Research on Application of Edge Computing in Real-time Environmental Monitoring System

Yunhao Teng¹, Jiapo Cui¹ and Wenrong Jiang²*
¹College of Engineering, Shanghai Polytechnic University, Shanghai, 201209, China
²Corresponding author’s e-mail: wrjiang@sspu.edu.cn

Abstract. In order to solve the problem of insufficient data processing ability and high delay of cloud computing in environmental monitoring system, combined with edge computing technology, this paper designs an environmental monitoring system which can process data in real time. The system uses the Baetyl edge computing frame to build an edge node on Raspberry Pi, the monitoring node collects environmental data such as temperature and humidity and sends the data to the edge node. The edge node will upload the monitoring data to Baidu Intelligent Cloud Platform after processing. The test results show that edge computing has obvious advantages in the application of environmental monitoring.

1. Introduction
The Internet of Things is another information technology revolution following the Internet. Environmental monitoring is a typical field in which IoT technology is applied. The application of the Internet of Things promotes informatization and is an important means to cultivate and develop the strategic new environmental protection industries[1], and it is of far-reaching significance in promoting the development of environmental protection in China.

With the rapid development of the Internet of Things and industrial upgrading, higher requirements have been put forward for the evolution of technology. With the increase of monitoring nodes in the real-time environmental monitoring system, the amount of data increases dramatically, which brings problems such as transmission congestion and increased computational complexity to the network, and the time delay in data transmission and calculation increases accordingly, which also brings new challenges to the real-time performance of the environmental monitoring system[2].

Edge computing is used to process environmental data near the edge of the data source, and the task of data processing is sunk to the edge, which is more efficient and ensures real-time performance. Using the advantages of edge computing, it can reduce response time, improve data processing capacity, reduce the network bandwidth and ensure data security. It has the characteristics of high reliability, low delay and high real-time[3], which can optimize the overall quality of data and improve the utilization efficiency under the premise of efficient processing, and has high application value.

2. Baetyl edge computing framework
Edge computing framework should deploy multiple edge nodes on the side close to the user, and these edge nodes are required to have certain storage capacity and computing capacity to deal with the general user needs and provide some simple services [4]. Because Edge computing can greatly optimize the computing model and improve the quality of service, it has been listed as an important topic in the Internet of Things industry in the academic world. Some Edge computing open source systems have also come out one after another, such as EdgeX Foundry released by Linux Foundation, Apache Edgent.
released by Apache Software Foundation, AWSGreengrass launched by Amazon and Link IoT Edge launched by Alibaba Cloud.

Baetyl is an open edge computing framework of Linux Foundation Edge, which aims to extend cloud computing, data and service seamlessly to edge devices [5]. It provides temporary offline and low-latency computing services, including data access, message routing, function computing, stream computing, AI inference and other functions. With the latest open source Cloud platform Baetyl-Cloud, application deployment, configuration issuance, system monitoring and other functions can be realized. Baetyl provides a complete "cloud management, edge operation and edge and cloud integration" integrated solution.

In Baetyl 1.0 version, the system runs in Docker containerization mode, consisting of master process, container service and remote management. The master process runs directly on the local operating system outside the container, it exposes a series of management control interfaces to the outside world, and saves and translates the management operations into Docker API primitives. Baetyl provides several official functional services that run as containers within Docker. Different services have different specific functions, resources used, and development languages, the Agent service is responsible for receiving remote console commands and forwarding them to the master process for OTA upgrades, the MQTT Hub service is responsible for the data connection of all modules and IoT devices, the Remote service is responsible for docking data to third parties, the Function Manager provides function calculation in multiple languages, etc., Baetyl, however, just needs to start and monitor them as standard containers.

![Figure 1. Architecture of Baetyl 1.0](image-url)

On July 8, 2020, Baetyl 2.0 was officially released. Baetyl 2.0 simplifies the original three levels of "master process, container service and remote management" to two levels of "service + remote management". Edge and remote management frameworks all evolve to Cloud Native, all local processes are running in Kubernetes, and a new remote management system Baetyl-Cloud is added, which supports the management of multiple edge nodes. The Cloud Management Suite is responsible for managing all resources, including nodes, applications, configurations, deployments, and more. It offers a wealth of applications. In order to facilitate the expansion of functions and access to third-party
services, all functions are implemented in the form of plug-ins. The Edge Computing Framework runs on Kubernetes on the Edge node, it manages and deploys all applications on the node, and provides a variety of functions through application services. Applications include system applications and common applications, system applications are all provided by Baetyl official, users do not need to configure.

![Figure 2. Architecture of Baetyl 2.0](image)

3. Environmental monitoring system design

3.1. Monitoring Node Design

The monitoring node adopts Arduino Uno as the main control panel, uses DHT11 temperature and humidity sensor to collect temperature and humidity data, MQ-2 gas sensor to collect combustible gas data, and ESP8266 as the communication module. The monitoring node is connected to the LAN through WiFi, and the temperature and humidity sensors and gas sensors collect environmental data, convert the collected data into JSON format, and then transmit it to the edge node under the same LAN through MQTT protocol.

**MQTT (Message Queuing Telemetry Transport), is a messaging technology designed for the Internet of Things, It is an open, lean, lightweight, and easy to implement protocol, especially suitable for embedded devices and mobile terminals with low bandwidth, unstable network, high cost of network and limited processor and memory resources [6].**

![Figure 3. Architecture of monitoring node](image)
3.2. Edge node design
The hardware of the edge node is built by using Raspberry Pi 4B. Raspberry Pi is a single-board computer as large as a bank card. It uses an ARM chip and has a number of external interfaces to plug into peripherals such as a screen, camera, keyboard and mouse. The Raspberry Pi Generation 4, released in 2019, is equipped with a 1.5GHz 64-bit ARM A53 processor and a maximum of 8G of memory. The built-in GPU supports H.265 video streaming with 4K resolution. In addition, it also integrates Wi-Fi, Gigabit wired network card and low-power Bluetooth, and the USB ports are also extended to 4 [7].

Figure 4. Function diagram of edge nodes

The edge node near the user side communicates with the monitoring node below and connects with the cloud server above, it is the core component of the edge computing mode in the system. The Baetyl edge computing framework built in this system mainly plays the role of data receiving, data forwarding and data processing. Data reception is implemented by the Baetyl-Broker application, a standalone version of the message subscription and publication center, it uses the MQTT3.1.1 protocol and can provide reliable message delivery services over low-bandwidth, unreliable networks. It serves as message-oriented middleware on the side of the Baetyl framework, providing message-driven interconnection capabilities for all services. Data processing and data forwarding are implemented by Kuiper-Kubernetes-Tool plug-in. Kuiper is a SQL-based lightweight edge streaming messaging engine that can run on resource-constrained edge devices for real-time processing of messages from Internet of Things devices. In order to adapt to Baetyl platform, Kuiper has launched an adapter plug-in, Kuiper-Kubernetes-Tool. In terms of data processing, Kuiper uses SQL to implement business logic. It calculates the average, maximum, minimum and times of temperature every 10 seconds, and groups them by device ID. Data forwarding can be done by adding an MQTT-type action to the configuration file of Kuiper-Kubernetes-Tool. The data processed by Kuiper is forwarded to the IotCore in the cloud.

4. System test and analysis
After completing the construction of Baetyl edge computing framework and real-time environment monitoring system, the network performance of the system was tested, mainly using packet loss rate and average delay as test indexes, and compared with the traditional cloud computing mode. The packet loss rate is obtained by counting the packets sent by the analog device and the packets received. The user sends acquisition instructions to the monitoring node, and the monitoring node collects data and uploads it to the device service layer. The delay is defined as from the time when the acquisition instructions are sent to the device service layer or the server of the cloud platform receives the data, so as to test the data transmission performance of the edge computing framework. The actual packet loss rate and average delay of the system obtained by the Ping test method are shown in the table.
Table 1. System actual packet loss rate test

| Computing mode | Number of data packets |
|----------------|------------------------|
|                | 100 200 300 400 500 600 700 |
| Packet loss in cloud computing mode | 0 1 1 2 1 2 1 |
| Packet loss in edge computing mode | 0 0 0 1 1 0 1 |

Table 2. System average delay test

| Computing mode | Number of data packets |
|----------------|------------------------|
|                | 100 200 300 400 500 600 700 |
| Average delay in cloud computing mode (ms) | 117 115 127 123 115 130 128 |
| Average delay in edge computing mode (ms) | 33 31 31 38 35 38 36 |

As can be seen from the data in the table, with the increase of the number of sent packets, the number of packet loss and delay in the two calculation modes increase slightly. In the traditional cloud computing mode, the average packet loss rate and delay are 0.29% and 124ms respectively, while in the edge computing mode, these two data are 0.11% and 35ms respectively, which reduces the average packet loss rate by about 62% and the average delay by about 72%. This indicates that the edge computing mode significantly optimizes the packet loss rate and real-time performance of the system, and improves the IoT performance of the system, which is the same as the expected results of the experiment.

5. Conclusion

In the era of the Internet of Everything, the cloud computing model is gradually unable to adapt to the development of emerging technologies in the Internet of Things industry due to the explosive growth of data volume. Under this background, the edge computing model comes into being. In this paper, the edge calculation model is applied to real-time environment monitoring system, we set up a Baetyl edge computing framework on the edge nodes to process the environmental data collected by the monitoring nodes, in the software design of the edge node, we used the Kuiper streaming data analysis engine to carry out streaming calculation on the environmental data, and obtained the average value, maximum and minimum value and times in units of 10 seconds, then uploaded the results to Baidu Intelligent Cloud Platform, and finally gave a test of the system performance. Edge computing makes the real-time environment monitoring system run more safe and reliable, and the network delay is lower, the application at the edge is expandable. The application of edge calculation in environmental monitoring has a good prospect.

Acknowledgments

This work was supported by the Collaborative Innovation Platform of Electronic Information Master under Grant No.A10GY21F015 of Shanghai Polytechnic University.

References

[1] Meng, H., Zhang, J., Yi, X., Xue, J. (2013) Application of IoT in environmental monitoring. Environmental Science and Management, 38: 10-12+86.
[2] Idrees, Z., Zou, Z., Zheng, L. (2018) Edge computing based IoT architecture for low cost air pollution monitoring systems: a comprehensive system analysis, design considerations & development. Sensors, 18: 3021.
[3] Shi, W., Cao, J., Zhang, Q., Li, Y., Xu, L. (2016) Edge computing: Vision and challenges. IEEE internet of things journal, 3: 637-646.
[4] Wang, S., Xu, J., Yang, K., Deng, Q. (2020) Research on Application of Edge Computing and Perceptual Fusion Techniques in Smart Self-service Coffee Machine. Journal of Chinese Computer Systems, 41: 1451-1457.
[5] Baetyl. (2019) Baetyl’s documentation. https://docs.baetyl.io/en/latest/
[6] Naik, N. (2017) Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP. In: IEEE International Systems Engineering Symposium (ISSE). Vienna. pp. 1-7.

[7] BinMasoud, A., Cheng, Q. (2019) Design of an IoT-based Vehicle State Monitoring System Using Raspberry Pi. In: International Conference on Electrical Engineering Research & Practice (ICEERP). Sydney. pp.1-6.