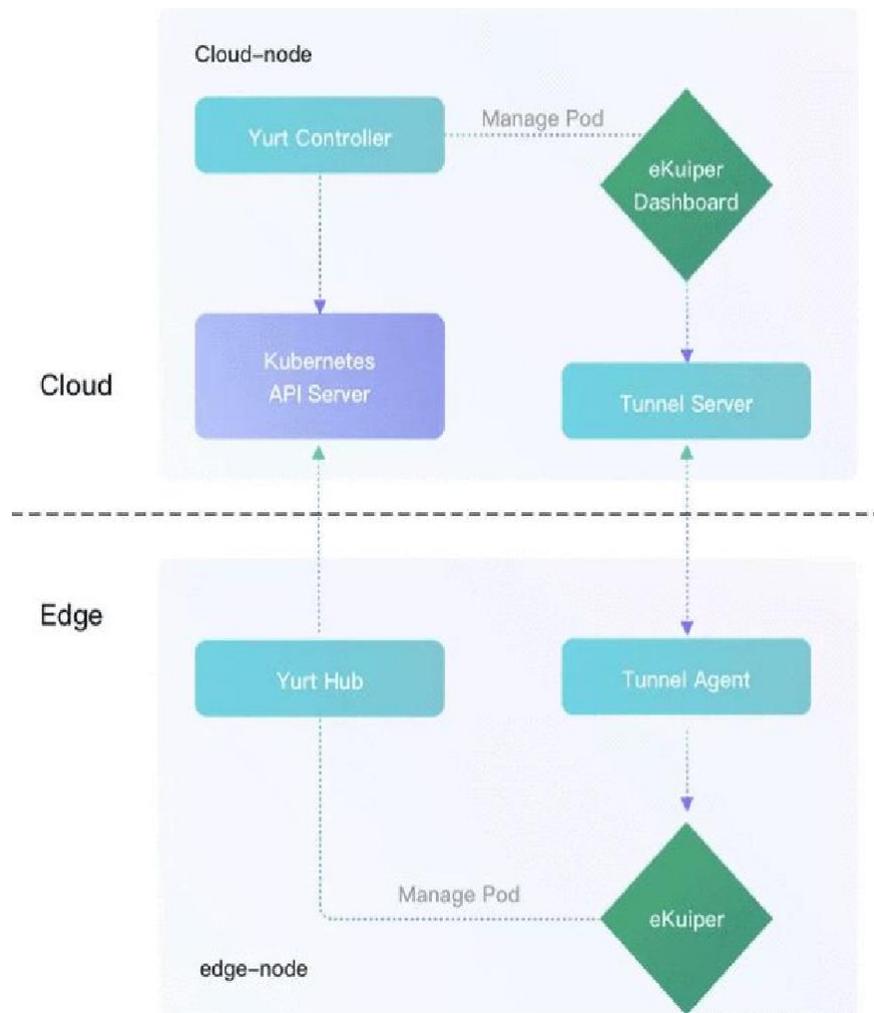
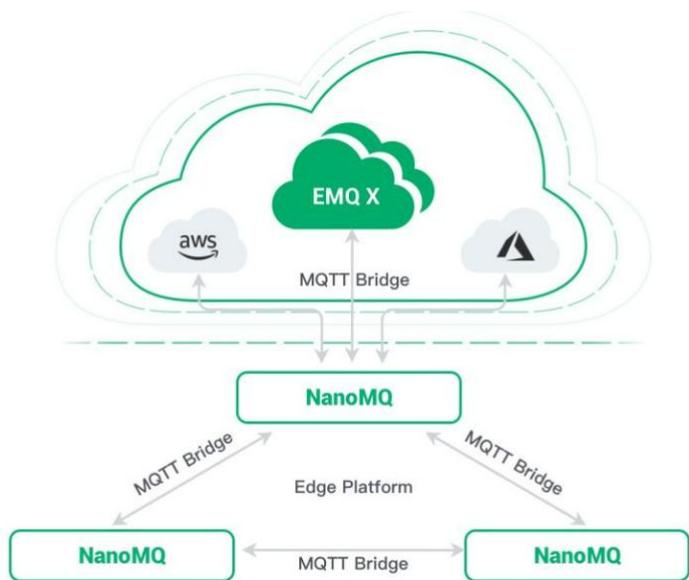
- 轻量化应用
- 轻量化模型

迭代开发演进

- 边缘应用需要更新迭代
- 模型迭代

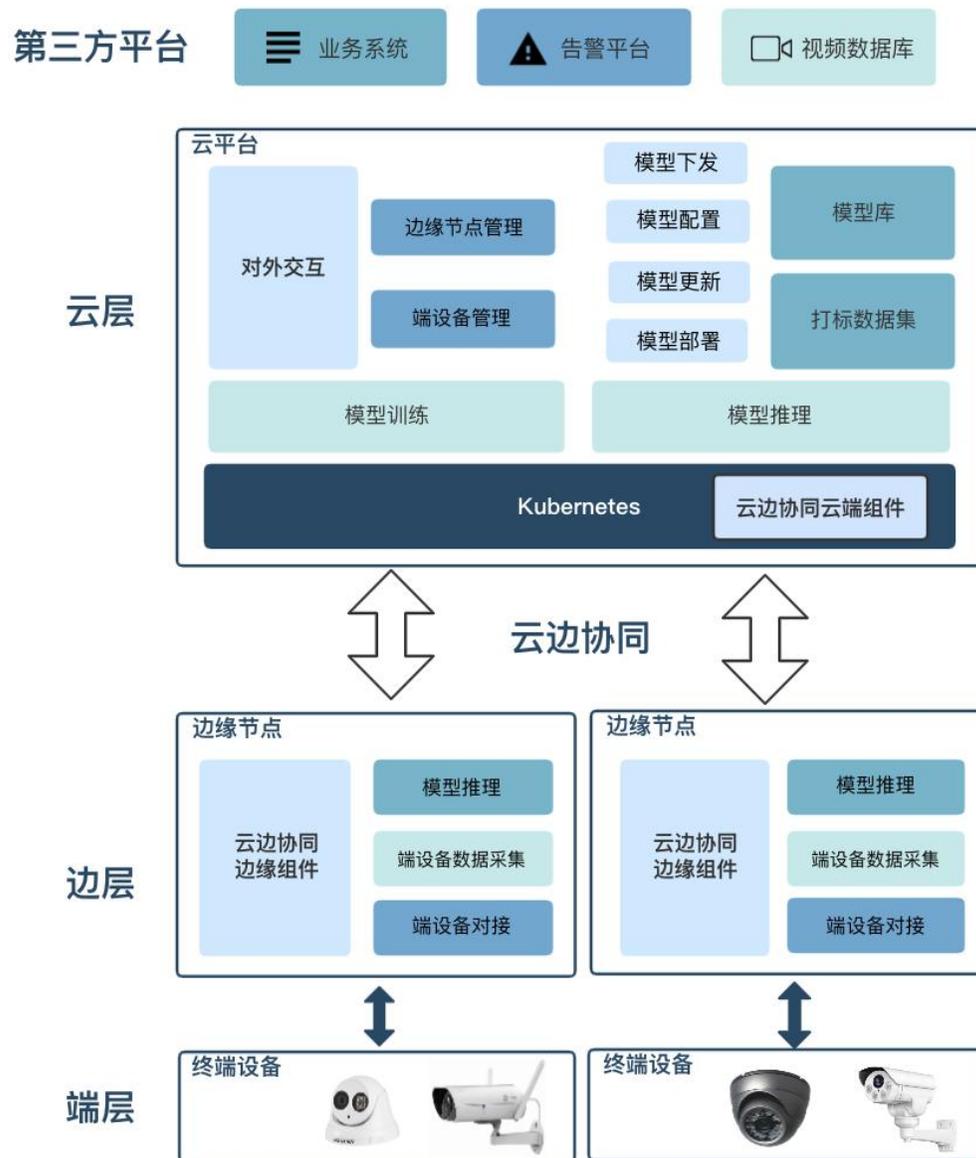
轻量级消息中间件

- 轻量级消息中间件支持物联网边缘场景的数据传输
- 轻量级流数据引擎，提供边缘流式数据的处理能力

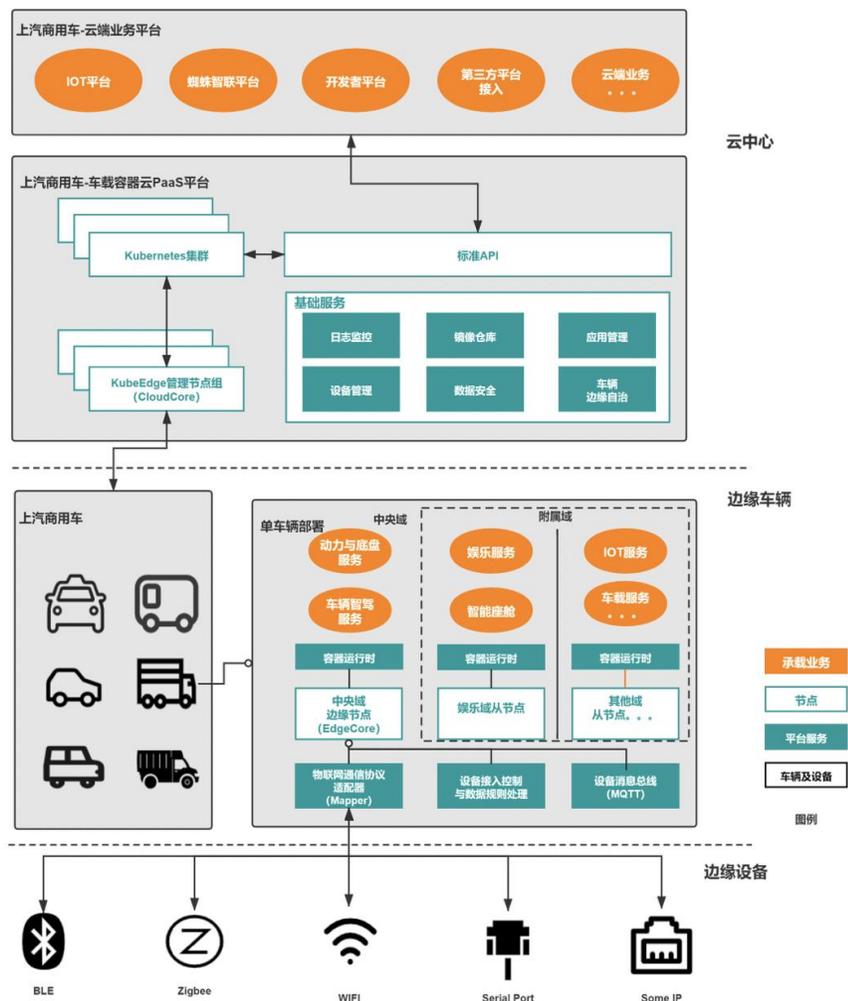


边缘智能中间件

- 轻量级智能计算框架
 - 如Tensorflow lite
- 轻量级算法模型
 - 模型剪枝
 - 知识蒸馏



应用案例：某商用车车云协同



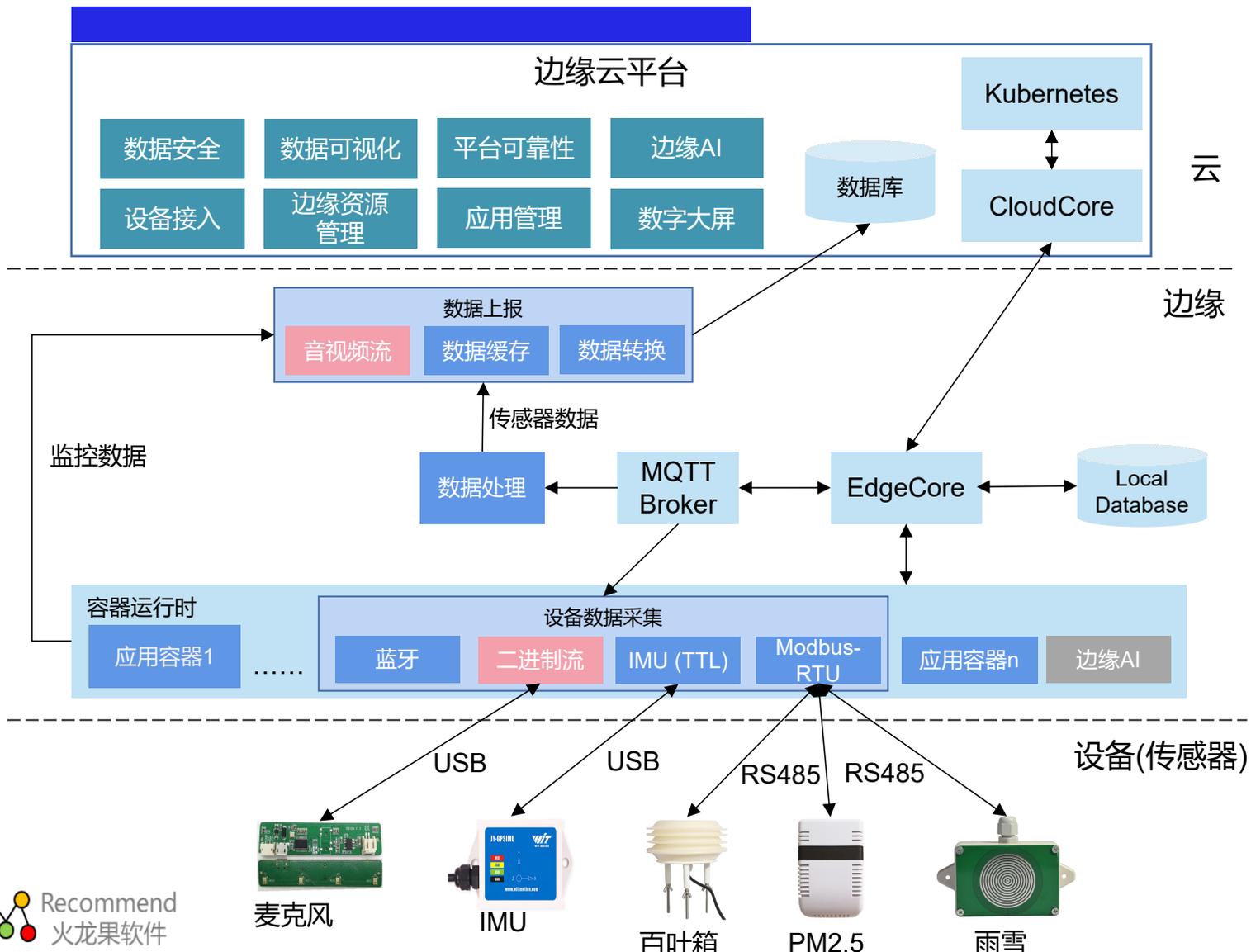
• 痛点:

- 车载服务器可用资源少
- 车载设备的数据可靠传输
- 车辆分布广，车载设备数量多

• 建设效果:

- 轻量化的配置管理组件，<20MB
- 基于边缘缓存的可靠数据传输
- 异步消息中间件应用

应用案例：某跨海大桥数字化



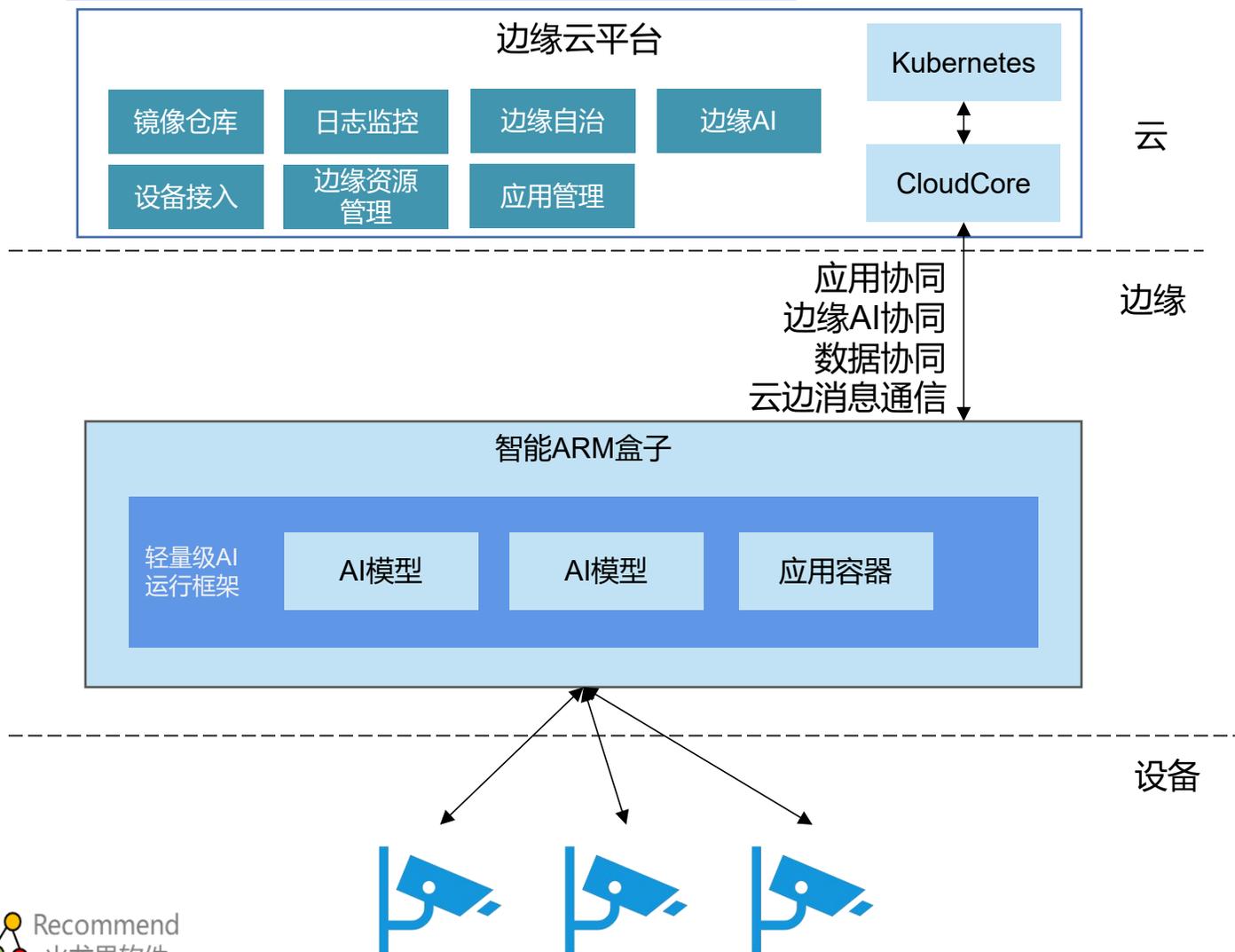
• 痛点:

- 数据传输量大
- 传输可靠性要求高
- 网络不稳定

• 建设效果:

- 边缘流数据处理组件，支持数据接入与压缩
- 轻量级数据存储实现小批量数据缓存

应用案例：某网点视频AI处理



• 痛点:

- 边缘算力需求高、资源有限
- 传输稳定性
- 关键任务实时性

• 建设效果:

- 轻量级边缘智能组件和人工智能框架
- 调度配置框架
- 事件可靠传输发布

总结与讨论

4

“

谢谢！

T

”