<https://zhuanlan.zhihu.com/p/79380625>

**一、物联网边缘计算环境分析**

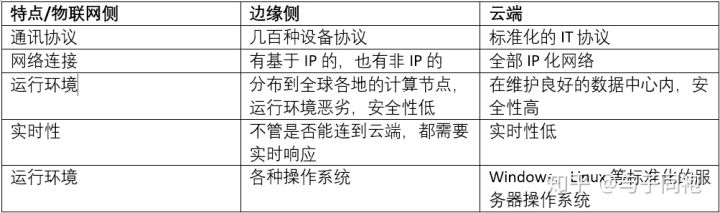
要深入理解边缘计算框架，我们首先要对物联网边缘计算所处的环境有一定的了解。

边缘计算所处的环境与云计算相比，要复杂许多：

1. 从设备连接来看，有上百种协议：MQTT、OPC UA、Modbus、BACnet、KNX等。不同协议应用在不同行业和不同的应用场合。
2. 从业务领域来看，有数据分析、数据管理、安全防护、系统管理等。
3. 从物联网应用开发语言来看，主流的就有Java、Python、NodeJS、.NET、Go、C/C++等。
4. 从边缘计算运行环境来看，有各种操作系统，Windows Core、Ubuntu Core、VxWorks、Yocto、Redhat、Debian、mynewt、FreeRTOS等。



下表是边缘侧与云端的环境特点的对比。

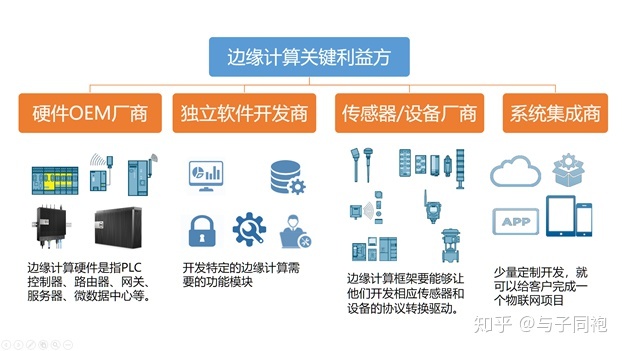


**二、项目干系人分析**

设计边缘计算框架，要考虑到各方利益，让所有人利益最大化，用的人才会多，才能形成良好的生态系统。

因此在设计前，首先要弄清楚边缘计算有哪些利益方。

边缘计算相关的利益方可分为四大类：硬件OEM厂商、独立软件开发商、传感器/设备厂商、系统集成商。



**一是边缘计算硬件OEM商。**边缘计算硬件是指PLC控制器、运动控制器、机器人、路由器、Hub、服务器、微数据中心等。这些厂商会将边缘计算软件移植到自家的硬件上，然后与自己家的云服务对接，然后给客户提供端到端的完成的物联网解决方案。

**二是传感器和设备制造商。**边缘计算框架要能够让他们开发相应传感器和设备的协议转换驱动。

**三是独立软件供应商。**他们的特点是在某一个业务领域很强。他们不需要去关心怎么和底层设备打交道。他们只需要写一些特定的边缘计算需要的功能模块，就可以赚钱。比如MongoDB、FogHorn、RedisEdge等软件厂商。

**四是系统集成商。**他们关心的是可以快速重用现有的功能，加上少量定制开发，就可以给客户完成一个物联网项目。

**三、边缘计算面临的五大挑战**

根据边缘计算所处的环境和相关利益方的诉求，边缘计算面临五大挑战：

1. 各种协议、各种模型、各种格式
2. 边缘侧分析
3. 快速持续的改善
4. 如何让不同供应商提供差异化增值服务
5. 如何最大化资源利用率

**挑战1：各种协议、各种模型、各种格式**

边缘计算是个承上启下的中间环节。它要能够做各种数据转换的工作：

1. 从边缘计算网关到被连接的物这个方向（南向），现场会有各种协议：BACNet, Modbus, OPC-UA，数控机床、机器人、PLC、运动控制器。BLE, Zigbee, ZWave, MQTT, SNMP。要能够处理各种设备连接协议，采集现场各种协议的设备数据，下发命令控制设备。
2. 从边缘计算网关到云平台这个方向（北向），要能够对接各种云平台（阿里云、Azure IoT Hub, AWS IoT Platform, Google IoT Core等）和企业信息系统（ERP，MES，CRM等）。要支持MQTT, AMQP, DDS，RESTful等各种协议。不同客户的云平台，哪怕都是MQTT协议的，MQTT传输的负载都会不同。在把数据发送到云平台这个方向之前，还可能需要对数据进行过滤、聚合、转换、丰富（增加设备相关的元信息、位置信息等）、格式转换（JSON、XML、CSV）、压缩（节省流量）、加密（为了安全）等。相当于边缘侧也要有企业应用集成EAI的功能。
3. 要能够处理各种物联网数据模型和数据格式，像JSON，XML，二进制序列化过的数据格式等。

**挑战2：边缘侧分析**

边缘侧分析是指边缘计算网关把设备数据发送给各种分析软件，然后把分析结果上传到云端，或者根据分析结果直接对现场设备进行本地控制。分析软件可以是Jboss Drools规则引擎、Esper复杂事件处理引擎、商业的FogHorn等。

在边缘侧进行数据分析的原因有：

* 数据都在云端处理传输成本太高。
* 数据都在云端处理存储成本太高。
* 数据都在云端处理的话，响应没有本地处理快。
* 边缘计算网关有时候会断网，连不上云。

边缘侧的智能分析软件从简单到复杂可分为：

1. 基于规则的规则引擎，比如Drools。
2. CEP复杂事件处理引擎，比如Esper。
3. 机器学习和人工智能软件，比如FogHorn。

边缘计算的智能分析要多智能取决于客户的需求和成本。因此边缘计算软件要能足够灵活，能够与上述不同类型的智能分析软件对接。并且需要封装底层的分析软件的复杂功能，让其更简单易用，让非专业人员也会使用，方便落地时实施，降低实施成本。

**挑战3：快速持续的改进**

要能快速响应客户需求和技术发展趋势。要能够持续的改进和更新其中的部分软件组件。

1. **能让边缘计算软件二次开发商持续改进：**比如对于采集物联网设备数据，一个边缘计算设备驱动开发商针对西门子S7-200到S7-1500全系列都开发了对应的采集驱动程序。后来，这家开发商为了提高软件竞争力，为用户提供更好的体验，方便部署人员，他们就根据西门子PLC的通信协议，开发了西门子PLC设备自动发现的功能。另一个例子是，边缘机器学习软件提供商开发了一套边缘侧的机器学习软件，过了几年又有了更强的机器学习算法，边缘计算软件要能够让它方便的改进机器学习算法。
2. **升级部分组件，而不会影响其他组件功能：**边缘计算软件由许多软件组件组成。如果其中有的软件组件需要升级版本，那么不应该影响到其他组件的功能。
3. **随着未来发展，改进性能，从而可以在性能更低的硬件上运行：**比如之前的边缘计算软件模块是用Java开发的，用C语言开发，从而占用内存更少，运行效率更高，可以在更低配置的硬件上运行。
4. **能够和云端的服务进行快速整合：**比如，通过边缘计算的一个agent调用云端的机器学习分析服务。

**挑战4：如何让不同供应商提供差异化增值服务**

边缘计算软件设计时，要考虑到用户，也要考虑到如何让用边缘计算软件进行二次开发的公司实现商业化盈利。**如何让基于边缘计算框架二次开发的公司通过二次开发，提供差异化的增值服务？如何让社区来托管那些重要但是无法商业化的边缘计算框架的功能模块？比如日志服务模块。**

有的厂商擅长机器学习，那就专注基于边缘计算软件开发机器学习软件，比如FogHorn。有的厂商擅长设备对接，那就开发更多的设备协议驱动程序，比如Kepware。有的厂商擅长于云端对接，那就开发能连接各种云服务的程序。有段厂商擅长数据库，那就开发边缘侧的数据库，比如RedisEdge和MongoDB。

**挑战5：如何最大化资源利用率**

从传感器到边缘计算网关到云服务器，有许多不同性能级别的计算、存储、网络、管理、安全资源。

如何部署边缘计算软件，从而适应现场的资源环境，更好的利用这些资源？

# 假设边缘计算软件有如下10项功能：

1. 数据采集、协议解析，比如根据Modbus RTU或OPC协议标准采集工控设备的实时数据

2. 数据上传到云端：把采集的设备实时数据根据指定的格式（如XML或JSON或自定义二进制）和通讯协议（如MQTT/HTTP/WebSocket等）发送到指定的云端服务器；

3. 数据分析：在上传到云端之前，对采集到的设备数据进行处理和分析；

4. 远程控制：接收云端下发的命令，并根据命令执行相应的动作，如修改PLC中某地址的值。

5. 元数据定义：定义网关对应的设备的型号、序列号、采集的数据项定义等信息。设备注册上线离线：定义设备如何注册到云端。

6. 数据告警：配置数据满足条件后的告警。

7. 实时数据库、历史数据库

8. 规则和触发的动作

9. 远程配置下发及远程软件升级

10. 系统管理和配置