The Application of Internet of Things and Oracle Database in the Research of Intelligent Data Management System

Yujiao Liu[1]*, Rajeev Kumar[2], Ashutosh Tripathi[3], Anil Sharma[4], Muskaan Rana[5]

[1]Chongqing Open University, Chongqing Business Vocational College, Chongqing, 400000, China
[2]Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab, India
[3]Department of ECE, University Institute of Engineering, Chandigarh University, Chandigarh, India
[4]Department of Computer Science, Faculty of Technology, Debre Tabor University, Ethiopia
[5]Department of Computer Science and Engineering Chandigarh University, Mohali, India
Emails: yujiaoliu7@126.com, Rajeev.kumar@chitkara.edu.in, ashu20034@gmail.com, anilsharma@dtu.edu.et, muskaan.e11410@cumail.in

Keywords: Sudden massive data, Internet of Things, Monitoring system, Oracle, Database

Received: February 16, 2022

The most critical issue in manufacturing is known as resource allocation. This article demonstrates an intelligent data management consisting resource allocation mechanism. The aim of the proposed system is to provide timely and effective decision for the resource allocation. Aiming at the needs of general large-scale monitoring systems, this paper designs an intelligent data management system that can provide fast data query and relieve sudden data congestion through in-depth research on Oracle database and data division. To the data access request from the front desk, the system can respond quickly through the real-time data monitoring module and the online analysis software OLAP mode database, which has far-reaching significance for the development of the Internet of Things and related systems. The experimental results show that, compared with the traditional system, the same bitmap index only occupies about 1/30 of the original table, and the data size is reduced by more than 10 times. The proposed model is compared with other state of art classifiers for evaluating percentage efficiency and F score. The experimental data verifies the characteristics of the system in this paper to strengthen the background data receiving and processing capabilities, and alleviate the problems such as the reduction of the system running rate and even the system paralysis caused by the sudden mass data.

Povzetek: Mehanizem dodeljevanja virov je implementiran s pomočjo inteligentnega sistema in baze Oracle.

1 Introduction

As a new generation of monitoring system development, IP-based network digital monitoring system has gradually become the main monitoring method in the contemporary era. At present, most of the monitoring is mainly used for indoor video monitoring with a small empty range. However, with the vigorous development of the Internet of Things technology, the information transmission technology with the object state as the basic data has broadened the development of the monitoring system [1]. More and more monitoring systems are gradually developed to rely on the Internet of Things technology to conduct unmanned monitoring in large outdoor spaces, such as intelligent bridge health detection, intelligent fire protection systems, environmental monitoring, etc. These systems play an important role in people's lives. A typical database - the modular design of Oracle database in intelligent data management system is shown in Figure 1 [2].

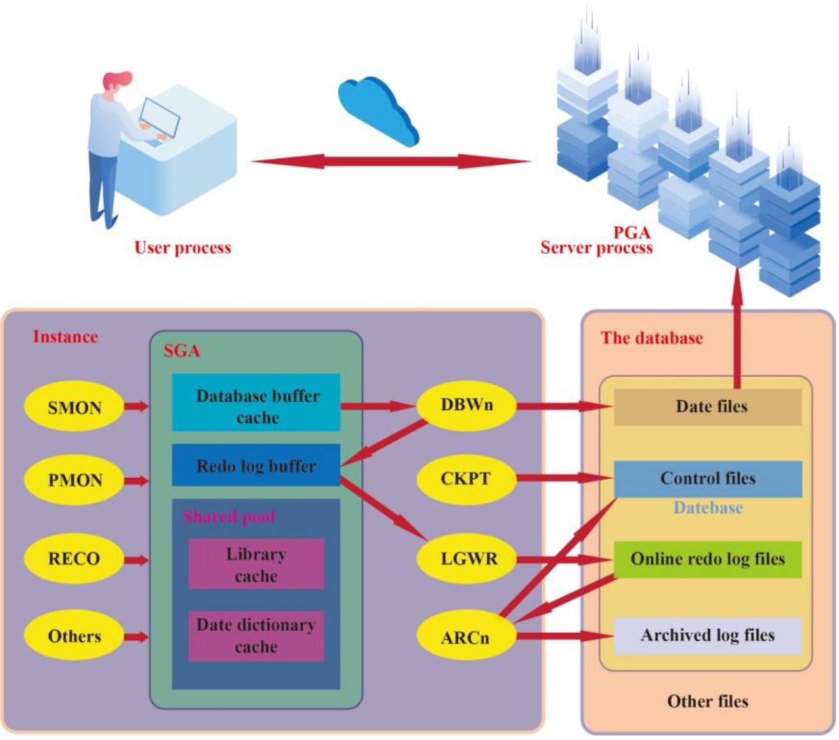

Figure 1: Modular design of Oracle database in intelligent data management system

In order to solve this problem in the monitoring system of the Internet of Things, this topic gives the detailed optimization design and the specific realization of the system from various aspects. The main research content of this topic is to develop the data layer of a new generation monitoring system based on the Internet of Things technology. In view of the impact of the sudden massive data generated by high-frequency collection on the background server, it can effectively solve the problem of excessive data loading and data volume. Too large and other problems [3]. Through in-depth research, there is still no universally applicable solution in the field of Internet of Things monitoring in China. The research on the undergraduate topic can effectively fill this gap, and the problem of reading and writing sudden massive data has been effectively discussed. It provides theoretical help and experimental data reference for relevant researchers in the future. The final output of this project is a data model that can solve the sudden massive data loading and reading and writing. This model can provide good support for the background data layer of the Internet of Things monitoring system and avoid the above problems caused by data [4, 5].

The rest of this article is organized as: Section 2 presents the most recent work carried out in the field of intelligent data management system. Section 3 consists the information about research methodology including system overflow and the implementation of business logic layer. The results and analysis part of the proposed scheme is covered in section 4. Section 5 describes the concluding remarks along with the future scope.

2 Related work

In this section the most recent work in intelligent data management system is discussed. One of the biggest characteristics of traditional monitoring systems is that there is less human-computer interaction, and the monitoring content is mostly image information. The amount of system data is usually maintained at a scale that increases linearly, and there is often a lack of analysis of the overall fluctuation trend of data over time [6]. The biggest feature of the new monitoring system based on the Internet of Things technology is to use the network to complete state-based monitoring, and to use the change trend of the monitoring object's own state as the standard for monitoring and analysis, so as to obtain comprehensive state information of the monitoring object. This system greatly increases the number of people According to the content of computer interaction, the monitor can change the monitoring mode according to the needs, which makes the originally stable growth of data volume more unpredictable [7]. Usually, due to the special needs of monitoring, high-density status collection of monitoring objects in a specific range will be performed, resulting in inevitable information peaks, which will bring greater data processing pressure to the background server processing. When the amount of data is overloaded, it may cause server congestion, slow message response, or even server crash. In order to avoid the problem of reading and writing caused by sudden massive data, it is necessary to provide a data layer structure suitable for the monitoring system mode, so as to perform data buffering and fast reading and writing of these data [8].

At present, there are few researches on the sudden mass database in the Internet of Things. Especially for massive data processing, basically most of the research papers are mainly analyzed in a database environment, such as the discussion of database partitioning technology, and the domestic research situation such as data table index design scheme as follows. Meng introduced a study from the National University of Defense Technology realized a real-time loading technology for TB-level massive data, and proposed a real-time loading system based on this technology. It mainly uses the SQL*Loader mechanism in the Oracle database to quickly process data storage, while using Database-specific swap partition method to quickly complete data loading [9]. Guo *et al.* from the School of Computer and Electronic Information of Guangxi University proposed a method to process massive data on a server, which avoids a series of huge initial hardware investment caused by the use of minicomputers with strong data storage capabilities at the hardware level question [10]. Zhen *et al.* from the Department of Ordnance Science and Technology of the Naval Aviation Engineering College proposed a realization method of multi-threading and double-buffer theory in the field of real-time data reception and storage [11]. Chen *et al.* gave a design and implementation of a massive burst signal acquisition system, and proposed an effective solution to the problems involved in high-speed acquisition frequency [12].

Foreign countries are much more in-depth than domestic research in massive data research. For example, the PI real-time database system developed by OSI software company in the United States is one of the most popular real-time databases today. It uses revolving door compression technology and secondary filtering technology to compress the massive data loaded into the database extremely efficiently, saving a lot of money. Hard disk space [13]. Research such as MARS [14] developed by Southern Methodist University, and System [15] in Princeton University's "Mass Storage Machine" project designed the "master version" of the database into the memory environment to make the system as a whole Architecture with greatly improved performance. Mitzutani *et al.* [16] presented a parallel processing structure using dual CPUs on the recovery architecture. Sidlauskiene [17] proposed a method based on the combination of log and shadow to solve the problem of occupying more memory space and needing to maintain a large number of page pointers.

Research on loading massive amounts of data in database clusters is still in its infancy. The American Supercomputer Application Center and the Department of Astronomy of Illinois State University jointly conducted research on the storage and query system for massive astronomical data [18]. The SDSS project in the United States has studied how to use SQL server clusters to quickly store data [19]. The evolution of artificial intelligence and Internet of Things is considered for several industrial applications and contributing towards social life [20-23].

In general, the current massive data processing technology is still a hot research topic, especially in today's booming Internet of Things, the stored data not only far exceeds the data generated by previous applications, but also has higher storage requirements. more stringent requirements. As a problem often faced in the development and application of current and future actual systems, the sudden mass data processing technology is the focus of research and solution.

3 Research methodology

This section includes the research design and methods about the system overflow. The implementation of business logic layer is also presented in this section.

3.1 System workflow

In order to meet the cross-platform characteristics and facilitate the general application in the Internet of Things, the background communication method adopts Web Service connection port, which enables unimpeded communication between Java EE architecture and .Net architecture [24]. In business, data is buffered by means of double buffering technology and file writing, and Memcached technology is also used to process, buffer and store data in memory, and then use multi-threading and batch processing to load data in the background [25]. Secondly, in terms of database, Oracle database is used to save data, which is mainly divided into two parts: real-time database and historical database. The sudden massive data is mainly stored in the real-time database, so this paper will elaborate on the design and implementation of the real-time database [26]. The background part mainly solves the problem of suddenness, mainly using buffering technology and caching technology to solve the problem of reading and writing, so the database is implemented by a single database. The main monitoring function process of the system is shown in Figure 2.

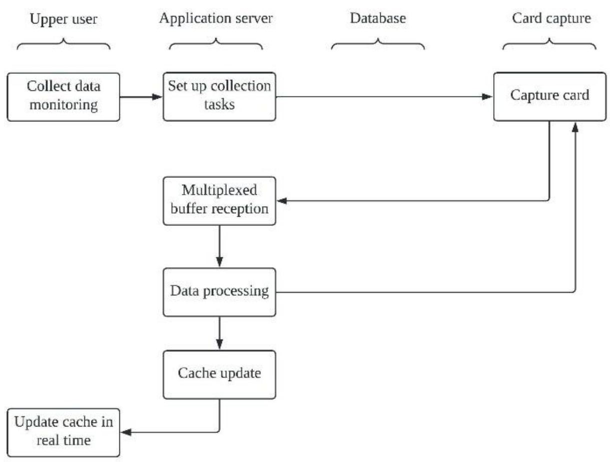

Figure 2: Flow chart of burst massive data collection function

The above process is the main execution steps when data is collected at high frequency. When the data is collected by other methods such as low frequency, the basic steps are the same as above. When performing data query, it mainly performs database access operations.

3.2 Implementation of business logic layer

The business logic layer mainly processes and responds to various requests of users on the server side, maintains the timed task queue, and continuously schedules the acquisition card for data collection according to the acquisition task. Using in-memory databases and caches to improve data processing efficiency [27].

Basic operations mainly include basic transaction operations such as adding, deleting, updating, and querying [28] performed by the user in the foreground. This module is responsible for receiving and parsing user requests from the foreground, and calling the corresponding module to access the database to obtain information, or modify the database content. After that, the obtained data is packaged according to the rules and returned to the front-end user for display.

The operation of the server on data interacts with the database through the ORM framework. The main monitoring objects are encapsulated by the decorator pattern in the design pattern idea. Taking the bridge object in bridge monitoring as an example, the class diagram is depicted in Figure 3.

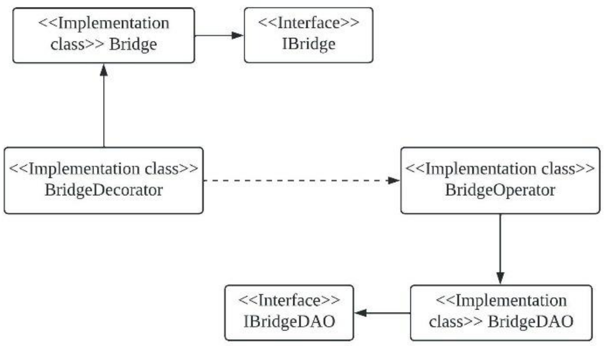

Figure 3: Basic module class diagram

IBridge completes the basic function definition of bridges, such as bridge number query, bridge parameter setting, bridge health status, etc. The Bridge class implements the IBridge interface, and rewrites each method as a concrete implementation. The BridgeDecorator class is used as the decoration class of the Bridge, and the Bridge object is called as a basic property. At the same time, the BridgeOperator inherits from BridgeDAO, which will extend the function of the database access class, and also exists as a base property for BridgeDecorator. In this way, new methods about bridges can be dynamically added to the BridgeDecorator. This interface-oriented design method meets the characteristics of JavaEE programming and is easy to maintain and upgrade.

The construction of other basic modules is similar to the above-mentioned modules. For example, the modules such as Line, Section and other modules complete the operation of business logic and database access by constructing their respective Decorator and Operator modules.

4 Results and Analysis

This section presents the description of result analysis of proposed scheme and the performance comparison of various indexes is also discussed in this section.

4.1 Data calculation transfer and batch loading method

The test data is mainly divided into 100,000 data loading and 1 million data loading. The data loading request is sent to the server through the simulation acquisition module to observe the processing efficiency of the system.

When performing a single insert operation of 100,000 pieces of data on the voltage acquisition table with calculation operation in the Oracle database, the average time for multiple sets of data is 2 minutes and 27 seconds; it is modified to a single insert operation of 1 million pieces of data. The data shows an average time of 23 minutes and 31 seconds. Instead, use 10 data as a group for batch data loading, and perform the insertion operation of 100,000 data in the voltage acquisition table with calculation operation. Multiple sets of data show that the average time is 2 minutes and 17 seconds; modified to 1 million data. For the insertion operation, multiple sets of data show that the average time is 23 minutes and 5 seconds. Using this batch method for insertion operations, the performance improvement is not obvious. The data is inserted directly into the database without calculation. In a single insert operation of 100,000 pieces of data, the average time of multiple sets of data is 19 seconds; if it is modified to a single insert operation of 1 million pieces of data, multiple sets of data show that the average time was 4 minutes and 44 seconds. Instead, 10 pieces of data are used as a group for batch data loading, and 100,000 pieces of data are inserted without calculation operations. Multiple sets of data show that the average time is 14 seconds; Multiple sets of data show that the average time is 3 minutes and 20 seconds.

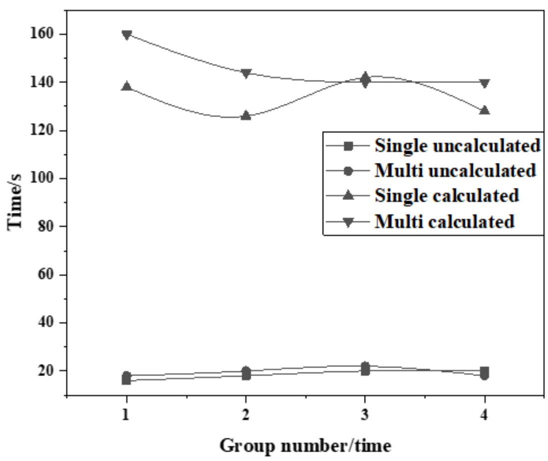

Figure 4: 100,000 data insertion tests

To sum up, it can be seen that when the server is used for batch data loading, as the amount of inserted data increases, the data insertion efficiency also increases gradually, but the efficiency does not increase linearly, and the efficiency increase is limited. If the calculation processing of the data is performed on the database, the consumption time is about 6 to 8 times compared with the simple insertion operation. 100,000 pieces of data are shown in Figure 4, and 1 million pieces of data are shown in Figure 5.

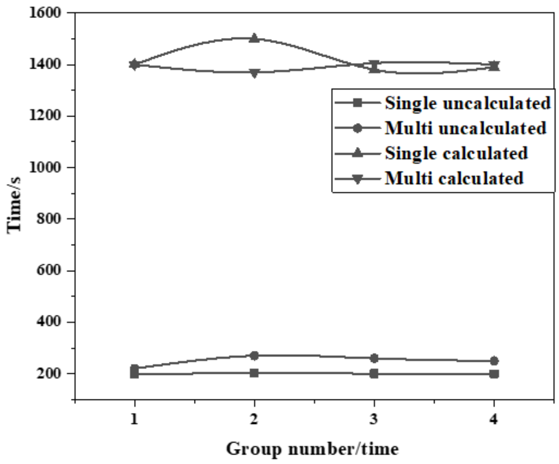

Figure 5: 1 million data insertion tests

When the amount of data is large, it is not advisable to temporarily buffer the data with memory, so the data needs to be cached in another way. Considering that Oracle database has a file bulk loading mechanism, a large amount of data can be buffered into files [29].

When the burst data is all stored in the form of files, the database tool SQL Loader is used to import the massive data into the database in parallel in the form of files. Taking 100,000 pieces of data as the test unit, it takes 0.23 seconds for the data to be buffered to the file, 2 seconds for batch importing into the database, and the comprehensive time is less than 2.2 seconds; with 1 million pieces of data as the test unit, the time for data buffering to the file is 6.8 seconds Second, the batch import database takes 110 seconds, and the comprehensive time is less than 117 seconds.

Although using SQL Loader to load data is not as efficient as direct bulk loading, this method does not have the risk of memory overflow and is more reliable. Through the analysis of the experimental data, it is known that when the data loading rate is higher than 100MB/S~150MB/S, the system is very likely to have the risk of memory overflow, so the method of loading the data into a file should be used for processing [30].

4.2 Data query

First, the B-tree query efficiency is compared. Data query experiments are carried out in four data tables. The data volumes in the data tables are 10,000, 100,000, 1 million, and 10 million, respectively. By building a B-tree index on it and querying it, the number of consistent reads per table is 3 data blocks, 3 data blocks, 4 data blocks, and 4 data blocks, respectively. It can be seen that even when the amount of data is very different, when conditional queries are performed on fields with unique constraints, the resources they consume, that is, the SQL execution efficiency, are almost indistinguishable. However, if the index is not built, the data consistency read will be greatly increased. Through experiments, it is found that there are 750, 8823, 102391, and 894721 data reads and writes respectively. This is extremely inefficient for massive data query. The index performance comparison chart is depicted in Figure 6.

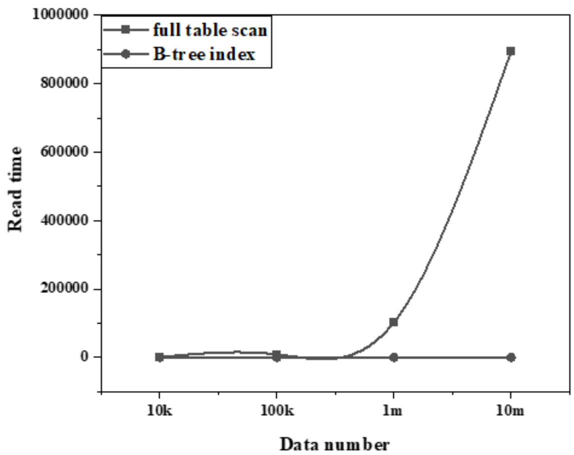

Figure 6: Performance comparison between B-tree index and full table scan

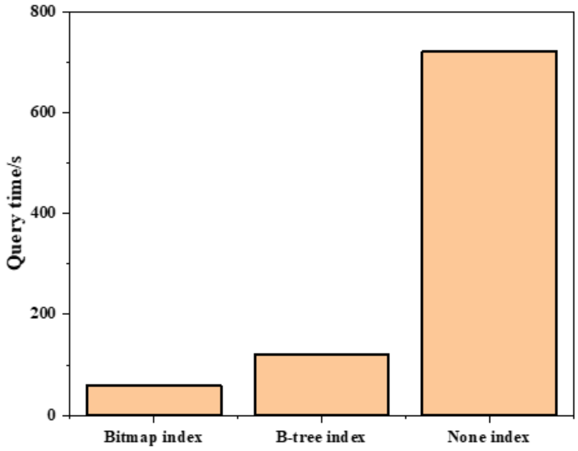

Figure 7: Performance comparison of various indexes

Secondly, for the parameter table, in some cases, the use of bitmap index can have a better improvement. When the data dispersion is very low, the use of B-tree index is often not a good choice [31]. As shown in Figure 8 below, for a parameter table with a data volume of 10,000, by constructing bitmap index and B-tree index to count and query parameters, the query efficiency under the bitmap index is significantly higher than the other two.

In addition to greatly improving the execution efficiency of specific queries, bitmap indexes can also greatly reduce the disk space occupied by the index. For a table with a data volume of 100,000, the space occupied by the B-tree index is basically more than half of the original table, while the same bitmap index only occupies about 1/30 of the original table, and the data size is larger than that of the B-tree index. More than 10 times smaller.

Finally, by adopting the partition strategy, the acceleration effect can also be mentioned for the query. For data tables, large-scale query operations are not suitable for indexes, and the database optimizer often uses partitions to query data directly. For example, in a large-scale data query in a data table with a data size of 50,000, due to the use of partition pruning, the efficiency of range partitioning will be higher than that of hash partitioning and other partitioning strategies. When performing a specific data query, that is, using "=" to determine, the data query efficiency of hash partitioning will be higher than that of range partitioning and other partitioning strategies. From a purely performance point of view, hash partitioning has high performance when the field repetition rate is low and the operation result set is small. For range partitioning and list partitioning, if the data is the same, the execution plan of the two is roughly the same, that is, there is no big difference in performance, but list partitioning can solve some specific data distribution problems, which is beneficial to the data according to certain way to manage [32].

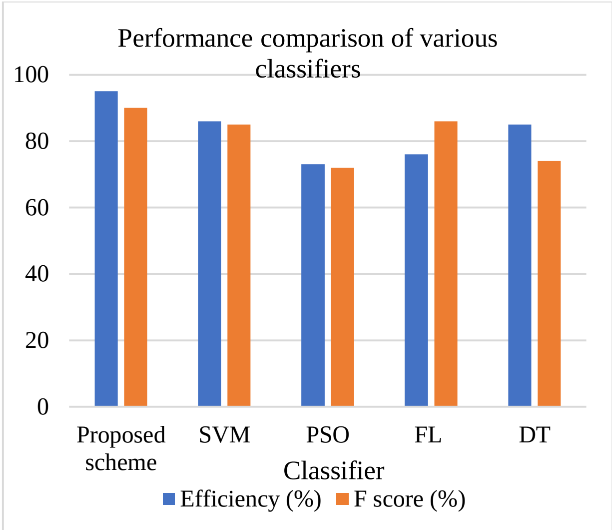

Figure 8: Efficiency and F score analysis of proposed scheme

The performance analysis of proposed scheme in terms of efficiency and F score is compared with existing state of art classifiers. This performance comparison is presented in Figure 8. The simulation results in terms of efficiency and F score of the proposed system and existing state of art systems are analyzed and the result is depicted in the above figure. The proposed scheme has the highest efficiency (95%) and F score (90%). The simulation analysis is done in less than 10 minutes and the overall performance is analyzed. It is observed from the analysis that the proposed scheme has superior performance in comparison with existing systems such as support vector machine (SVM), particle swarm optimization (PSO), fuzzy logic (FL), and decision tree (DT).

5 Conclusion

Through the research of the subject, the design and implementation of the monitoring system to deal with the sudden mass data have been basically completed, and the data test has been carried out for the part that has been realized, and good results have been obtained. The experiments have proved that by using double-buffering multi-threading and loading data, good processing efficiency can be obtained in a single-database environment. Although there are many problems in the design and implementation of sudden massive data, such as memory cache and multi-threaded data processing, etc., but in the end, they have been well solved by consulting the data, not only making them aware of the sudden massive data. With a deeper understanding of handling, the abilities are also exercised to solve scientific research problems. At present, there are still many aspects of the data layer that need to be studied in depth, such as distributed databases and other issues. From the experimentation, it is analyzed that the proposed model achieves better efficiency and F score when compared with existing state of art classifier such as SVM, PSO, FL and DT.

However, it is believed that with the continuous development of monitoring system technology and wider research on IoT-related applications on the basis of current popular cloud computing and cloud storage technology, there will be a lot of application space. The Internet of Things monitoring system with the help of new technology will have a better solution for the processing of sudden massive data. At that time, data will no longer be the bottleneck of system operation, and the Internet of Things will be widely used in society, bringing greater benefits to the society.

References

[1] Zheng, F., & Zheng, B. (2021, August). Research on the Optimization and Application of Intelligent Data Acquisition and Alarm System Based on Internet of Things. In *Journal of Physics: Conference Series* (Vol. 1992, No. 2, p. 022083). IOP Publishing. *1088/1742-6596/1992/2/022083*

[2] Hemanth, D. J., Shakya, S., & Baig, Z. (Eds.). (2019). *Intelligent Data Communication Technologies and Internet of Things: ICICI 2019* (Vol. 38). Springer Nature. *https://doi.org/10.1007/978-3-030-34080-3*

[3] Dan, J., Zheng, Y., & Hu, J. (2022). Research on sports training model based on intelligent data aggregation processing in internet of things. *Cluster Computing*, *25*(1), 727-734. *https://doi.org/10.1007/s10586-021-03469-z*

[4] Meher, M., & Rostamy, D. (2021). Hybrid of differential quadrature and sub-gradients methods for solving the system of Eikonal equations. Nonlinear Engineering, 10(1), 436-449. *https://doi.org/10.1515/nleng-2021-0035*

[5] Mi, Z., Wang, T., Sun, Z., & Kumar, R. (2021). Vibration signal diagnosis and analysis of rotating machine by utilizing cloud computing. Nonlinear Engineering, 10(1), 404-413. *https://doi.org/10.1515/nleng-2021-0032*

[6] Xue, B. (2021, January). Information Fusion and Intelligent Management of Industrial Internet of Things under the Background of Big Data. In *2021 13th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)* (pp. 68-71). IEEE. *10.1109/ICMTMA52658.2021.00025*

[7] Wang, Z., & Sharma, A. (2021). Research on transformer vibration monitoring and diagnosis based on Internet of things. *Journal of Intelligent Systems*, *30*(1), 677-688. *https://doi.org/10.1515/jisys-2020-0111*

[8] Sittrop, D., & Crosthwaite, C. (2021). Minimising Risk—The Application of Kotter's Change Management Model on Customer Relationship Management Systems: A Case Study. *Journal of Risk and Financial Management*, *14*(10), 496. *https://doi.org/10.3390/jrfm14100496*

[9] Meng, Q. (2019, August). A Study on the Urban Emergency Management System Based on the Internet of Things. In *International Conference on Management Science and Engineering Management* (pp. 645-655). Springer, Cham. *https://doi.org/10.1007/978-3-030-21248-3_47*

[10] Gao, L., Yang, Q., Zou, B., Liu, Q., & Wang, C. (2021, March). Research on Data Asset Management System of Graph Database Based on Internet of Things. In *Journal of Physics: Conference Series* (Vol. 1802, No. 3, p. 032134). IOP Publishing. *10.1088/1742-6596/1802/3/032134*

[11] Zhen, H., Kumar, P. M., & Samuel, R. D. J. (2021). Internet of Things Framework in Athletics Physical Teaching System and Health Monitoring. *International Journal on Artificial Intelligence Tools*, *30*(06n08), 2140016. *https://doi.org/10.1142/S0218213021400169*

[12] Chen, G., Xiao, X., Zhao, X., Tat, T., Bick, M., & Chen, J. (2021). Electronic textiles for wearable point-of-care systems. *Chemical Reviews*. *https://doi.org/10.1021/acs.chemrev.1c00502*

[13] Jiang, Z. G., & Shi, X. T. (2021). Application Research of Key Frames Extraction Technology Combined with Optimized Faster R-CNN Algorithm in Traffic Video Analysis. *Complexity*, *2021*. *https://doi.org/10.1155/2021/6620425*

[14] Wen, C., Yang, J., Gan, L., & Pan, Y. (2021). Big data driven Internet of Things for credit evaluation and early warning in finance. *Future Generation Computer Systems*, *124*, 295-307. *https://doi.org/10.1016/j.future.2021.06.003*

[15] Aneja, S., En, M. A. X., & Aneja, N. (2022, January). Collaborative adversary nodes learning on the logs of IoT devices in an IoT network. In *2022 14th International Conference on COMmunication Systems & NETworkS (COMSNETS)* (pp. 231-235). IEEE. *10.1109/COMSNETS53615.2022.9668602*

[16] Mitzutani, I., Ramanathan, G., & Mayer, S. (2021, November). Semantic data integration with DevOps to support engineering process of intelligent building automation systems. In *Proceedings of the 8th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation* (pp. 294-297). *https://doi.org/10.1145/3486611.3492413*

[17] Sidlauskiene, J. (2021). What Drives Consumers' Decisions to Use Intelligent Agent Technologies? A Systematic Review. *Journal of Internet Commerce*, 1-38. *https://doi.org/10.1080/15332861.2021.1961192*

[18] Sun, H., Tan, Y. A., Zhu, L., Zhang, Q., Li, Y., & Wu, S. (2022). A fine-grained and traceable multidomain secure data-sharing model for

intelligent terminals in edge-cloud collaboration scenarios. *International Journal of Intelligent Systems*, *37*(3), 2543-2566. *https://doi.org/10.1002/int.22784*

[19] Khalil, R. A., Saeed, N., Masood, M., Fard, Y. M., Alouini, M. S., & Al-Naffouri, T. Y. (2021). Deep learning in the industrial internet of things: Potentials, challenges, and emerging applications. *IEEE Internet of Things Journal*, *8*(14), 11016-11040. *10.1109/JIOT.2021.3051414*

[20] Singh, P. K., & Sharma, A. (2022). An intelligent WSN-UAV-based IoT framework for precision agriculture application. *Computers and Electrical Engineering*, *100*, 107912. *https://doi.org/10.1016/j.compeleceng.2022.107912*

[21] Zeng, H., Dhiman, G., Sharma, A., Sharma, A., & Tselykh, A. (2021). An IoT and Blockchain-based approach for the smart water management system in agriculture. *Expert Systems*, e12892. *https://doi.org/10.1111/exsy.12892*

[22] Sharma, A., & Singh, P. K. (2021). UAV-based framework for effective data analysis of forest fire detection using 5G networks: An effective approach towards smart cities solutions. *International Journal of Communication Systems*, e4826. *https://doi.org/10.1002/dac.4826*

[23] Sharma, A., Singh, P. K., & Kumar, Y. (2020). An integrated fire detection system using IoT and image processing technique for smart cities. *Sustainable Cities and Society*, *61*, 102332. *https://doi.org/10.1016/j.scs.2020.102332*

[24] Khan, I. K., Wei, Q., Chapman, S., Kc, D. B., & Kihara, D. (2015). The PFP and ESG protein function prediction methods in 2014: effect of database updates and ensemble approaches. *GigaScience*, *4*(1), s13742-015. *https://doi.org/10.1186/s13742-015-0083-4*

[25] Chen, C., Huang, T. S., Huang, J. C., Shih, C. H., & Du, Y. (2021). Music Intelligent Push Play and Data Analysis System Based on 5G Internet of Things. *Mathematical Problems in Engineering*, *2021*. *https://doi.org/10.1155/2021/6670534*

[26] Wang, X. (2022). Application of 3D-HEVC fast coding by Internet of Things data in intelligent decision. *The Journal of Supercomputing*, *78*(5), 7489-7508. *https://doi.org/10.1007/s11227-021-04137-0*

[27] Gao, X., Li, Q., & Liu, F. (2021, April). Research on the New normal Technology and Application of artificial Intelligence in the Internet of things. In *Journal of Physics: Conference Series* (Vol. 1865, No. 4, p. 042062). IOP Publishing. *10.1088/1742-6596/1865/4/042062*

[28] Qi, Y., & Wu, H. (2021, April). Fusion Application of Big Data and Cloud Computing In the Internet of Things. In *Journal of Physics: Conference Series* (Vol. 1881, No. 3, p. 032013). IOP Publishing. *10.1088/1742-6596/1881/3/032013*

[29] Zhou, G. (2021, March). The Application of Computer in Enterprise Economic Management Under the Background of Internet of Things. In *The International Conference on Cyber Security Intelligence and Analytics* (pp. 281-289). Springer, Cham. *https://doi.org/10.1007/978-3-030-70042-3_41*

[30] Caiqian, Z., & Xincheng, Z. (2021). Multimedia system and database simulation based on internet of things and cloud service platform. *Journal of Intelligent & Fuzzy Systems*, *40*(2), 2613-2624. *10.3233/JIFS-189253*

[31] Yue, S., Du, Y., & Zhang, X. (2021). Research and application of agricultural internet of things technology in intelligent agriculture. In *Journal of Physics: Conference Series* (Vol. 1769, No. 1, p. 012020). IOP Publishing. *10.1088/1742-6596/1769/1/012020*

[32] Zhou, B., Liu, Y., Xie, Y., Wang, J., Hao, Z., & Meng, J. (2021, May). Research and Application of Intelligent Street Lamp Platform Based on Ubiquitous Internet of Things. In *Journal of Physics: Conference Series* (Vol. 1920, No. 1, p. 012068). IOP Publishing. *10.1088/1742-6596/1920/1/012068*