Research and Implementation of Water Quality Monitoring System Based on Cloud Platform

LIU Qian [1, a], HOU Min [1, b], YANG Huayong[2, c] and ZHANG Guoan [1, *]

1 School of Electronics and Information, Nantong University, Nantong 226019, China
2 Zhongtian Technology Marine Systems CO., LTD, Nantong 226010, China
*a764348952@qq.com, b1299566710@qq.com, cyanghy@chinaztt.com, *zhang.guoan@ntu.edu.cn*

Keywords: cloud platform, water quality monitoring, data synchronization, visualization, privilege management.

Abstract. According to the problem of water pollution and deterioration of water quality, a water quality monitoring system was constructed based on cloud platform and the underwater observation networks of environmental protection enterprises. The cloud data synchronization and management were realized through Kettle. The management window was designed to realize cloud data visualization. A regular expression based access control method was designed to achieve privilege management. The problem caused by the large amount of data stuck was solved through the server-side paging and interval sampling, etc. The actual development and deployment shows that the platform works well, which improves the real-time quality and reliability of water quality monitoring effectively and provides an application case for the development of science and technology in the field of environmental protection including water source monitoring.

1. Introduction

Environmental protection is one of the most urgent problems in today's society which is directly related to the sustainable development of society and the health and well-being of people. In order to realize the energy conservation and emission reduction tasks in the "13th Five-Year Plan", China urgently needs to improve the monitoring system for energy conservation and emission reduction, and realize real-time monitoring of environmental data. To this end, the construction of the Internet of Things in the field of environmental protection is a very important task, and the water quality monitoring system is an important part of it [1, 2].

In the water quality monitoring system, various water quality monitoring devices send monitoring data to the water quality monitoring center in real time through wire or wireless communication. With the continuous emergence of new monitoring areas and sites, the amount of observation data will be more and more [3, 4]. Observation data from various parts of the country are continuously transmitted to the monitoring center 24 hours a day, which has high requirements for the storage and computing power of the monitoring center server. With its strong processing capacity, storage capacity and high

cost performance, cloud computing can solve the problems of high costs of operation and maintenance and limited service scale of enterprises.

This paper designed a water quality monitoring system based on the underwater observation network of environmental protection enterprises and cloud platform. The data obtained from underwater observation network were placed on the cloud platform, realizing the synchronization, display, management and analysis of cloud data.

2. Overall Design

Overall Architecture. The water quality monitoring system based on cloud platform is mainly composed of three parts: underwater observation network, cloud service platform and application layer. The overall system architecture is shown in Fig. 1. The underwater observation network consists of shore base control system, submarine optical electric composite cable, submarine junction box and high performance sensors, which are placed in vital water resources, rivers, lakes and sea bottoms [5]. The system is able to be continuous, real time, home position monitoring. Cloud service platform mainly provides data storage and computing power for the system, which is the data support for various application businesses. The application layer constructs a management window to realize the cloud data visualization based on the B/S architecture [6]. The main functional modules include real-time data visualization, historical data visualization and user privilege, etc., providing application interfaces for user access.
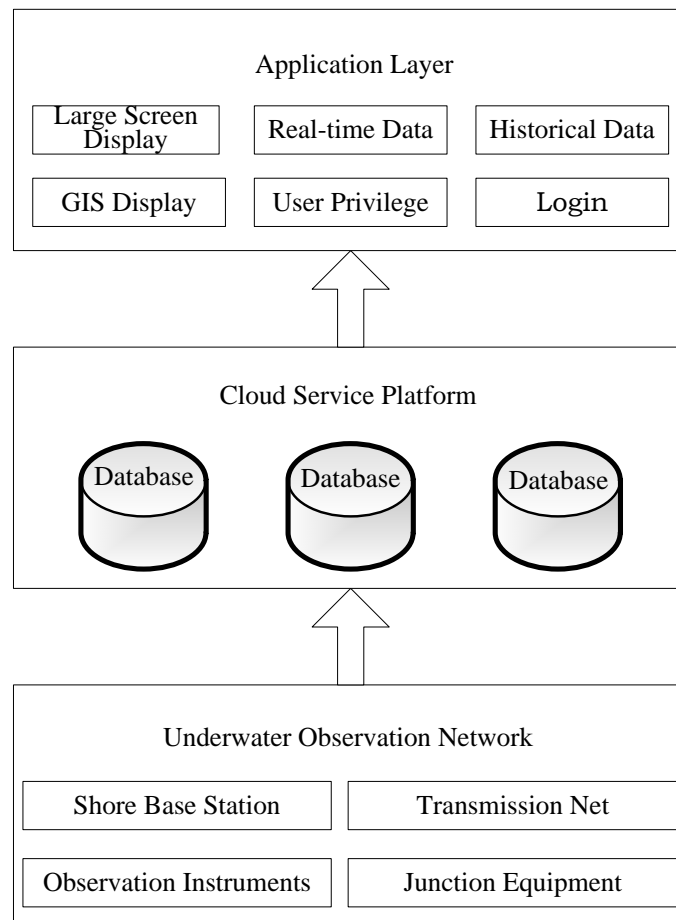

Fig. 1 Overall system architecture

Development Architecture. The front-end development technology of the system mainly adopts HTML (5) /CSS (3) /JavaScript and so on, and the back-end development technology mainly adopts JSP, Servlet and so on. Due to the continuous increase of new monitoring areas and sites, considering the scalability and coupling of the platform, the system architecture uses the Java EE based on Spring MVC and Mybatis. The architecture has a good hierarchical mode, which separates the code dependencies between layers. It is easy to develop with low cost, mainly divided into the following layers: domain object layer, data access object layer, service layer, controller layer and view layer.

3. Detailed Design

Database Design. The platform adopts MySQL database. The basic idea of designing the cloud transaction database is to divide the different databases by observation areas, and to divide the different tables by different sites and equipment nodes in the observation area.

According to the comprehensive needs of the platform, the observation area table area, the observation site table area_spot, the device node table spot_node and the plurality of sensor tables connected to the junction box are designed. In MySQL, the table area is used to store area information such as area_id, area_name and area_note. Each observation area includes several sites, and the observation site table area_spot is used to store area_spot , spot_id, spot_name and spot_note. Each observation site is further divided into a plurality of device nodes. The device node table spot_node consists of spot_node, node_id, node_name, node_type, and node_note, etc..

The type of device node is mainly divided into primary junction box and secondary junction box. The primary junction box is used to connect secondary junction box for large capacity electricity transmission. The secondary junction box is connected to underwater in situ observation instruments to monitor underwater environment parameters real time, dynamic and all day. A E-R diagram of multi-parameters sensor is shown in Fig. 2.
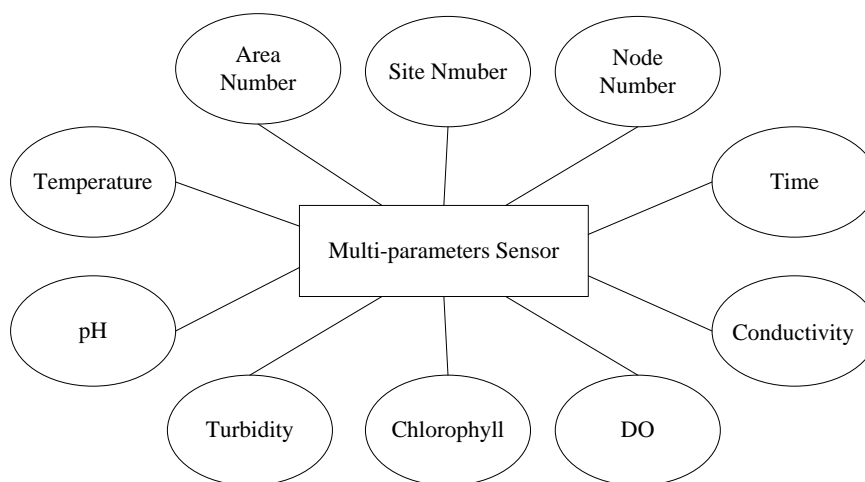

Figure 2 E-R Diagram of a Multi-parameters Sensor

Data Cloud Synchronization. The platform needs to integrate the monitoring parameters scattered in the local database into the cloud database, and uses the Kettle tool to synchronize and manage the cloud data. When configuring a database connection, some information need to be filled in including the connection name, connection type, access mode, host name, database name etc.. When the configuration is completed, the connection test must ensure that Kettle can access to avoid errors in synchronization. Kettle's Job has a timing function that can control multiple transformations at the same time, including data field modification, field addition, data de duplication of multiple nodes and so on.

Fig. 3 shows the transformation and execution of an observation site in Kettle. The observation site has three observation nodes, which are stored in a table of the local MySQL database. The entire transformation implements incremental synchronization by dragging, configuring, and joining input controls, switch/case controls, constant controls, output controls, and a small amount of code.
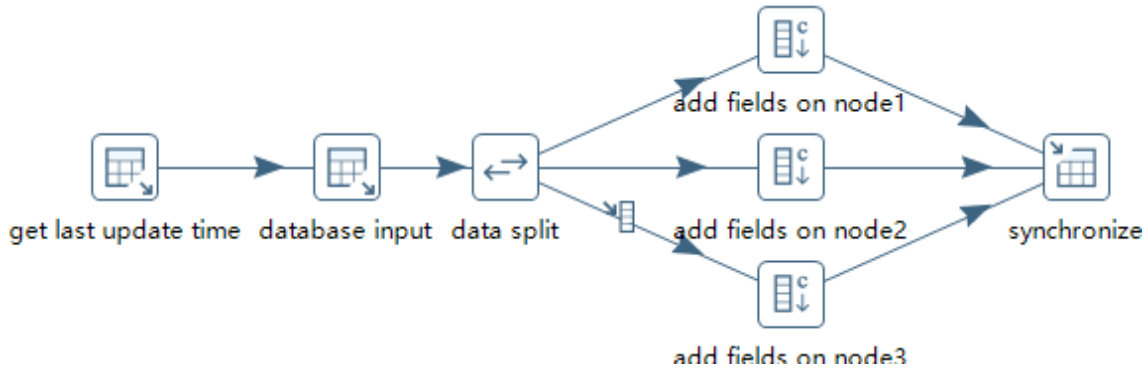

Fig. 3 Transformation and execution of an observation site in Kettle

Real-time Data Visualization. This paper uses a JSON data structure to communicate between the web client and the server. When web client requests data, the server returns the JSON data to the web client. The web client parses the data in JSON format into the JavaScript object through the function of $.parseJSON() in jQuery and renders the view.

To solve the problem of webpage jamming caused by a large amount of data, data collection functions for the dial data and the line chart data are designed respectively. The dial displays real-time data, requesting the server to return the data within the current 5 minutes and updating every 30s, while the line chart requests the server to return the data within the current 24 hours and updates every minute. At the same time, interval sampling is used to get the line chart data. When querying the database, the sampling time interval is designed to be 15 minutes. The query statement is set to *"where data_time>=#{start_time} and data_time <= #{end_time} and MINUTE (data_time) %#{t}= 0 "*, which can conditionally select data from the table to show the data trend of one day without returning all data. The real-time data visualization is shown in Fig. 4.
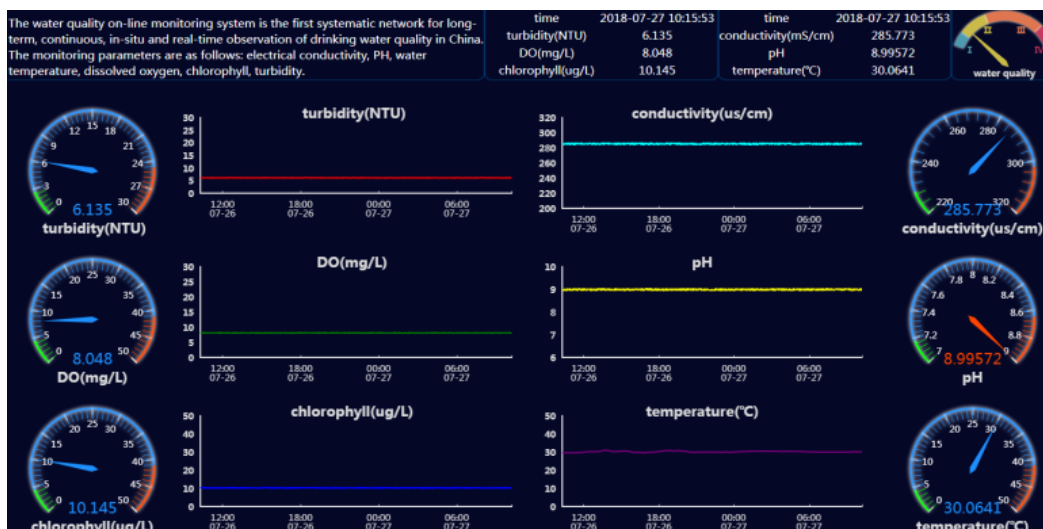

Fig. 4 Real-time data visualization

Historical Data Visualization. The paper adopts server-side paging. The mode is that the client requests data to display, and does not request data not to display. Firstly, it obtains the total number of

the query data records, sets the number of pages per page, and then connects the database to query the corresponding data when checking the other pages.

Different sampling methods are used to improve performance of searching for historical trend graphs. When query time interval is more than half a year, the time is grouped according to the date, and the data is obtained once every day. The time of data obtained is closest to the current time, and then the corresponding records are extracted from the database by this time. When query time interval is less than half a year, the data is obtained once every hour. The core code is as follows.

*@Select("select * from "+ NTYH_SENSOR_TABLE +" where data_time in ("+"select max (data_time) from "+ NTYH_SENSOR_TABLE +" group by DATE_FORMAT(data_time, '%m-%d -% Y')) having data_time>=#{start_time}and data_time<=#{end_time} ")*

List<NtyhSensor> selectEverydaySensorDataByTopOne(@Param("start_time") Timestamp start_time, @Param("end_time") Timestamp end_time);

*@Select("select * from "+ NTYH_SENSOR_TABLE +" where data_time in ("+"select max (data_time) from "+ NTYH_SENSOR_TABLE +" group by DATE_FORMAT(data_time, '%m-%d-%Y-%H')) having data_time>=#{start_time} and data_time<=#{end_time} ")*

List<NtyhSensor> selectEveryhourSensorDataByTopOne(@Param("start_time") Timestamp start_time, @Param("end_time") Timestamp end_time);

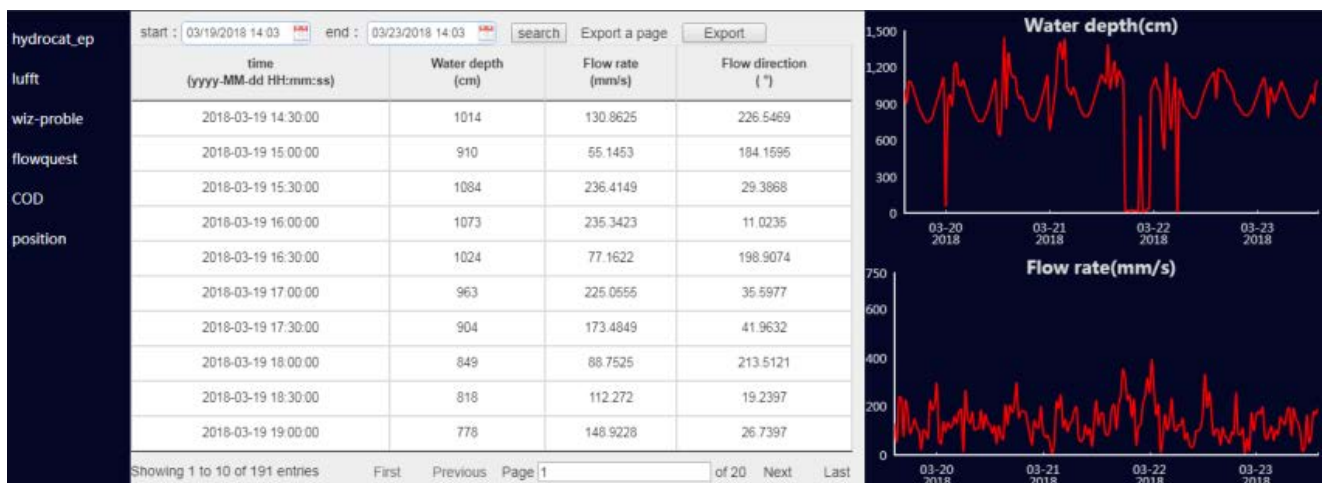Historical data visualization is shown in Fig. 5.

Fig.5 Historical data visualization

Privilege Management. Due to the continuous addition of new monitoring areas and sites to the platform, considering the independence of each region and site and the needs of user management, the system designs three types of user roles: super administrator, area administrator and regular user. Among them, the super administrator has the highest authority, mainly for the entire system management and maintenance, including the complete display, real-time query and historical query of all sites, data downloading and management of all users, etc.; the area administrator can only access and view all site information under the region; the regular user only has access to the corresponding sites.

The regular expression based access control method mainly completes the matching of paths by creating a property file *privilege.properties*, which stores the paths corresponding to the user privilege in the form of key-value pairs. The interceptor verificates relevant permissions according to the rule matching in the property file.

4. Conclusion

The water quality monitoring system based on cloud platform and the underwater observation network of environmental protection enterprise monitors water quality parameters long term, continuously, real time and in-situ. The system synchronizes the data acquired by the underwater observation network to the cloud platform in real time. Online monitoring, historical analysis and authority management of information have effectively improved the timeliness and reliability of water quality monitoring, providing technical support for data trend analysis in the next stage. At present, the platform has been put into use at several sites such as Yunhu in Nantong, which provide application practice for the development of environmental protection in China.

Acknowledgements

References

[1]M. H. Berlian, T. E. R. Sahputra, B. J. W. Ardi, L. W. Dzatmika, A. R. A. Besari, R. W. Sudibyo, and S. Sukaridhoto: Design and implementation of smart environment monitoring and analytics in real-time system framework based on internet of underwater things and big data. 2016 International Electronics Symposium (IES) (2016), pp. 403-408.

[2]Z. Wang and B. Xu: Design and implementation of the water environmental quality monitoring system based on big data. 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Guilin (2017), pp. 2092-2096.

[3]L. Bertholdo, C. G. D. Silva, G. D. A. Umbuzeiro and L. C. Júnior: Classification, Association and Clustering of Water Body Data: Application to Water Quality Monitoring. Environmental Processes(2017), 4(4), 813-831.

[4]S. H. S. Ariffin, M. A. Baharuddin, M. H. M. Fauzi, N. M. A. Latiff, S. K. S. Yusof and N. A. A. Latiff: Wireless water quality cloud monitoring system with self-healing algorithm. 2017 IEEE 13th Malaysia International Conference on Communications (MICC), Johor Bahru (2017), pp. 218-223.

[5]D. M. Toma, A. Mànuel-Làzaro, M. Nogueras and J. D. Rio: Study on Heat Dissipation and Cooling Optimization of the Junction Box of OBSEA Seafloor Observatory. IEEE/ASME Transactions on Mechatronics, vol. 20, no. 3 (2015), pp. 1301-1309.

[6]X. Qin, Y. Luo, N. Tang and G. Li: DeepEye: An automatic big data visualization framework. Big Data Mining and Analytics, vol. 1, no. 1 (2018), pp. 75-82.