

# 边缘计算开源平台 现状分析

## The Present Situation of Open Source Platforms for Edge Computing



梁家越/LIANG Jiayue<sup>1</sup>, 刘斌/LIU Bin<sup>2</sup>, 刘芳/LIU Fang<sup>1</sup>

(1. 中山大学, 广东 广州 510006;

2. 中国科学院电子学研究所, 北京 100190)

(1. Sun Yat-Sen University, Guangzhou 510006, China;

2. Institute of Electronics, Chinese Academy of Sciences, Beijing 100190, China)

**摘要:** 给出了边缘计算平台的一般性功能框架,并基于平台的设计目标和部署方式,对当前典型的边缘计算开源平台进行了分类。从设计目标、目标用户、可扩展性、系统特点和应用领域等方面,对 EdgeX Foundry、Apache Edgent、Azure 物联网(IoT) Edge、CORD 和 Akraino Edge Stack 等典型开源平台进行了对比和分析。最后介绍了可用于构建边缘计算平台的一些开源软件,并讨论了边缘计算平台的未来发展方向和研究热点。

**关键词:** 边缘计算; 开源平台; 物联网; 开源软件

**Abstract:** In this paper, the general functional framework of edge computing platforms is analyzed, and the category of these platforms based on their design target and deployment characteristic is discussed. The existing typical open source edge computing platforms such as EdgeX Foundry, Apache Edgent, Azure Internet of Things (IoT) Edge, CORD and Akraino Edge Stack are introduced with comparison from the perspectives of design target, target user, scalability, system characteristic and application area. Finally, some open source software that can be used to build edge computing platform is introduced, and the future trends and research topics of edge computing platform are discussed.

**Key words:** edge computing; open source platform; IoT; open source software

DOI: 10.12142/ZTETJ.201903002

网络出版地址: <http://kns.cnki.net/kcms/detail/34.1228.TN.20190520.1814.002.html>

网络出版日期: 2019-05-20

收稿日期: 2018-12-30

随着万物联网的趋势不断加深,网络边缘产生的数据在急剧增长。受限于网络带宽增长速度远远跟不上数据的增长数据,传统云计算模式需要解决带宽和延迟这两大瓶颈,边缘计算<sup>[1]</sup>应运而生。边缘计算中的边缘(edge)指的是网络边缘上的计算和存储资源,这里的网络边缘与数据中心相对,无论是从地理距离还是网络距离上来看都更贴近用户<sup>[2]</sup>。作为一种新的计算范式,边缘计算将计算任务部署

于接近数据产生源的网络边缘,利用边缘资源为用户提供大量服务或功能接口,大大减少上传至云数据中心的数据量,缓解网络带宽压力,同时可以更好地解决数据安全和隐私问题。边缘计算平台对边缘计算领域的推广和发展有着重要的意义和影响<sup>[3]</sup>。

### 1 边缘计算开源平台概述

边缘计算系统是一个分布式系统范例,在具体实现过程中需要将

其落地到一个计算平台上,各个边缘平台之间如何相互协作提高效率,如何实现资源的最大利用率,对设计边缘计算平台、系统和接口带来挑战。例如,网络边缘的计算、存储和网络资源数量众多但在空间上分散,如何组织和统一管理这些资源,是一个需要解决的问题。在边缘计算的场景下,尤其是物联网,诸如传感器之类的数据源,其软件和硬件以及传输协议等具有多样性,如何方便有效地从数据源中采集数

据也是一个需要考虑的问题。此外,在网络边缘的计算资源并不丰富的条件下,如何高效地完成数据处理任务也是需要解决的问题。

目前,边缘计算平台的发展方兴未艾。由于针对的问题不同,各边缘计算平台的设计多种多样,但也不失一般性。边缘计算平台的一般性功能框架如图1所示。在该框架中,资源管理功能用于管理网络边缘的计算、网络和存储资源。设备接入和数据采集分别用于接入设备和从设备中获取数据。安全管理用于保障来自设备的数据的安全。平台管理功能用于管理设备和监测控制边缘计算应用的运行情况。

各边缘计算平台的差异可从以下方面进行对比和分析:

(1)设计目标。边缘计算平台的设计目标反映了其所针对解决的问题领域,并对平台的系统结构和功能设计有关键性的影响。

(2)目标用户。在现有的各种边缘计算平台中,有部分平台是提供给网络运营商以部署边缘云服

务;有的平台则没有限制,普通用户可以自行在边缘设备上部署使用。

(3)可扩展性。为满足用户使用动态增加和删除的需求,边缘计算平台需要具有良好的可扩展性。

(4)系统特点。面向不同应用领域的边缘计算开源平台具有不同的特点,而这些特点能为边缘计算应用的开发或部署带来方便。

(5)应用场景。常见的应用领域包括智能交通、智能工厂和智能家居等多种场景,还有增强现实(AR)/虚拟现实(VR)应用、边缘视频处理和无人车等对响应时延敏感的应用场景。

根据边缘计算平台的设计目标和部署方式,可将目前的边缘计算开源平台分为3类:面向物联网端的边缘计算开源平台、面向边缘云服务的边缘计算开源平台、面向云边融合的边缘计算开源平台。

## 2 面向物联网端的边缘计算开源平台

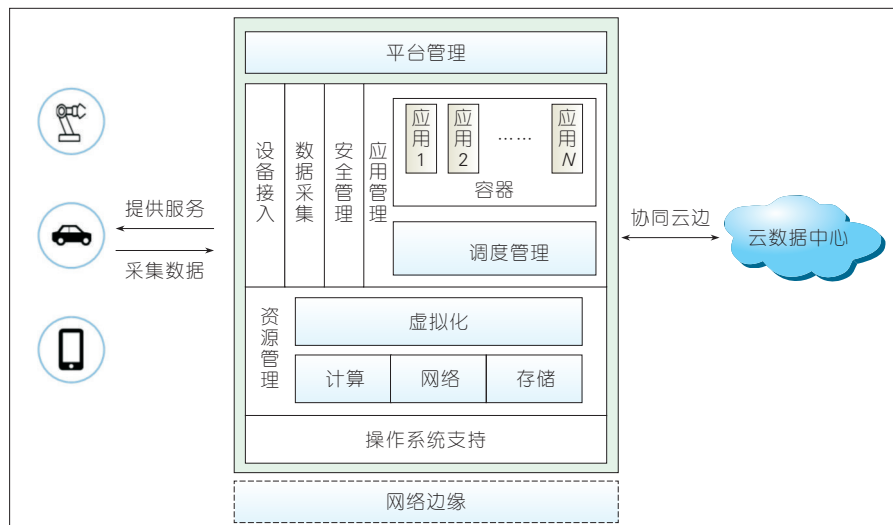
面向物联网端的边缘计算开源

平台,致力于解决在开发和部署物联网应用的过程中存在的问题,例如设备接入方式多样性问题等。这些平台部署于网关、路由器和交换机等边缘设备,为物联网边缘计算应用提供支持。代表性的2个平台是Linux基金会发布的EdgeX Foundry<sup>[4]</sup>和Apache软件基金会的Apache Edgent<sup>[5]</sup>。

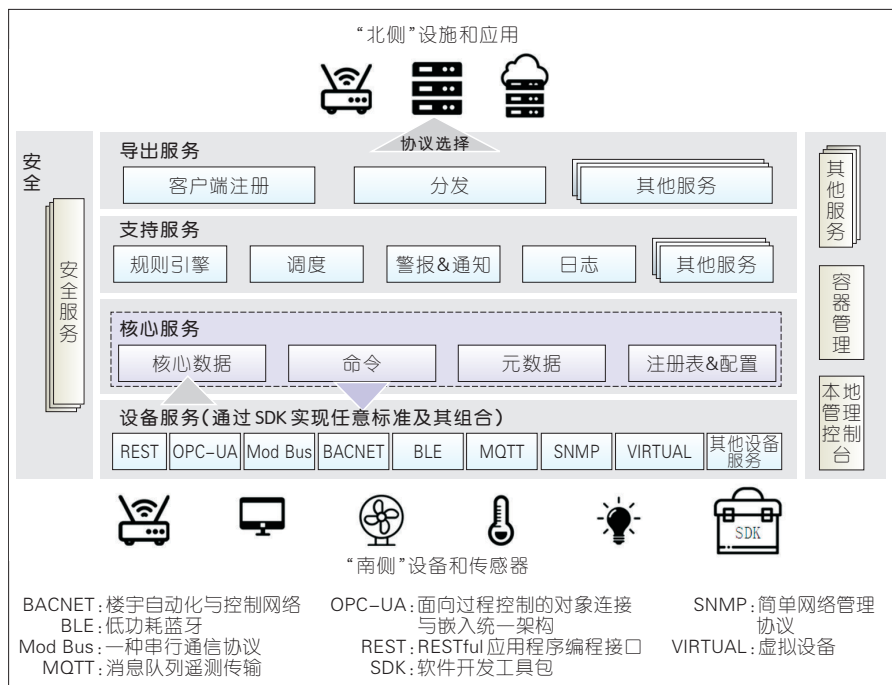
### 2.1 EdgeX Foundry

EdgeX Foundry是一个面向工业物联网边缘计算开发的标准化互操作性框架<sup>[4]</sup>,部署于路由器和交换机等边缘设备上,为各种传感器、设备或其他物联网器件提供即插即用功能并做管理,进而收集和分析它们的数据,或导出至边缘计算应用或云计算中心做处理。EdgeX Foundry针对的问题是物联网器件的互操作性。目前,具有大量设备的物联网产生大量数据,迫切需要结合边缘计算的应用;但物联网的软硬件和接入方式的多样性给数据接入功能带来困难,影响了边缘计算应用的部署。EdgeX Foundry的主旨是简化和标准化工业物联网边缘计算的架构,创建一个围绕互操作性组件的生态系统<sup>[4]</sup>。

图2展示了EdgeX Foundry的架构<sup>[4]</sup>。在图中的最下方是“南侧”,指的是所有物联网器件,以及与这些设备、传感器或其他物联网器件直接通信的边缘网络。在图的最上方是“北侧”,指的是云计算中心或企业系统,以及与云中心通信的网络部分。南侧是数据产生源,而北侧收集来自南侧的数据,并对



▲图1 边缘计算平台一般性功能框架



▲图2 EdgeX Foundry的架构图

数据进行存储、聚合和分析。如图2所示,EdgeX Foundry位于南侧和北侧两者之间,由一系列微服务组成,而这些微服务可以被分成4个服务层和2个底层增强系统服务。微服务之间通过一套通用的Restful应用程序编程接口(API)进行通信。

(1)设备服务层。设备服务层主要提供设备接入的功能,由多个设备服务组成。每个设备服务是用户根据设备服务软件开发工具包(SDK)编写生成的一个微服务。EdgeX Foundry使用设备文件去定义一个南侧设备的相关信息,包括源数据格式、存储在EdgeX Foundry中的数据格式,以及对该设备的操作命令等信息。设备服务将来自设备的数据进行格式转换,并发送至核心服务层。目前,EdgeX Foundry提供了消息队列遥测传输协议

(MQTT)、ModBus串行通信协议和低功耗蓝牙协议(BLE)等多种接入方式。

(2)核心服务层。核心服务层由核心数据、命令、元数据、注册表和配置4个微服务组件组成。核心数据微服务存储和管理来自南侧设备的数据、元数据微服务存储和管理设备的元数据。命令微服务将定义在设备文件的操作命令转换成通用的API,提供给用户以监测控制该设备。注册表和配置微服务存储设备服务的相关信息。

(3)支持服务层。支持服务层提供边缘分析和智能服务,以规则引擎微服务为例,允许用户设定一些规则,当检测到数据满足规则要求时,将触发一个特定的操作。例如,规则引擎可监测控制温度传感器,当检测到温度低于25℃时,触发对空调的关闭操作。

(4)导出服务层。导出服务层用于将数据传输至云计算中心,由客户端注册和分发等微服务组件组成。前者记录已注册的后端系统的相关信息,后者将对应数据从核心服务层导出至指定客户端。

(5)系统管理和安全服务。系统管理服务提供安装、升级、启动、停止和监测EdgeX Foundry微服务的功能。安全服务用以保障来自设备的数据和对设备的操作安全。

最新版本的EdgeX Foundry没有为用户自定义应用提供计算框架,用户可以将应用部署在网络边缘,将该应用注册为导出客户端,进而将来自设备的数据导出至应用来处理。EdgeX Foundry的设计满足硬件和操作系统无关性,并采用微服务架构。EdgeX Foundry中的所有微服务能够以容器的形式运行于各种操作系统,且支持动态增加或减少功能,具有可扩展性。EdgeX Foundry的主要系统特点是为每个接入的设备提供通用的Restful API以操控该设备,便于大规模地监测控制物联网设备,满足物联网应用的需求。EdgeX Foundry的应用领域主要在工业物联网,如智能工厂、智能交通等场景,以及其他需要接入多种传感器和设备的场景。

## 2.2 Apache Edgent

Apache Edgent是一个开源的编程模型和微内核风格的运行时,它可以被嵌入到边缘设备上,用于提供对连续数据流的本地实时分析<sup>[5]</sup>。Edgent解决的问题是如何对来自边缘设备的数据进行高效的分



析处理。为加速边缘计算应用在数据分析处理上的开发过程,Edgent提供一个开发模型和一套API用于实现数据的整个分析处理流程。

基于Java等的开发环境,Edgent应用的开发模型如图3所示。

该模型由提供者、拓扑、数据流、数据流的分析处理、后端系统5个组件组成。

(1)提供者。一个提供者对象包含了有关Edgent应用程序的运行方式和位置信息,并具有创建和执行拓扑的功能。

(2)拓扑。拓扑是一个容器,描述了数据流的来源和如何更改数据流的数据。数据的输入、处理和导出至云的过程都记录在拓扑中。

(3)数据流。Edgent提供了多种连接器以不同方式接入数据源,比如支持消息队列遥测传输(MQTT)、超文本传输协议(HTTP)和串口协议等,用户还可以添加自定义代码以控制传感器或设备的数据输入。

(4)数据流的分析处理。Edgent

提供一系列功能性的API以实现数据流的过滤、分裂、变换等。

(5)后端系统。由于边缘设备的计算资源稀缺,Edgent应用程序无法支撑复杂的分析任务。用户可以使用连接器,通过MQTT和Apache Kafka方式连接至后端系统,或者连接至IBM Watson IoT平台进一步对数据做处理。

Edgent应用可部署于运行Java虚拟机的边缘设备中,实时分析来自传感器和设备的数据,减少了上传至后端系统如云数据中心的数据量,并降低了传输成本。Edgent的主要系统特点是提供了一套丰富的数据处理API,切合物联网应用中数据处理的实际需求,降低应用的开发难度并加速开发过程。Edgent的主要应用领域是物联网,此外,它还可以被用于分析日志、文本等类型的数据。

### 3 面向边缘云服务的边缘计算开源平台

网络运营商的网络边缘,如蜂

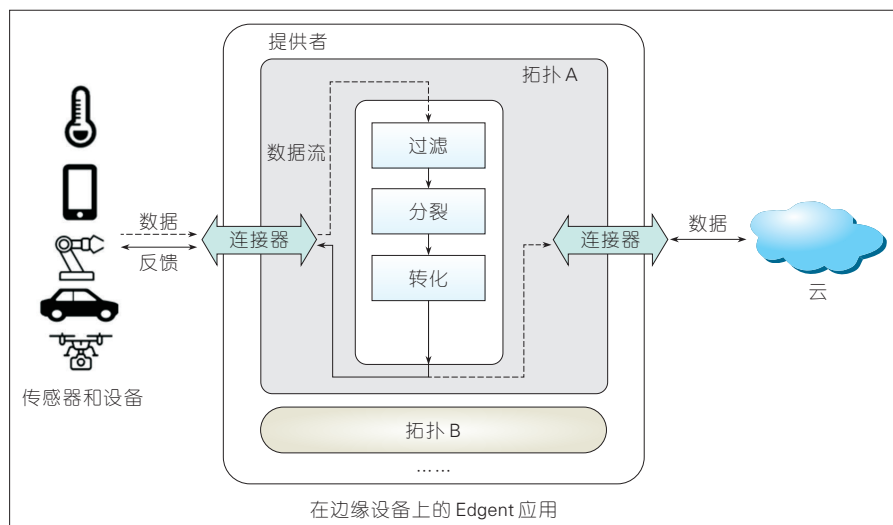
窝网络基站、中央端局和在网络边缘的小型数据中心等,是用户接入网络的地方,其计算、存储和网络资源也可用以部署边缘计算应用。面向边缘云服务的边缘计算平台着眼于优化或重建网络边缘的基础设施以实现在网络边缘构建数据中心,并提供类似云中心的服务。代表性的平台有开放网络基金会(ONF)的CORD项目<sup>[6]</sup>和Linux基金会的Akraio Edge Stack项目<sup>[7]</sup>。

#### 3.1 CORD

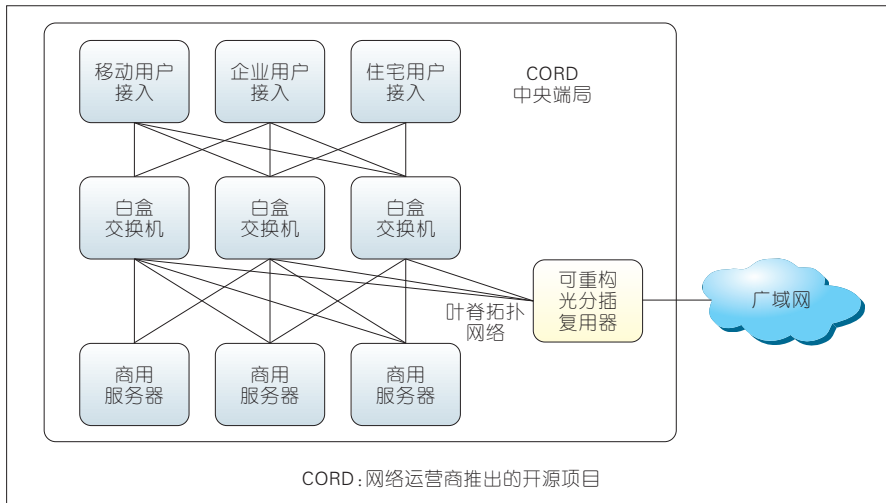
CORD是为网络运营商推出的开源项目,旨在利用软件定义网络(SDN)、网络功能虚拟化(NFV)和云计算技术重构现有的网络边缘基础设施,并将其打造成可灵活地提供计算和网络服务的数据中心。现有网络边缘基础设施构建于由电信设备供应商提供的封闭式专有的软硬件系统,不具备可扩展性,无法动态调整基础设备的规模,导致资源的低效利用。CORD计划利用商用硬件和开源软件打造可扩展的边缘网络基础设施,并实现灵活的服务提供平台,支持用户的自定义应用。

图4为CORD的硬件架构图<sup>[6]</sup>,CORD利用商用服务器和白盒交换机提供计算、存储和网络资源,并将网络构建为叶脊拓扑架构以支持横向网络的通信带宽需求。此外,CORD使用专用接入硬件将移动、企业和住宅用户接入网络中。

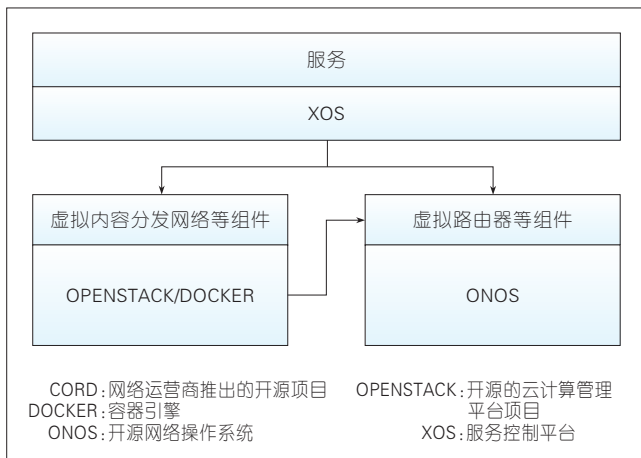
CORD的软件架构<sup>[6]</sup>如图5所示,云平台管理项目OpenStack用以管理计算和存储资源,创建和配置虚拟机以及提供基础设施即服务



▲图3 Edgent应用的开发模型



▲图4 CORD的硬件架构

▲图5  
CORD的软件架构

的服务体验。

对于用户而言, CORD在运营商网络边缘提供的边缘云服务, 具有与云计算相同的优点, 即无需用户提供计算资源和搭建计算平台, 降低了软硬件和管理成本。此外, 有线和无线网络的广泛分布使用户提交边缘计算应用不受地理位置的影响。目前, 运营商正研究如何部署CORD, 中国联通发起成立了CORD产业联盟, 推动CORD项目的落地。

### 3.2 Akraino Edge Stack

Akraino Edge Stack是一个面向高性能边缘云服务的开源项目, 并为边缘基础设施提供整体的解决方案。Akraino Edge Stack致力于发展一套开源软件栈, 用于优化边缘基础设施的网络构建和管理方式, 以满足边缘计算云服务的要求, 例如高性能、低延迟和可扩展性等。

Akraino Edge Stack项目涉及的范围从基础设施延伸至边缘计算应用, 其范围可以划分为3个层面<sup>[7]</sup>。在最上面的应用层面, Akraino Edge Stack致力于打造边缘计算应用程序的生态系统以促进应用程序的开发。中间层面着眼于开发中间件和框架以支持上面层次的边缘计算应用。在这个层面中, Akraino Edge Stack将开发API和框架以接入现有互补性的开源边缘计算项目, 例如上述的面向物联网的互操作性框架EdgeX Foundry, 最大化利用开源社区的现有成果。在最下面的基础设施层面中, Akraino Edge Stack将提供一套开源软件栈用于

(IaaS)功能。开源网络操作系统(ONOS)为网络提供控制平面, 用于管理网络组件如白盒交换网络结构等, 并提供通信服务。容器引擎Docker使用容器技术来实例化提供给用户的服务。服务控制平台XOS用于整合上述软件, 以组装、控制和组合服务。

根据用户类型和使用案例的不同, CORD可被具体实现为M-CORD、R-CORD和E-CORD。以M-CORD为例, M-CORD面向无线网络(特别是5G网络), 使用蜂窝网络将用户接入。M-CORD基于

NFV和云计算技术将蜂窝网络功能进行分解和虚拟化, 实现网络功能的动态扩展同时增强资源利用率。在此基础上, M-CORD支持多接入边缘服务, 为用户提供定制服务和差异化体验质量(QoE)。此能力切合具备移动性的边缘计算应用的需求, 能通过无线网络为手机、无人车和无人机等移动设备的边缘计算应用就近提供强大的计算能力。R-CORD和E-CORD同样可以在网络边缘支持住宅用户或企业用户的边缘计算应用, 如VR和AR应用等, 以获得更快的响应时间和更好

优化基础设施。此外, Akraino Edge Stack 为每种使用案例提供蓝图以构建一个边缘计算平台。每个蓝图涵盖上述3个层次的声明性配置,其中包括对硬件、各层面的支撑软件、管理工具和交付点等的声明。

Akraino Edge Stack 基于使用案例提供边缘云服务,可部署于电信运营商的塔楼、中央端局或线缆中心等。其应用领域包括边缘视频处理、智能城市、智能交通等。

#### 4 面向云边融合的边缘计算开源平台

云计算服务提供商是边缘计算的重要推动者之一,基于“云边融合”的理念,致力于将云服务能力拓展至网络边缘。目前,亚马逊公司推出了 AWS Greengrass<sup>[8]</sup>,微软公司推出了 Azure IoT Edge<sup>[9]</sup>,阿里云公司推出了物联网边缘计算平台 Link IoT Edge<sup>[10]</sup>,微软公司在 2018 年宣布将 Azure IoT Edge 开源。

Azure IoT Edge 是一种混合云和边缘的边缘计算框架,旨在将云功能拓展至如路由器和交换机等具备计算能力的边缘设备上,以获得更低的处理时延和实时反馈。Azure IoT Edge 运行于边缘设备上,但使用与云上的 Azure IoT 服务相同的编程模型;因此,用户在开发应用的过程中除对计算能力的考量外,无需考虑边缘设备上部署环境的差异,可以将云上原有的应用迁移至边缘设备上运行。

如图 6 所示, Azure IoT Edge 由 IoT Edge 模块、IoT Edge 运行时和 IoT Edge 云界面组成,前两者运行

在边缘设备上,后者则是一个在 Azure 云上提供服务的管理界面。

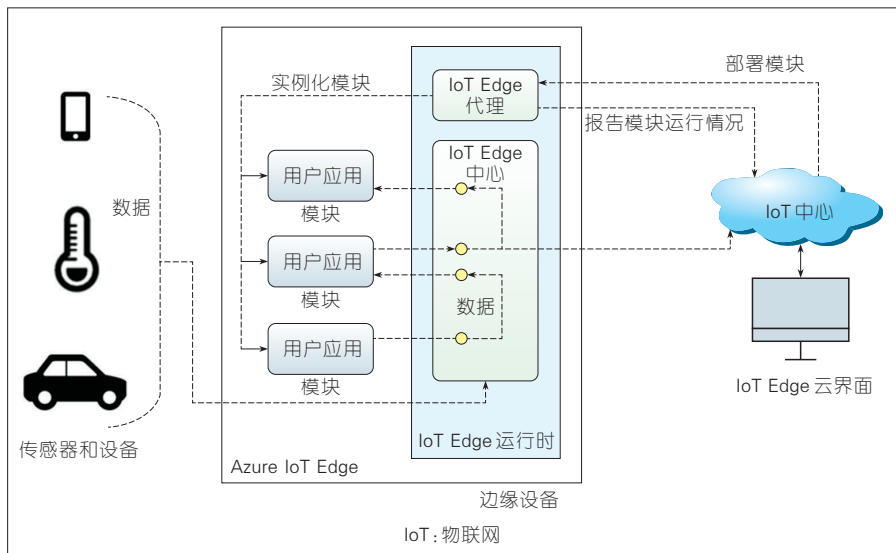
(1)IoT Edge 模块。IoT Edge 模块对应于用户的边缘计算应用程序。一个模块镜像即一个 Docker 镜像,模块里包含用户的应用代码,而一个模块实例就是一个运行着对应的模块镜像的 Docker 容器。基于容器技术, IoT Edge 具备可扩展性,用户可动态添加或删除边缘计算应用。由于相同的编程模型, Azure 机器学习和 Azure 数据流分析等 Azure 云服务也可以部署到 IoT Edge 模块,此特性便于在网络边缘部署复杂的人工智能应用,加快了开发过程。

(2)IoT Edge 运行时。IoT Edge 运行时由 IoT Edge 中心和 IoT Edge 代理 2 个组件构成,前者负责通信功能,后者负责部署和管理 IoT Edge 模块,并监测控制模块的运行。IoT 中心是在 Azure 云上的消息管理中心, IoT Edge 中心与 IoT 中心连接并充当其代理。IoT Edge 中

心通过 MQTT、高级消息队列协议 (AMQP) 和 HTTPS 协议获取来自传感器和设备的数据,实现设备接入的功能。此外, IoT Edge 中心作为消息中转站,连接 IoT Edge 模块之间的消息通信。IoT Edge 代理从 IoT Hub 接收 IoT Edge 模块的部署信息,实例化该模块,并保证该模块的正常运行,如对故障模块进行重启,并将各模块的运行状态报告至 IoT 中心。

(3)IoT 云界面。云界面提供了设备管理的功能。用户通过云界面进行添加设备,部署应用和监测控制设备等操作,为用户大规模部署边缘计算应用提供了方便。

Azure IoT Edge 的主要系统特点是有强大的 Azure 云服务的支持,尤其是人工智能和数据分析服务的支持。Azure IoT Edge 具有广阔的应用领域,除了物联网场景,原有在云上运行的应用也可以根据需求迁移至网络边缘上运行。目前 Azure IoT Edge 已有智能工厂、智能



▲ 图 6 Azure IoT Edge 的架构图



灌溉系统等使用案例。

## 5 构建边缘计算平台的开源软件

一些开源软件能为边缘计算平台提供组件功能,具体包括:

(1)网络管理。开源网络自动化平台(ONAP)<sup>[11]</sup>是一个为物理或虚拟网络功能的自动化和实时策略驱动化编排提供平台的开源项目,在边缘云中可用于定制网络服务。Linux 基金会于2018年将ONAP与其他5个功能类似的开源项目合并成 LF Networking Fund<sup>[12]</sup>项目。ONOS<sup>[13]</sup>是一个开源网络操作系统,它提供一个控制平面用于管理网络组件,运行软件程序和模块,并为终端主机提供通信服务。目前,ONAP应用于 Akraino Edge Stack 项目中,ONOS应用于CORD项目中。

(2)容器技术。Docker作为一个开源的应用容器引擎,被多个边缘计算平台使用以提供灵活的应用部署方式。Kubernetes<sup>[14]</sup>是一个对容器化应用进行自动化部署、扩展和管理的开源项目,可用于边缘计算平台中以提供可靠和可扩展的容器编排。

(3)云平台。OpenStack是一个开源的云计算管理平台项目,可用于构建边缘云,管理边缘云基础设备的计算、存储和网络资源。StarlingX<sup>[15]</sup>是一个用于构建分布式边缘云的开源项目,提供一套完整的云基础架构软件栈,现应用于 Akraino Edge Stack 项目中。

(4)人工智能技术。引入人工智能技术是边缘计算的发展趋势之

一。Acumos 人工智能(AI)<sup>[16]</sup>是一个用于训练、部署和共享AI应用程序的开源框架,可用于边缘云中提供AI边缘计算应用的快速开发和部署。Acumos AI目前已用于 Akraino Edge Stack 项目中。

## 6 结束语

边缘计算平台为基于边缘计算的应用提供了一种降低延迟、提高数据处理实时性的架构和软件栈支撑。目前,边缘计算平台仍处于快速发展时期,本文中我们对其未来的发展趋势进行如下分析:

(1)面向特定应用场景的专用性边缘计算平台将迅速发展。目前,智能家居、智能工厂和智慧城市都是受关注的应用场景,如何系统化支持这些场景的功能需求是边缘计算平台的发展趋势之一。

(2)边缘计算平台的数据安全与隐私保护问题将变得更加突出。边缘设备相互协作时的数据安全如何保证,边缘平台如何避免暴露用户的位置隐私信息、尤其是边缘计算设备的计算资源,存储能力有限,使得边缘计算平台的安全防御具有更高的挑战性。

(3)边缘计算平台将结合更多人工智能技术。目前,以深度学习为代表的人工智能技术发展十分迅速,但“智能边缘”的研究才刚刚起步,未来的边缘计算平台将引入人工智能技术,实现边缘数据的智能处理。

### 参考文献

- [1] SHI W, CAO J, ZHANG Q, et al. Edge Computing: Vision and Challenges[J]. IEEE Internet of Things Journal, 2016, 3(5):637-646. DOI: 10.1109/JIOT.2016.2579198

- [2] 赵梓铭, 刘芳, 蔡志平, 等. 边缘计算: 平台、应用与挑战[J]. 计算机研究与发展, 2018, 55(2): 327-337. DOI: 10.7544/issn1000-1239.2018.20170228
- [3] 施巍松, 刘芳, 孙辉, 等. 边缘计算[IM]. 北京: 科学出版社, 2018:3-6
- [4] EdgeX Foundry[EB/OL]. [2018-12-30]. <https://www.edgexfoundry.org>
- [5] Apache Edgent[EB/OL]. [2018-12-30]. <http://edgent.apache.org>
- [6] CORD[EB/OL]. [2018-12-30]. <https://www.opennetworking.org/cord/>
- [7] Akraino Edge Statck[EB/OL]. [2019-02-22] [2018-12-30]. <https://wiki.akraino.org>
- [8] AWS Greengrass[EB/OL]. [2018-12-30]. <https://aws.amazon.com/greengrass/>
- [9] Azure IoT Edge[EB/OL]. [2018-12-30]. <https://azure.microsoft.com/zh-cn/services/iot-edge/>
- [10] 物联网边缘计算[EB/OL]. [2018-12-30]. <https://www.aliyun.com/product/iotedge?spm=5176.cnioledge.IOT.7.483f61efVjFy39>
- [11] ONAP[EB/OL]. [2018-12-30]. <https://www.onap.org/>
- [12] LF Networking Fund[EB/OL]. [2018-12-30]. <https://www.lfnetworking.org/>
- [13] ONOS[EB/OL]. [2018-12-30]. <https://onosproject.org/>
- [14] Kubernetes[EB/OL]. [2018-12-30]. <https://kubernetes.io/>
- [15] StarlingX[EB/OL]. [2018-12-30]. <https://www.starlingx.io/>
- [16] Acumos AI[EB/OL]. [2018-12-30]. <https://www.acumos.org/>

### 作者简介



梁家越, 中山大学数据科学与计算机学院在读硕士研究生; 研究方向为边缘计算。



刘斌, 中国科学院电子学研究所在读博士生; 研究方向为边缘计算。



刘芳, 中山大学数据科学与计算机学院副教授; 研究方向为计算机系统结构与存储技术; 先后主持和参加国家自然科学基金、科技部重点研发、“863”计划等项目10余项; 获中山大学“百人计划”中青年杰出人才, 并获军队科技进步二等奖1次; 发表论文70余篇, 授权国家发明专利10余项。