

面向 IoT 的边缘节点平台架构^①



邢照庆¹, 崔允贺^{1,2}, 吕晓丹^{1,2}, 钱清³, 申国伟^{1,2}, 赵建朋⁴

¹(贵州大学 计算机科学与技术学院, 贵阳 550025)

²(贵州大学 省部共建公共大数据国家重点实验室(筹), 贵阳 550025)

³(贵州财经大学 信息学院, 贵阳 550025)

⁴(国家电网浙江省电力有限公司, 杭州 310007)

通信作者: 崔允贺, E-mail: yhcui@gzu.edu.cn

摘要: 针对边缘节点体系结构支持功能单一、模块划分混乱、向云端传输数据中的带宽开销大以及数据传输安全问题, 设计开发了一套面向 IoT 的边缘节点平台架构. 该平台架构包括设备管理、数据管理、资源管理、应用管理以及导出服务管理 5 个模块, 使用最小二乘法等算法在边缘侧实现数据采集、数据过滤、数据加密、指标分析等功能. 测试结果表明, 该平台能够无间断收集周围环境信息, 分析环境数据, 能够降低边缘侧向云端传输数据的成本, 同时保证数据安全可靠传输.

关键词: 边缘节点; 物联网 (IoT); 最小二乘法; 数据处理; 边缘计算

引用格式: 邢照庆, 崔允贺, 吕晓丹, 钱清, 申国伟, 赵建朋. 面向 IoT 的边缘节点平台架构. 计算机系统应用, 2022, 31(5): 85-93. <http://www.c-s-a.org.cn/1003-3254/8532.html>

IoT-oriented Edge Node Platform Architecture

XING Zhao-Qing¹, CUI Yun-He^{1,2}, LYU Xiao-Dan^{1,2}, QIAN Qing³, SHEN Guo-Wei^{1,2}, ZHAO Jian-Peng⁴

¹(School of Computer Science and Technology, Guizhou University, Guiyang 550025, China)

²(State Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, China)

³(School of Information, Guizhou University of Finance and Economics, Guiyang 550025, China)

⁴(State Grid Zhejiang Electric Power Co. Ltd., Hangzhou 310007, China)

Abstract: The structure of the edge node system is exposed to problems of the single function, the confusing module division, and the high bandwidth overhead, and security risks in edge-to-cloud data transmission. To solve these problems, this study designs and develops an IoT-oriented edge node platform architecture. The platform architecture consists of five modules, i.e., device management, data management, resource management, application management, and exporting service management. The designed platform architecture applies algorithms such as the least square method to achieve functions including data collection, data filtering, data encryption, and indicator analysis on the edge side. The experimental results indicate that the platform can continuously collect surroundings information, filter and analyze data, and ensure the safe and reliable transmission of data while reducing the cost of edge-to-cloud data transmission.

Key words: edge node; Internet of Things (IoT); least square method; data processing; edge computing

随着越来越多的企业和个人从传统的 IT 系统向基于云的服务迁移, 近年来云数据中心数量不断增加. 为提供高可用的计算能力, 云数据中心大多运行在专

用的基础设施硬件上, 如高性能处理器、内存模块和大规模存储阵列^[1]. 此外, 云数据中心通过集中式数据库^[2] 或基于分布式计算框架 (如 MapReduce^[3]) 处理提

① 基金项目: 国家自然科学基金 (61902085); 贵州省科技计划 (黔科合基础 [2020]1Y267); 贵州大学引进人才科研项目 (贵大人基合字 (2019)52 号)

收稿时间: 2021-08-09; 修改时间: 2021-09-13; 采用时间: 2021-10-19; csa 在线出版时间: 2022-02-25

供海量的数据处理分析服务。尽管云计算能够提供动态的资源管理方法,基于云的运营模式能够优化成本以提高其竞争力,但在物联网(Internet of Things, IoT)、自动驾驶、智能家居等许多场景中,云计算模式也存在一些缺点。例如,在IoT中,基于云计算的数据处理需要将终端设备产生的大量数据传输至云端,占用大量网络带宽。此外,云数据中心与物联网终端设备之间的传输时延降低了终端业务响应速度^[4]。因此,云计算模式无法为海量物联网设备提供快速业务响应,例如自动驾驶、智能家居等对实时性要求较高的场景。此外,在一些首先考虑数据隐私和安全的场景中,例如终端设备接入场景,对于终端设备身份认证的不足可能导致数据的泄露,此外在智能制造场景中,不同设备之间各种通信协议混杂,边缘侧无法对数据做统一安全处理,数据长距离传输可能使基于云数据中心的数据分析存在安全风险^[5]。

为解决数据长距离传输导致的成本上升、业务响应时间长以及安全性差的问题,边缘计算应运而生。边缘计算^[6]是指在靠近边缘设备的一侧,开发部署集网络、计算、存储、应用核心能力为一体的平台,就近提供边缘侧数据处理服务。这种计算模式将原先在数据中心处理的数据转移至边缘侧进行处理,减轻了数据中心的处理压力的同时也提高了数据处理的速度,保证了数据的安全性并且提高了服务质量。

目前针对边缘节点平台架构,国内外众多学者展开了研究。例如:2018年ONF发布了CORD—专注于提供边缘计算服务的平台^[7]。CORD能够在蜂窝基站、边缘小型数据中心等用户接入网络的地方提供类似于云端的服务。微软在2017年发布了Azure IoT Edge^[8]。Azure IoT Edge通过在边缘设备上部署应用程序提供边缘计算服务,用户可以通过物联网边缘云接口自主部署边缘应用程序。2018年Apache基金会发布了开源项目Apache Edgent^[9],专注于边缘IoT数据的分析,旨在加速数据分析平台的开发。Apache Edgent使用连接器获取外界数据流,经过过滤、分割、转换进行数据流的处理,最后将初步处理的数据流发送到云端进行进一步处理。

上述边缘节点平台架构或自主开发,或利用已有软件进行二次开发,但具有以下缺点:(1)体系结构中的模块功能设计混乱,缺乏一些功能模块。例如在Apache Edgent中缺乏安全管理模块,使其缺乏边缘

IoT设备的认证功能,无法保证接入边缘IoT设备的合法性,使得边缘端向云端的数据传输缺乏保护机制,容易造成边缘数据丢失、泄露等问题。(2)处理的数据种类单一。例如Apache Edgent只能处理边缘IoT设备的结构化ZigBee协议数据,无法检测边缘节点CPU、内存、磁盘使用率以及空闲率等边缘节点自身数据。同时,Apache Edgent只能简单判断数据是否损坏、验证数据格式的正确性,无法对边缘损坏数据做拟合修复等细粒度的处理。

当前面向IoT的边缘节点平台架构存在功能模块设计混乱、数据处理方式单一等问题。针对这些问题,本文在设计边缘节点过程中面临如下挑战:(1)边缘节点功能模块划分及模块交互;(2)边缘IoT数据处理流程编排。为解决上述挑战,本文按照平台功能对模块进行划分,设计模块交互过程,设计了包含数据过滤、数据清洗、数据质量、数据统计、数据存储等过程的数据处理流程,保证数据处理的合理性。总之,本文主要贡献总结如下:(1)本文设计开发了面向IoT的边缘节点平台架构,包含设备管理、数据管理、应用管理、资源管理、导出服务5大模块,具有较强的通用性,不受特定IoT场景的限制。(2)本文设计的边缘节点平台架构对数据处理流程进行了合理编排,使得边缘节点可以处理各种场景的IoT数据,实现了对边缘侧各种场景IoT数据预处理。

1 面向IoT的边缘节点平台架构

1.1 边缘节点平台架构功能设计

为使边缘节点能够满足不同IoT场景下不同硬件设备的功能需求,本文采用了模块化、集成化的设计思路,以增强边缘节点的功能扩展性和可靠性。本文所设计的边缘节点平台架构包括设备管理模块,数据管理模块,安全管理模块,资源管理模块,导出服务模块5个模块,平台功能模块设计如图1所示。

该架构中,南侧设备由各种IoT设备(传感器等)组成,负责采集环境数据,并且通过RS485、WiFi以及ZigBee通信协议将IoT设备数据传输到边缘节点。

设备管理模块用于对IoT设备进行身份认证,对接入平台的IoT设备工作状态进行监测,统计IoT设备的各项信息并生成设备台账。

数据管理模块将边缘节点接收的不同数据帧格式转换为统一的数据帧格式;通过数据清洗重复、缺

失、异常的数据进行删除、补全以及修复,保证数据安全以及检索方便对数据进行分类统计,将数据进行处理维护,进行存储分类,保证数据的安全以及完整性.

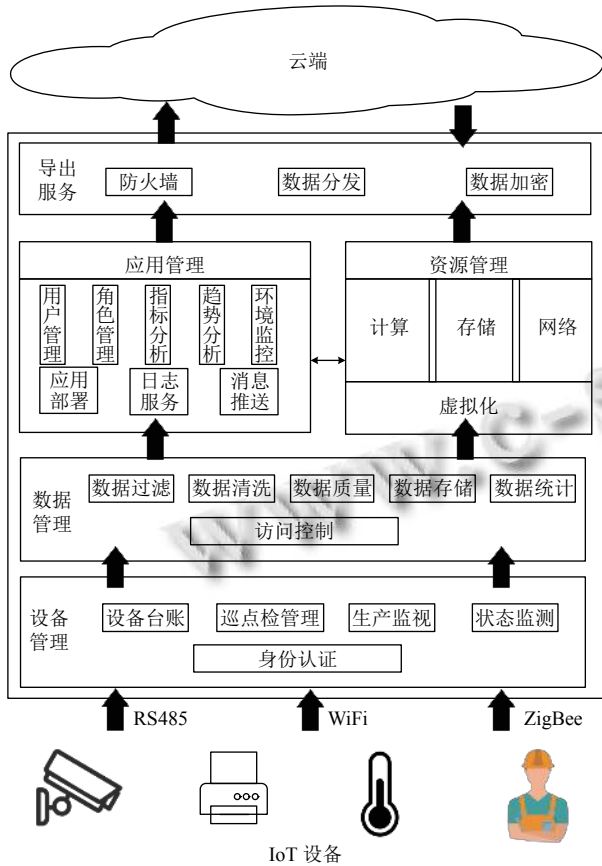


图1 平台功能模块设计图

应用管理模块包括系统应用、数据应用以及用户应用:系统应用记录平台运行日志,对平台消息对用户进行消息推送,根据具体用途对应用进行部署;数据应用分析设备管理模块以及数据管理模块传输的数据,发现并解决其中存在的问题;用户应用对平台用户账号以及权限进行管理,记录用户信息并保障平台内部数据安全.

导出服务模块在边缘节点出口处设置包过滤防火墙,对出口数据进行分析,并通过非对称数据加密算法保障输出数据的安全性,随后将边缘节点过滤、分析后的有价值数据发送到云端.

资源管理模块负责对边缘节点计算、网络、存储资源信息进行监测,通过监测边缘节点的CPU空闲率、内存使用率、网络状态以及磁盘使用状态,对边缘节点运行的负载情况进行可视化展示,帮助运维人

员快速做出决策,保证边缘节点正常运转.

1.2 数据流设计

IoT环境数据是本系统的数据主体,系统依靠数据传输和处理连接各个功能模块.数据流图如图2所示,本文将边缘设备监测的IoT环境信息转换为stream数据流形式,将监测数据以数据流的形式在系统中传输处理,依此设计了数据流图.边缘传感器收集到IoT环境数据后,首先使用设备管理模块、数据管理模块、资源管理模块、应用管理模块处理收到的数据,最后使用导出服务模块将处理过的数据传输到其它边缘节点或者云端.数据流传输过程中使用安全管理模块保障数据安全.具体步骤如步骤1-步骤6所示.

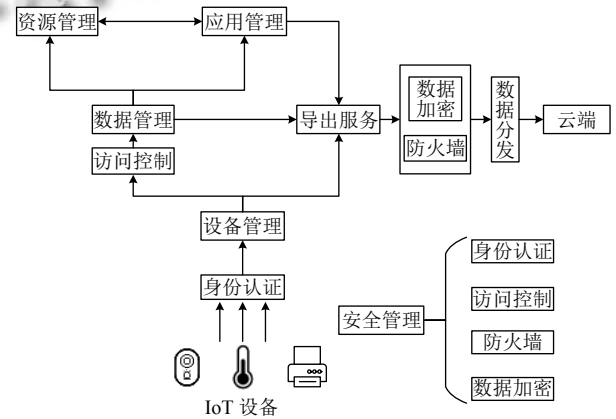


图2 数据流向图

步骤1. IoT设备将采集到的数据传输到身份认证子模块,由身份认证子模块进行身份认证,认证通过后将采集的数据传输至设备管理模块;

步骤2. 设备管理模块将IoT设备传输的数据传输到访问控制子模块,设备类型、设备ID等设备相关数据传输到导出服务模块;

步骤3. 访问控制子模块对访问用户以及传输数据分析之后将数据传输到数据管理模块;

步骤4. 数据管理模块通过执行数据过滤、数据清洗、数据统计、数据存储等操作,随后将数据分别传输至应用管理模块、资源管理模块及导出服务模块;

步骤5. 应用管理模块和资源管理模块双向传输数据,应用管理模块对数据做进一步深层次的数学分析,资源管理模块负责监测边缘节点内部计算、网络、存储资源,随后应用管理模块和资源管理模块将处理后的数据传输至导出服务模块;

步骤6. 导出服务模块将数据传输到数据加密子模

块以及防火墙子模块, 随后通过数据分发子模块将数据传输到云端.

2 边缘节点平台架构模块设计

2.1 设备管理模块设计

设备管理模块主要包括数据获取、设备台账、身份认证、状态监测等功能. 南侧设备以及传感器收集到的数据经过身份认证进入节点, 确保数据来源正常. 为了保证设备正常运行, 对设备进行状态监测、巡点检管理等, 最后将监测信息写入设备台账, 对边缘设备做全面管理. 该模块具体实现逻辑如图3所示.

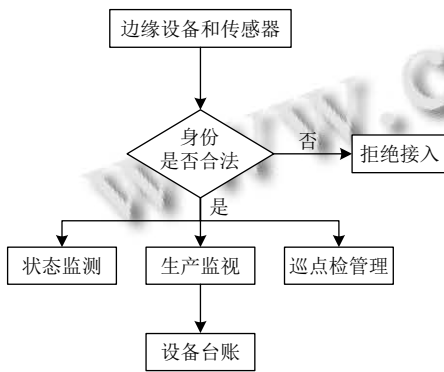


图3 设备管理模块逻辑图

为全面收集工业现场的各种环境信息, 本文布置了各种低功耗温湿度传感器, 并且以此构建无线传感器网络用于环境数据的采集传输, 另外在靠近数据源的地方建立了边缘节点, 用于环境数据的就近处理和实时监控.

图4所示为本文使用的无线传感器网络拓扑, 其以自组织的方式部署微节点形成传感器网络, 通过配置协调器节点和终端节点实现身份认证, 协调器和终端通过协调PIN ID、发射信道、发射功率等参数建立安全连接, 完成传感器网络的组网过程. 上述微节点集成了传感器和通信模块, 通过部署传感器, 这些微节点可以直接监测周围的环境.

本文使用RS485温湿度传感器采集温湿度数据, 使用EBYTE公司的E180-DTU(ZG250-485)无线数传电台作为传感器网络通信模块. 同时, 使用ZigBee协议将传感器收集到的数据发送到传感器网络中的路由节点, 随后通过路由节点将数据发送到边缘节点. 此外, 本文采用工控机作为边缘节点, 通过在工控机中部署

Web服务器和数据库, 为运维人员提供可视化的监控服务并对数据进行存储.

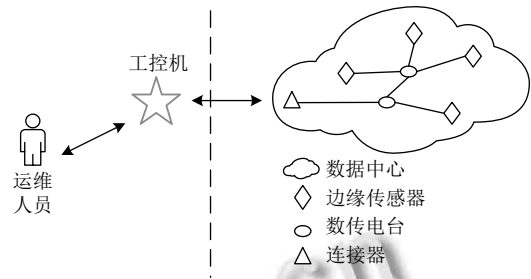


图4 无线传感器网络拓扑

2.2 数据管理模块设计

数据管理模块包括数据过滤、数据清洗、数据存储等功能. 边缘节点通过串口通信程序接收到IoT传感器发送的数据帧后, 首先判断采集到的IoT传感器的数据是否为标准数据帧的格式. 对于不合法的数据帧, 数据管理模块通过数据质量模块取出应答帧中有效数据位, 并将十六进制的原始数据转换为十进制的环境信息数据. 之后通过数据清洗、数据过滤功能筛选出异常数据, 建立数据统计表并存入MySQL数据库进行持久化存储. 该模块具体实现逻辑如图5所示.

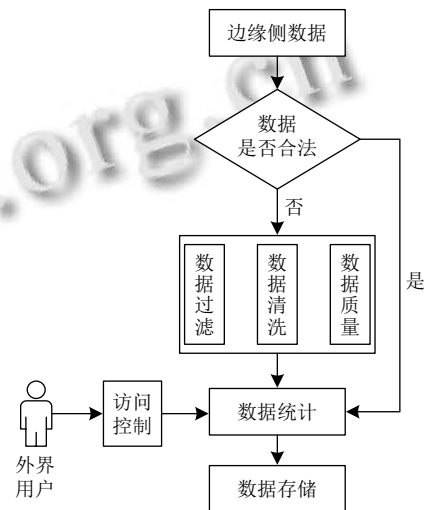


图5 数据管理模块逻辑图

由于外界环境的干扰, 容易造成IoT设备数据值存在缺失或者数据帧格式错误, 对数据统计影响较大. 因此本文在数据清洗功能中通过对数据帧解码, 观察数据变化的规律建立了最小二乘法^[10]的数学模型, 对数据进行预处理. 接收到的数据帧包括地址码、功能

码、有效数据以及校验码, 首先将数据帧转化为字符串类型数据, 并将其中有效数据提取出来, 通过数据解码将十六进制数据转换为十进制数据, 并对数据建立拟合曲线。

拟合曲线中有较多的值存在误差, 为使拟合数据和真实数据离差平方达到最小, 本文使用了最小二乘法进行数据拟合。在拟合过程中, 首先在坐标系中描出数据对应的点, 根据数据点的分布拟定回归曲线, 使得回归曲线上的点与对应数据点的纵坐标的离差平方和达到最小即为误差最小, 最终得出传感器数据点的最优的匹配函数, 如式(1)所示:

$$\varphi = \min \sum_{i=1}^n Q_i^2 = \min \sum_{i=1}^n [F(x_i) - y_j]^2 \quad (1)$$

其中, Q_i^2 是拟合函数上点的纵坐标与真实数据的离差平方, 其和达到最小, 则误差达到最小; $F(x_i)$ 为拟合曲线函数; y_j 为真实数据值, 通过观察真实数据值得到规律, 确定 $F(x_i)$ 的函数类型以及系数 k_j , 从而得到 φ 的表达式为:

$$\varphi = \min \sum_{i=1}^n Q_i^2 = \min \sum_{i=1}^n [ax + b - y_j]^2 \quad (2)$$

最后通过 R^2 判断拟合程度, 保证数据值与真实数据值误差保持在较低水平, 提高监测的准确度, 帮助运维人员对于环境做出正确的决策^[11]。

2.3 应用管理模块设计

应用管理模块管理对象主要包括数据应用、用户应用以及系统应用。用户应用通过基于角色的访问控制策略对平台中不同用户权限进行划分, 划分不同用户能够访问的数据域, 并由管理员统一进行管理; 数据应用通过设置定时器任务实现环境监控功能, 显示工业现场当前环境信息, 包括当前周围环境的温度、湿度等信息, 并根据时间以及异常数据产生个数得到异常数据信息产生的速率对数据做指标分析; 系统应用使用日志服务记录平台所有用户操作, 为之后问题回溯提供支持, 管理员通过消息推送将平台信息告知平台用户并且可以根据实际需求对应用进行部署, 提高平台的可用性。该模块具体设计如图6所示。

为在边缘侧对数据进行预处理, 提取出有价值的信息, 本边缘节点平台架构设计了指标分析以及趋势分析功能。指标分析通过统计一个定时器任务时间 T 内的异常数据产生个数 N 得出异常数据产生速

率 θ , 并对数据做可视化展示。通过对数据规律观察总结设定 T 为180 min。其表达式为:

$$\theta = \frac{N}{T} \quad (3)$$

趋势分析功能通过计算一个定时器任务时间内的数据最大值、最小值、平均数以及中位数对数据趋势进行分析, 并以可视化图表的方式进行展示, 帮助运维人员做出正确决策。

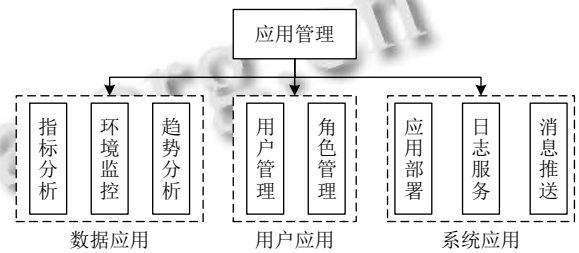


图6 应用管理模块设计图

2.4 资源管理模块设计

资源管理模块主要实现对于边缘节点计算、存储等资源的监控管理功能。通过资源管理模块将边缘节点资源虚拟化, 平台管理员可以获得当前边缘节点的CPU、内存、服务器以及磁盘状态等信息。通过这些信息, 平台管理员可以对当前边缘节点负载情况做出快速决策, 并及时对资源进行合理调度, 调整边缘节点负载情况, 保证平台总体高可用性。该模块具体设计如图7所示。

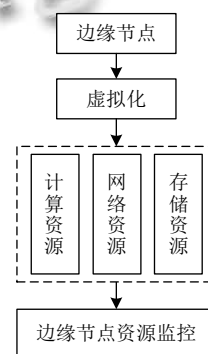


图7 资源管理模块逻辑图

资源管理模块通过Java虚拟机(Java virtual machine, JVM)对边缘节点的计算、存储资源进行虚拟化, 屏蔽掉与具体平台相关的信息, 使得本平台可以在多种平台上运行。监测的计算资源包括CPU核心数、用户使用率、系统使用率以及当前空闲率; 存储资源包括磁



图 11 设备台账

为保证边缘节点平台架构的稳定性及可靠性,本文设计了资源监控模块,通过监测边缘节点本身的 CPU 占用率、内存使用率 PC 主机信息,运维人员可以对边缘节点的计算、存储、网络资源做统一调度,保证边缘节点运行的稳定性.资源监控如图 12 所示.



图 12 资源管理

本文使用折线图和柱状图将传感数据可视化.运维人员选择不同的温湿度传感器,可以快速生成温湿度折线图.如图 13、图 14 所示,温湿度图可以自动显示最大值、最小值和平均值.为对数据做更进一步的分析,本文还设计了指标分析模块,统计一定时间序列内的异常数据数量和产生速率,并以柱状图的形式表示,方便运维人员确定不同时间段的环境信息.

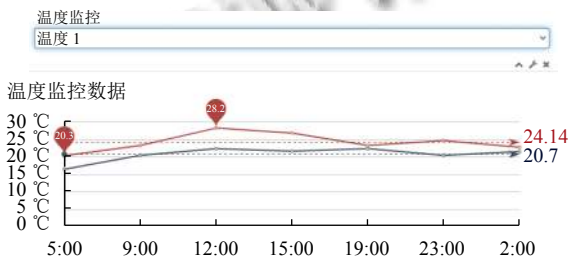


图 13 温度监控

本文使用 IBM Watson Platform 作为云端,如图 15 所示,通过身份认证登录到 IBM Watson Platform, IBM Watson Platform 接收 IoT 数据信息,并且显示异常数

据接收的时间、数据、数据类型,将这些数据做长期的、大规模的分析处理,并且做出决策.

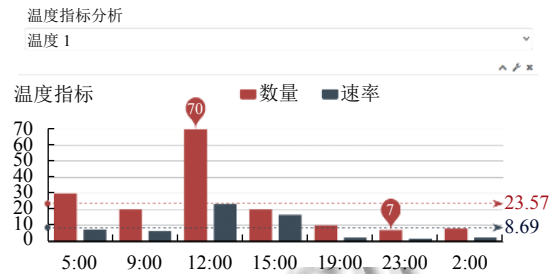


图 14 温度指标分析

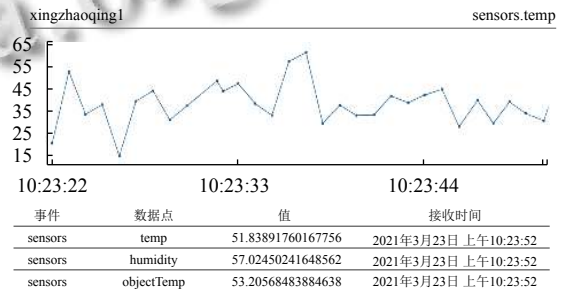


图 15 导出服务

4 平台功能测试与分析

4.1 平台功能分析

与 Apache Edgent、Azure、CORD 等现有边缘节点平台相比,本文根据 IoT 场景的功能需求,清晰地对平台功能模块进行了划分,设计了面向 IoT 的边缘节点平台架构.如表 1 所示,该边缘节点平台架构具有设备管理、数据管理、应用管理、资源管理以及导出服务功能,比上述现有边缘节点平台功能更全面,更能够适用于不同的 IoT 场景.

表 1 边缘节点平台功能模块分析

Edge computing	功能模块				
	设备管理	数据管理	应用管理	资源管理	导出服务
CORD	N	Y	N	Y	N
Azure	N	Y	Y	Y	N
Edgent	N	Y	N	N	Y
本文	Y	Y	Y	Y	Y

4.2 平台环境搭建

根据系统的功能需求,本文对系统环境进行了部署,如图 16 所示,通过温湿度传感器终端设备收集周围环境信息,将终端设备接入 ZigBee 无线传感器网络

中,通过协议转换等过程将数据流传输到工控机中,通过边缘节点系统的数据分析处理等过程,在PC终端将监控信息进行显示。

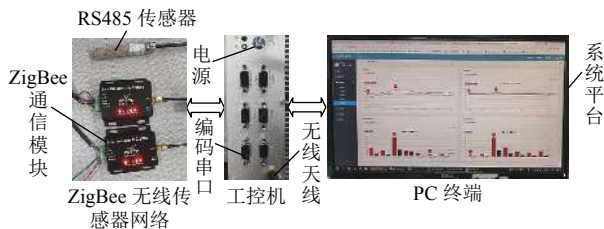


图 16 边缘节点系统环境

4.3 实验测试

图 17 为所设计的边缘节点平台监测边缘节点本身的数据的示意图,该边缘节点平台能够监测的边缘节点自身的 CPU、内存以及磁盘的总量、已用部分和剩余部分等数据。如图所示,本实验中,边缘节点 CPU 总量为 12 个,可用 10.58 个,空闲率为 88.17%,内存总量为 15.83 GB,可用为 8.17 GB,空闲率为 51.61%,磁盘总量为 123.3 GB,可用为 117.5 GB,空闲率为 95.29%。因此当前边缘节点可以正常运行处理数据,处于低负载状态。通过对边缘节点本身进行监测运维人员可以轻松地判断边缘节点负载情况,并对边缘节点的任务合理的迁移。

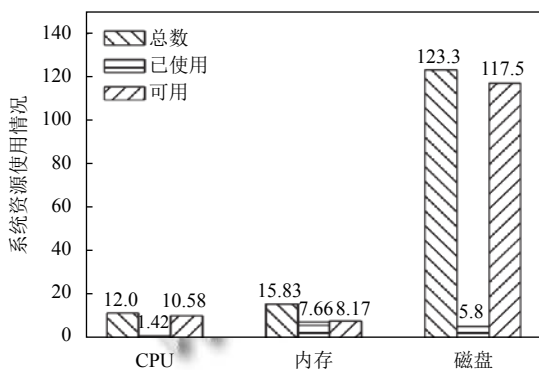


图 17 边缘节点监测数据

图 18、图 19 分别为对传感器的温度、湿度监测数据的数学分析。本文收集数据的时间为 2021 年 4 月 12 日 09:21:32–2021 年 4 月 12 日 11:03:12,在此期间对某数据中心机房的温湿度进行了监测,监测期间共记录了 298 条数据。

本文对数据进行了更进一步的数学分析,结果表明:温度基本稳定在 23–25℃ 之间,湿度基本稳定在

44%–46% 之间,标准差分别为 0.612 5 和 0.139 0,因此数据中心机房温湿度处于比较稳定的状态,因温湿度而造成的设备损坏以及短路风险较低,有利于服务器的正常运转。

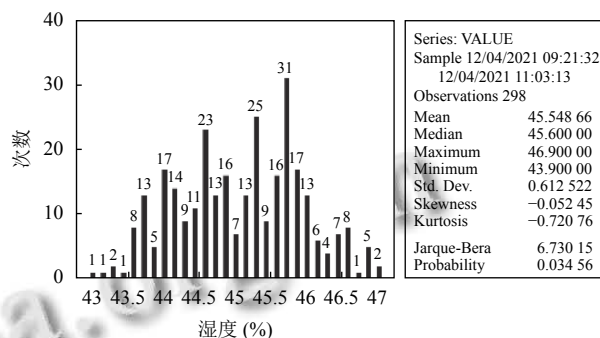


图 18 数据中心湿度数据分析

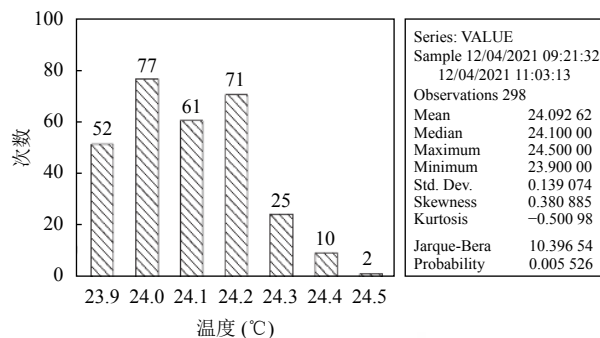


图 19 数据中心温度数据分析

5 结语

本文设计了面向 IoT 的边缘节点平台架构,通过构建无线传感器网络对工业现场周围的环境信息进行实时获取,并在边缘工控机中进行数据处理分析,最终通过可视化的方法在边缘 PC 终端进行显示。不仅增强了数据的安全性,保障数据不丢失,而且在边缘侧处理大量的数据,降低了云端的数据处理量和传输时延。

在未来的工作中,将在边缘节点中设计用于任务调度的模块,从而提高数据分发的服务质量,减少资源开销,并在真实环境中验证其有效性。

参考文献

1 Wang YZ, Shen YL, Su CC, et al. CFHider: Control flow obfuscation with Intel SGX. IEEE Conference on Computer Communications (IEEE INFOCOM 2019). Paris: IEEE, 2019. 541–549.

- 2 Cheng K, Shen YL, Wang YZ, *et al.* Strongly secure and efficient range queries in cloud databases under multiple keys. IEEE Conference on Computer Communications (IEEE INFOCOM 2019). Paris: IEEE, 2019. 2494–2502.
- 3 Wang YZ, Shen YL, Jiang XH. Practical verifiable computation—A mapreduce case study. IEEE Transactions on Information Forensics and Security, 2018, 13(6): 1376–1391. [doi: [10.1109/TIFS.2017.2787993](https://doi.org/10.1109/TIFS.2017.2787993)]
- 4 Wang YZ, Shen YL, Wang H, *et al.* MtMR: Ensuring mapreduce computation integrity with merkle tree-based verifications. IEEE Transactions on Big Data, 2018, 4(3): 418–431. [doi: [10.1109/TBDATA.2016.2599928](https://doi.org/10.1109/TBDATA.2016.2599928)]
- 5 Cheng K, Wang LM, Shen YL, *et al.* Secure k-NN query on encrypted cloud data with multiple keys. IEEE Transactions on Big Data, 2021, 7(4): 689–702.
- 6 边缘计算产业联盟. 边缘计算与云计算协同白皮书 (2018年). <http://www.econsortium.org/Lists/show/id/335.html>. [2021-08-05].
- 7 Cord. <https://www.opennetworking.org/cord>. 2018. [2021-08-05].
- 8 Azure IoT. <https://azure.microsoft.com/en-us/overview/iot/>. 2018. [2021-08-05].
- 9 Apache Edgent. <https://incubator.apache.org>. 2018. [2021-08-05].
- 10 楚彬. 稳健整体最小二乘算法与应用研究 [硕士学位论文]. 成都: 西南交通大学, 2015.
- 11 张政国, 吴艾玲. 最小二乘小波支持向量机在电力负荷预测中的应用. 兰州交通大学学报, 2016, 35(4): 65–71. [doi: [10.3969/j.issn.1001-4373.2016.04.013](https://doi.org/10.3969/j.issn.1001-4373.2016.04.013)]
- 12 ECHARTS. <https://echarts.apache.org/zh/index.html>. (2002-06-13)[2021-08-05].