

Resource Management for Reliable IoT Environments

Siwoo Byun

Anyang university, Kyoungkido, South Korea
swbyun30@daum.net

Abstract

Sensor network consists of a large number of sensor nodes that combine physical sensing capabilities with networking capabilities. The objective of this research is to introduce the base architectures of IoT sensor network, fog computing and edge computing technology. In this paper, a new approach of handling IoT data is proposed for efficient data service in IoT environment. Proposed scheme can transmit compressed sensor data by reducing overhead in advance through column-based data compression.

Keywords: *Sensor data, Column-based compression, Resource control*

1. Introduction

Advances in computing and networking technologies have made extensive use of small information sensor devices [Figure 1]. Each information device could include its management tools and a small database. Flash memory is one of the best candidates for data management in ubiquitous computing environments.

Sensor database is in integral component of the increasing reality of the Internet of Things (IoT) environment [1][2][3][4]. Much of the data transmitted is sensor data. The huge volume of data produced and transmitted from sensing devices can provide a lot of information but is often considered the next big data challenge for businesses. Recently, various sensor database systems, including TinyDB and Cougar, have been developed for the efficient management of sensor data in the USN environment.



Figure 1. Examples of IoT nodes

Article history:

Received (January 26, 2020), Review Result (March 1, 2020), Accepted (April 9, 2020)

IoT edge computing is defined as a distributed computing infrastructure that includes a large number of sensor devices that are well connected to each other. It is a new paradigm that provides high computing resources close to distributed IoT devices [Figure 2]. Therefore, IoT edge node can collect, classify, and analyze raw data streams locally instead of transferring them to the remote cloud, significantly reducing traffic overheads and speeding up the IoT big data process. However, where IoT edge nodes should be placed to facilitate communication between IoT devices and the edge nodes is still an important issue.

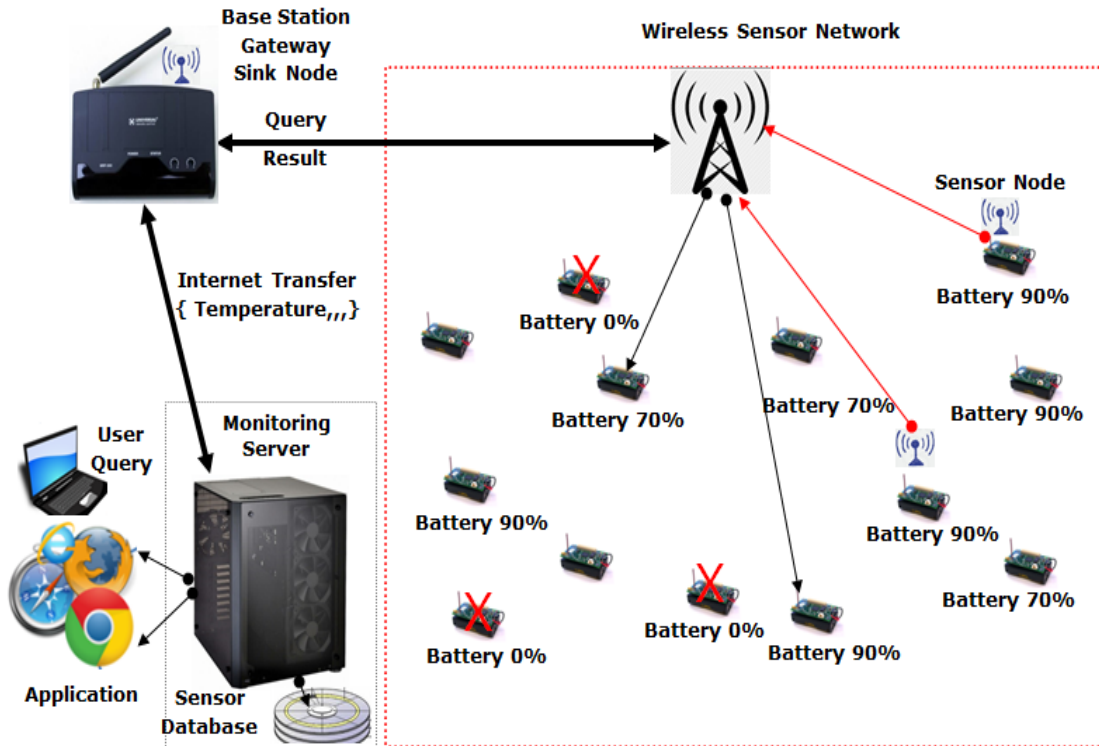


Figure 2. Example of IoT sensor environment

To use IoT application as an example [Figure 3], a widely deployed security sensor consists of a smart lock, a video/audio recorder, and various sensor monitors (e.g. lighting sensors, occupancy classification sensors, motion sensors, etc.). These security devices are difficult to combine unless they are from the same vendor. However, edge computing can seamlessly services security applications. That is, edge computing can provide related service such as 1) unified interface for integrating all kinds of independent devices, 2) flexible resources to support storage and computation, and 3) real-time processing and low-latency response.



Figure 3. Example IoT application

Once the corresponding platform is installed, each sensor is connected to the client. These service applications can be installed on independent virtual machines, and they can also perform advanced processing logic in VMs, which can handle the data shared by these secure monitor applications. For example, if a motion sensor detects suspicious movements in a room, a cleaning robot equipped with a video camera is ordered to check out the exact location. Real-time video analytics will process these videos and check whether it is a false alarm. Then, notification and report will be sent to the homeowner, and the security system will report it to the police station.

2. Related works

Though cloud computing offers some advantages for both users and service providers, it also has certain limitations. The prominent limitations of cloud computing include requirement of high-speed reliable Internet connectivity and sometimes multi-homing requirements to avoid outages, high latency, undefined security etc. New trends in networking such as large sensor networks, mobile data networks and real-time streaming applications have characteristics that cloud computing cannot satisfy.

Since the cloud system is located within the Internet, there is no central control. Moreover, there are many problems with network topology and heterogeneous networks. A major problem that seriously affects service quality is the network delay [Figure 4]. Real-time applications are significantly affected by delay and jitter due to the network latency. However, it is very difficult to control delay and jitter from latency in large heterogeneous networks on an Internet scale. Therefore, the problems caused by latency are not going to be resolved soon.

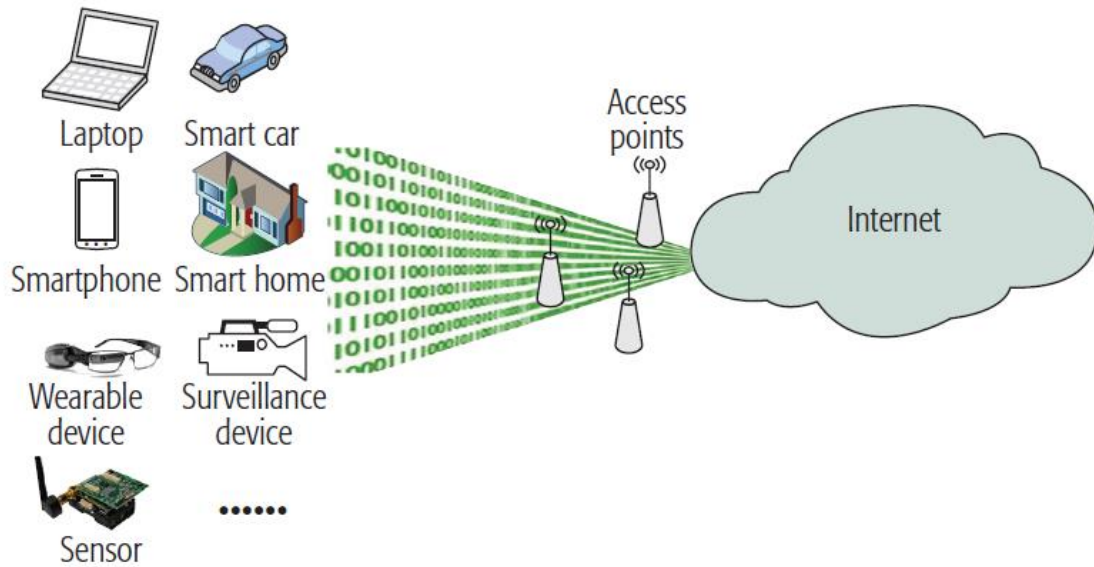


Figure 4. Example IoT network delay

Another problem with cloud computing is data security and privacy [5]. In cloud platforms, data transmission and system responses traverse many intermediate networks, depending on the distance between users and systems. Thus, if customer data is located in the cloud over the public Internet, there is a risk that customer integrity and confidentiality will be compromised. That is, cloud systems now face a variety of security threats due to their implementation within the Internet.

Edge computing pushes data and computing power from centralized host to network edges. Edge computing offers high levels of scalability, reliability and fault tolerance as well as sub-second response to end users. Advantages of edge computing include a significant reduction in data transfer across the network, reduced congestion, reduced latency, elimination of centralized bottlenecks, improved security of encrypted data, and reduced exposure to hostile elements.

3. Proposed model

In general, an IoT network consists of three components: a sensor device, an IoT gateway, and a cloud network, each meaning a data source, a data communication network, and data processor [5][6].

[Figure 5] shows the proposed IoT architecture based on the edge computing [7][8][9]. IoT Edge computing servers are closer to end users than cloud servers which is located far away. Although IoT edge computing servers provide weaker computing power than cloud servers, it has better quality of service (QoS) and lower network latency for end users. In general, the structure of IoT edge computing can be divided into three types: front-end, near-end and far-end [6].

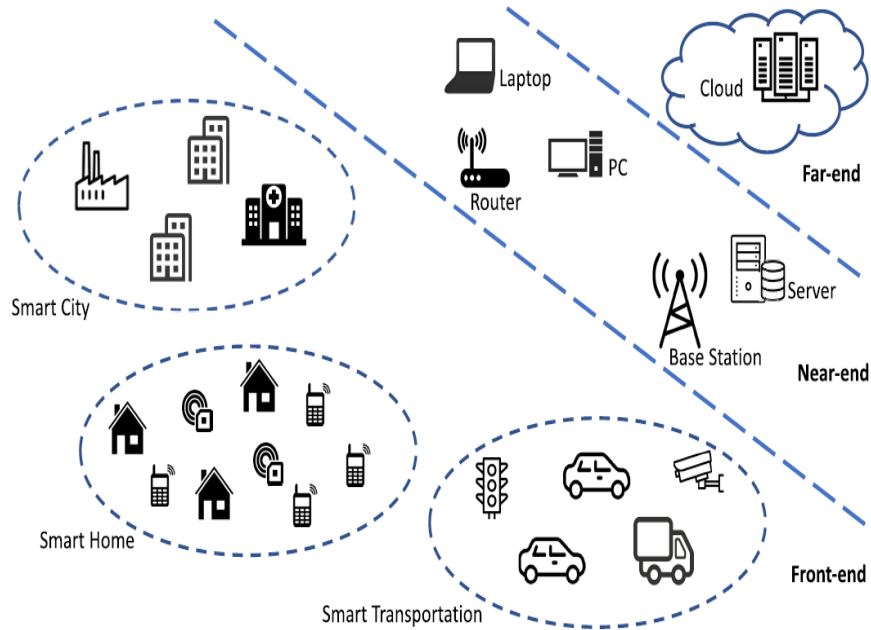


Figure 5. Architecture of IoT network model

Traditional row-based storage systems have to read many unrelated data from data repository, so they have performance limitations when handling a number of concurrent, diverse queries. However, in column-based storage model, only the columns related to IoT query need to be taken from the data repository, which is densely compressed to improve read efficiency like the column-store [10][11].

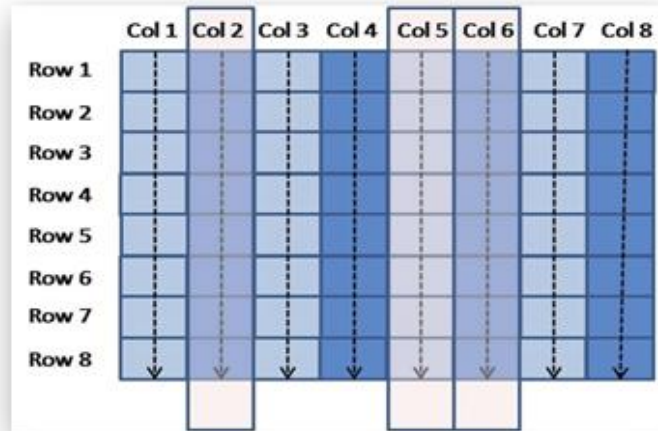


Figure 6. Architecture of column-based storage

Since IoT devices have very limited storage space, all data collected or generated should be sent to the storage server. If all devices simultaneously transfer data to cloud storage, this will be a significant burden to the network. Instead, sending data to several edge storage based on the characteristics of the edge topology reduces the long-distance traffic.

For this, load balancing and resource allocation technologies for edge-based storage are essential. It monitors different storage demand rates and uses data replication for traffic and storage balancing. Additionally, selecting the nearest edge storage node and weighted data control, can reduce the storage access time.

IoT edge-gateway shares fast local network with front-end sensor devices. Thus, IoT edge gateways provide much faster transmissions as compared to cloud servers connected to the slow Internet. However, the IoT edge gateway has limited memory and storage space as compared to the cloud servers.

Although most sensor data are initially processed with the maximum I/O performance, if the storage device or memory becomes overloaded, the following data is not processed at that speed. Therefore, it is needed to prevent this overloading and to keep data availability and reliability. The approach of the study aims to increase data distribution speed across the entire network and to increase data availability and reliability by adding an IoT gateway called fog-gateway in the entry point of the cloud servers.

If the edge and fog gateways are efficiently combined, they provide faster network transmission and higher data efficiency for overall IoT network. That is, it is effective to increase the communication efficiency of edge gateways to increase the storage efficiency of fog gateways, to offload sensor data that is not needed on edges, and to send it to the cloud. In this respect, this study proposes a new management framework called Compression-based Offloading Management to improve the availability and reliability of IoT data.

Edge gateways perform data-based offloading to reduce system overheads such as I/O delay, severe transmission delay, memory shortage, and CPU overload. Second, to enhance the data availability and reliability, the edge gateways are interconnected to the fog gateways below the cloud layer.

Proposed scheme also controls the transfer rate according to the queue length of a task in edge gateway. Proposed scheme utilizes column-based compression skill and uses a real-time data compression library called lzo, which is not only easy to handle source code but also controls compression ratio and speed by changing compression levels.

We tested the efficiency of compression based on temperature data and found that it could be compressed in the 48% range by numerical repetition and token mapping characteristics. The column-wise compression modules are constructed by multiple chain connections with small compression segments rather than full block compression, in order to reduce both compression and decompression overhead as much as possible.

Table 1. Test of column-based data compression (room temperature, °C)

Control Options	Ratio(%)	Speed (MB/s)
Speed_Max	48.6	4.5753
Speed_Min	57.3	1.2967
Size-Min	63.0	0.231

Data: {25, 25, 24, 21, 21, 21, 26, 26, 27, 25, 26, 25, 25, 21, 23, 24, 23, 23, 23, 26, 26,25...}

It is also possible to reduce the amount of the transmitted data by using variable cycle transfer. For example, if a sensor node sends data in the normal range every 3 seconds and sends data in the abnormal range every 1 second, it can reduce the amount of sending data by one third. In fact, most of the data is normal, so this may reduce it further. That is, suppressing repetitive similar data can enhance performance much more.

4. Conclusion

In this paper, a new resource management scheme is devised to provide reliable data service for IoT information processing. In order to allow the IoT users to use the reliable data services, we used data-based gateway model which exploits both fog and edge gateway computing. Our scheme can transmit compressed sensor data by reducing overhead in advance through column-based data compression.

References

- [1] <https://www.ics.uci.edu/~dsm/ics280sensor/readings/data/02-771.pdf>, Feb 25, (2019)
- [2] P. Bonnet, J. Gehrke, and P. Seshadri, "Towards sensor database systems, the 2nd international conference on mobile data management," Hong Kong, January 8-10, (2001) DOI: 10.1007/3-540-44498-X
- [3] S. Yeon and J. Park, "IoT platform analysis and issues," ETRI Insight Report, vol.27, pp.1-56, (2016)
- [4] Wei Yu, Fan Liang, Xiaofei He, William Grant Hatcher, Chao Lu, Jie Lin, and Xinyu Yang, "A survey on the edge computing for the internet of things," IEEE Access, vol.6, pp.6900-6919, (2017) DOI: 10.1109/ACCESS.2017.2778504
- [5] Shanhe Yi, Zijiang Hao, Zhengrui Qin, and Qun Li, "Fog computing: platform and applications," The 3rd IEEE Workshop on Hot Topics in Web Systems and Technologies, Washington, DC, USA, November 12-13, (2015) DOI: 10.1109/HotWeb.2015.22
- [6] Gopika Premsankar, Mario Di Francesco, and Tarik Taleb, "Edge computing for the internet of things: a case study," IEEE Internet Of Things Journal, vol.5, no.2, pp.1275-1284, (2018) DOI: 10.1109/JIOT.2018.2805263
- [7] Amir M. Rahmani, Tuan Nguyen Gia, Behailu Negash, Arman Anzanpour, Iman Azimi, Mingzhe Jiang, and Pasi Liljeberg, "Exploiting smart e-Health gateways at the edge of healthcare Internet-of-Things: A fog computing approach," Future Generation Computer Systems, vol.78, no.2, pp.641-658, (2018) DOI:10.1016/j.future.2017.02.014
- [8] S. Byun and S. Jang, "Asymmetric index management scheme for high-capacity compressed databases," Journal of Korea Academia-Industrial, vol.17, no.7, pp.293-300, (2016) DOI: 10.5762/KAIS.2016.17.7.293
- [9] S. Ahn and K. Kim. "A join technique to improve the performance of star schema queries in column-oriented databases," Journal of Korean Institute of Information Scientist and Engineers, vol.40, no.3, pp.209-218, (2013)
- [10] Edson Tavares de Camargo, Elias P. Duarte Jr., and Fernando Pedonez, "A consensus-based fault-tolerant event logger for high performance applications," Euro-Par 2017: Parallel Processing, pp.415-427, (2017) DOI: 10.1007/978-3-319-64203-1_30
- [11] C. Cachin and M Vukolic, "Blockchain consensus protocols in the wild," Technical Report arXiv:1707.01873, IBM Research - Zurich, (2017)

This page is empty by intention.