Research on Data Analysis of The Vessel Shore Report

Taizhi Lv[1,2], Yongbing Chen[1], Peiyi Tang[2]

[1]*Jiangsu Maritime Institute, Nanjing,211170, China*
[2]*Nanjing Huihai Transportation Technology Company Limited, Nanjing,210000, China*

Abstract

With the increasing expansion of international trade, the shipping industry has developed rapidly. In order to regulate the ship sailing, China maritime regulatory authority has implemented a vessel shore reporting system. When a ship arrives or leaves the port, it should report to the maritime regulatory authority the package voyage, on-board personnel, passenger and cargo handling and other information. The ship arrival and departure reports are very high value. For improving intelligent level of maritime management, a data analysis system for vessel shore reports is design and implemented. The system collects the data from the report databases, transfers the data by Kafka, cleans the data by Flink, stores massive report data by Doris, queries the data by MyBatis plus and displays the chart by Echarts.

Keywords

Massive data analysis, vessel shore report, Doris, MyBatis plus, Spring Boot, Vue

1. Introduction

At present, most marine countries have developed the ship reporting system in combination with their national laws and actual conditions, among which the influential ones are the ship reporting systems of Australia, the United States and Japan [1]. In order to simplify the approval process for ships arriving and leaving the port, China has implemented the vessel shore report system since 2016 [2]. The implementation of the report system is of great significance to the reform and development of maritime affairs. The information in the report includes the dynamic information of ship voyage, the information of on-board personnel, the information of passenger and cargo loading and unloading. The report system of ship arrival and departure is an important way for the maritime administration to master the ship dynamics and cargo information, and the report data contains important value [3].

In order to fully mine the report data of ship arrival and departure, a report data analysis system is designed and implemented. This system uses different charts to visualize the port circulation cargo volume, ships and other information. These charts are helpful for the maritime administrator to master the ships and ports status. It can strengthen the ships supervision, and ensure the safety of water navigation. The system uses ETL (Extract-Transform-Load) tool to acquired the ship report data and AIS (automatic identification system) data. By the Kafka message queue, these data are transferred to the stream computing platform Flink. The Flink platform executes the data cleaning and fusing operations. The processed data is stored in the Doris and Redis database. The development mode of front-end and back-end separation is used in data visualization. The back-end uses the combination of Spring Boot + MyBatis plus to operate the Doris and Redis database. The front-end uses Vue framework to show the analysis result. Echarts is used in displaying the various charts.

ISCIPT2022@7th International Conference on Computer and Information Processing Technology, August 5-7, 2022, Shenyang, China
EMAIL: lvtaizhi@163.com (Taizhi Lv); 2473541293 @qq.com (Yongbing Chen); 1417451722 @qq.com (Peiyi Tang)

2. Background

2.1. Ship arrival and departure report

The vessel shore report is a system that ship users report the ship arrival and departure information to the maritime administration by mobile apps, web applications, etc. [3]. The ship shall report the information of arriving and leaving the port to the maritime administration at the place of expected departure or arrival 4 hours in advance, but the advance time shall not exceed 24 hours.

2.2. AIS

AIS is a digital navigation aid system composed of shore based and ship borne equipment, which realizes the navigation status and ship information interaction between ships and between ships and shore based [4]. The main functions of AIS include exchanging ship navigation information, guiding ship navigation, reducing ship collision avoidance, etc.

3. System architecture

As shown in Figure 1, this system is composed of data acquisition layer, data aggregation layer, data cleaning layer, data storage layer and data visualization layer. The adopted technologies include Flink, Kafka, Doris, Redis, Spring Boot, Vue, etc. The ETL (extract transform load) tool is used to regularly extract the arrival and departure reports and AIS data. The data is transmitted to the Flink cluster by the Kafka cluster. The Flink cluster cleans the ship arrival and departure reports to remove duplicate, abnormal and unmatched data with fusing AIS data. The processed data is stored in Doris and Redis through Kafka cluster, and the statistical results are displayed through Spring Boot and Vue.

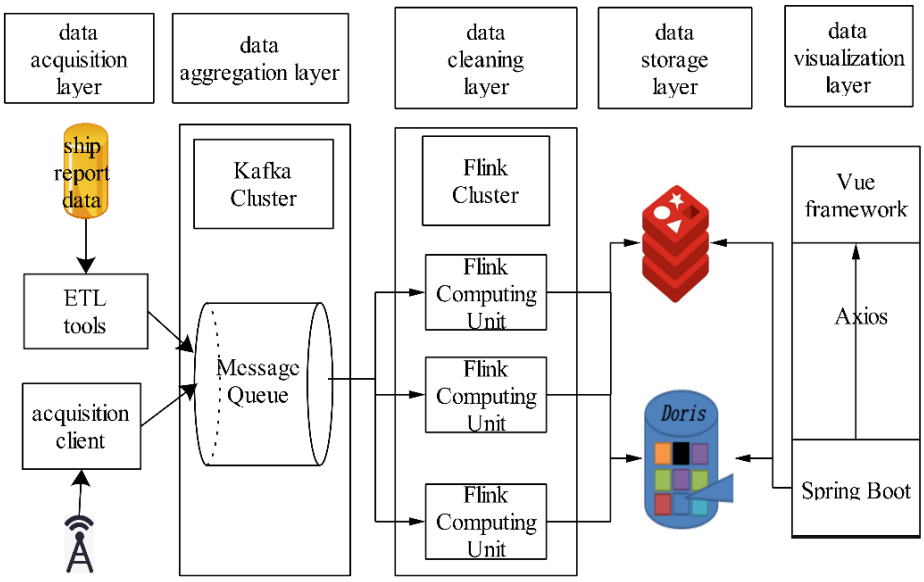

Figure 1: The system architecture diagram

3.1. data acquisition layer

The data acquisition layer acquires ship report and AIS data by data extraction technology. The real-time ETL tool extracts the ship report incremental from the ship arrival and departure report database.

3.2. data aggregation layer

This layer adopts three servers to realize a Kafka cluster. It can meet the high-performance, safe and reliable transmission of AIS and vessel shore report data [5]. The client of the data acquisition layer acts as the producer pushes the data to the Kafka cluster. Flink cluster acts as the consumer to process data in real-time.

3.3. data cleaning layer

Because the report data is entered manually, it cannot fully meet the requirements of analysis, and it is necessary to clean the ship report data. De-duplication is to delete the duplicate data and eliminate the redundant data. Based on time to live (TTL) feature of Redis, the same report data can be de-duplicated within 1 hour. Abnormal data cleaning is to identify and delete the wrong ship arrival and departure report data. The main errors are mismatching with AIS data, abnormal navigation data, abnormal cargo information, etc.

3.4. data storage layer

The traditional database is more suitable for the data transactional operation, while the system is mainly used to query massive ship arrival and departure report data. Doris is a MPP (Massively Parallel Processing) database, and it can provide millisecond query response performance for massive data [6-7]. In order to query the massive ship arrival and departure port report data, the Doris database is used in this system. This system takes 11 servers to build Doris database cluster, which is composed of two FE servers, 8 BE servers and 1 server for monitoring the cluster and dispatch tasks. Doris cluster structure is shown in Figure 2.

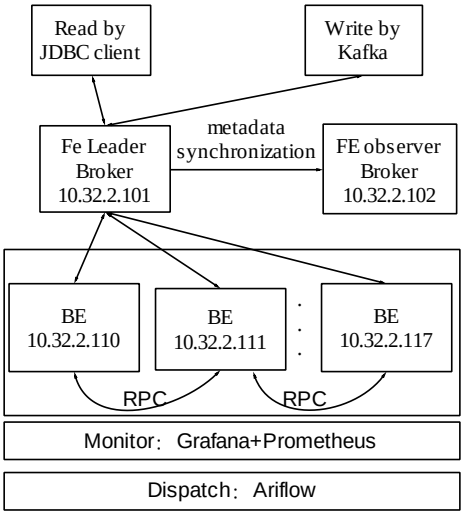

Figure 2: Doris cluster architecture in this system

3.5. data visualization layer

This data visualization layer adopts B/S architecture and the development mode of front-end and back-end separation. The data visualization layer is mainly composed of three parts: the large screen visualization for the report data statistics, the overview visualization about port statistics, and the report data query.

The large screen visualization module summarizes and displays the arrival and departure report information of the current year, including: statistics of the total cargo transportation volume and types of ships in each province, ranking of the port transportation activity in each province, proportion of cargo transportation types, the total number of domestic and foreign ships in each month, the total

transportation volume in each month, etc. The report data query module displays the ship arrival and departure report information of and the ship status at the port according to the name of the ship or MMSI (maritime mobile service identify) code entered by the user. The displayed information includes cargo type and container quantity. The main types of cargo carried by vessels are analyzed through the statistics of the proportion of cargo carried by vessels in previous years. The main service areas and other relevant information can be obtained by analyzing the main entry and exit cities of ships.

3.5.1. Back-end

As shown in Figure 3, the back-end is divided into four layers: model layer, map layer, service layer and controller layer.

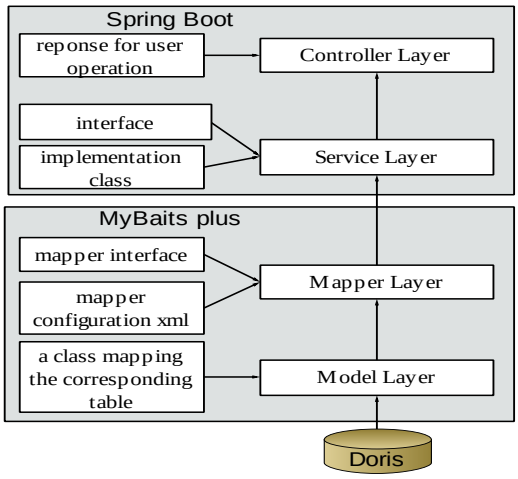

Figure 3: Back-end architecture

The model layer mainly implements the mapping between the Java class and the related table. This system mainly consists of the ship arrival and departure table, the ship cargo information table, the container information table, and the ship crew information table. These four tables cover the main data for the vessel shore report. For the report data is stored in Doris, and the ship info is store in Redis, this system should connect the different data source. The model layer uses an open-source dynamic pool tool for multi data source configuration. The mapper layer uses MyBatis plus framework to access data by encapsulating SQL statements. The service layer can directly call the mapper interface by binding mapper components by marking @autowritten [8]. It executes the business logic and transfers the data to the JSON format. The control layer is responsible for front-end and back-end interaction by accepting the front-end requests and returning the responding.

3.6. Front-end

The front-end uses Vue cli scaffold to quickly build Vue framework [9]. Element UI and Echarts library is used in the front-end, and the architecture is shown in Figure 4.

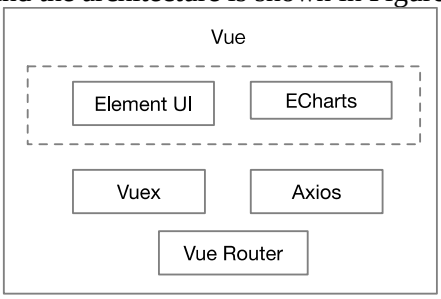

Figure 4: Front-end architecture

Echarts is a data visualization chart library of JavaScript, providing intuitive, vivid, interactive and personalized data visualization charts [10]. The data acquisition of the chart is based on the Axios asynchronous request, and then the data is bound to the corresponding ECharts chart object. ECharts draws the corresponding chart on the corresponding DOM container. The visualization page is basically displayed according to this process. Figure 5 shows the index page of the large screen visualization module. After Vue initialization, the page requests the data required in the chart from the back-end by Axios. After receiving the request, the back-end will call from the control layer, service layer, mapping layer and model layer to the database. It obtains the data according to the parameters passed by the front-end. The data will be returned to the front-end layer. The front-end will bind the data to the ECharts object to display the chart.

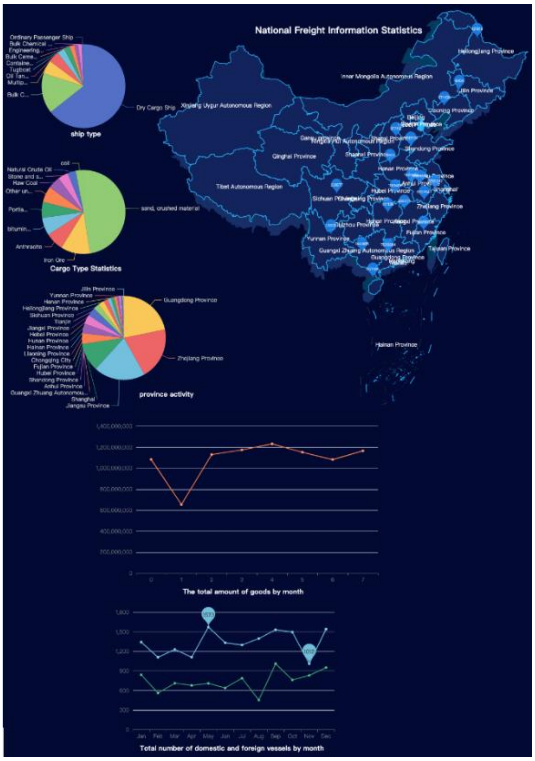

Figure 5: The index page of the large screen visualization

Figure 6 shows the ship arrival and departure report data. The user clicks the query button after inputting the searched ship name or MMSI. The front-end obtain the data information by the get request. It designs the wait animation and the status query button during the query process, and it improves the interactivity of the UI.

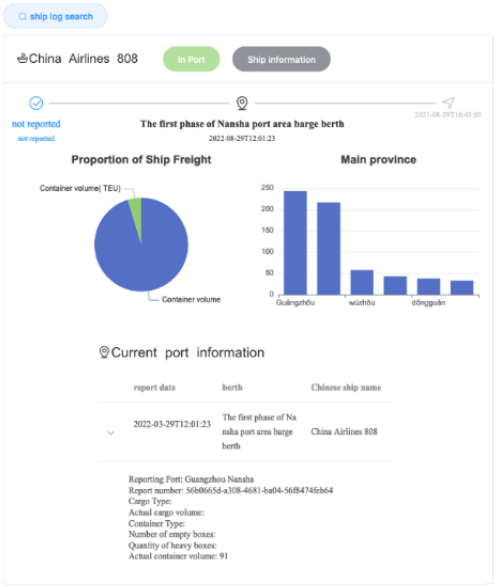

Figure 6: A ship arrival and departure report information

4. Conclusions

The report data of ship arrival and departure not only plays an important role in ship maritime supervision, but also can explore the status of regional macroeconomic development. In the context of building smart maritime, this research can improve the efficiency of ships arriving and leaving the port and reduce the waiting time for ships to load and unload goods in the port. By analyzing the ship arrival and departure data, the traffic of ships between provinces, the historical freight data, etc., the ship, port portrait is drawn. It not only can strengthen maritime management, but also has certain reference value for regional economic development.

Based on the mainstream big data technology, the system realizes the analysis and visualization of ship arrival and departure data. The system extracts data in real-time by the ELT tool, completes message transmission by Kafka, cleans data in real-time by Flink, stores the processed data in the MPP data warehouse Doris, and displays the analysis results by the front-end and back-end separation technology. The back-end reads data from Doris based on the Spring Boot+ MyBatis plus framework, converts the data into JSON format and returns it to the front-end. The front-end displays the results on the page through Vue + ECharts.

5. Acknowledgements

This work was financially supported by the excellent scientific and technological innovation team of Jiangsu colleges and universities (Maritime big data team), the young academic leaders for the QingLan Project of Jiangsu College and University, the Qianfan science and technology team of Jiangsu Maritime Institute (Big data analysis and application research team), and the shipping big data collaborative innovation center of Jiangsu Maritime Institute.

6. References

[1] Rind, Sobia, Soo-Jong Mo, and Yung-Ho Yu. "Study on development of vessel shore report management system for IMO MSP 8." Journal of Advanced Marine Engineering and Technology 40.5 (2016): 418-428.

[2] Cao, Aiguo. "On deepening the reform on the reporting system for ship's port entry and departure." China Maritime Safety, 2015.07 (2015): 26-28.

[3] Feng haidong. "Some thoughts on the precise ships supervision under the reporting system for ship arrival and departure." Journal of Transport Management Institute Ministry of Transport 27.03(2017):13-16+32.
[4] Hörteborn, Axel, et al. "A revisit of the definition of the ship domain based on AIS analysis." The Journal of Navigation 72.3 (2019): 777-794.
[5] Hugo, Åsmund, Brice Morin, and Karl Svantorp. "Bridging MQTT and Kafka to support C-ITS: A feasibility study." 2020 21st IEEE International Conference on Mobile Data Management (MDM). IEEE, 2020: 371-376.
[6] Su, Fei, et al. "A survey on big data analytics technologies." International Conference on 5G for Future Wireless Networks. Springer, Cham, 2017: 359-370.
[7] Bani, Fajar Ciputra Daeng, and Abba Suganda Girsang. "Implementation of database massively parallel processing system to build scalability on process data warehouse." Procedia Computer Science 135 (2018): 68-79.
[8] Li, Juan, and Jiajun Xu. "Outdoor Air Quality Real-time Monitoring System for Sports Athletes." 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). Vol. 5. IEEE, 2021: 719-722.
[9] bin Uzayr, Sufyan, Nicholas Cloud, and Tim Ambler. "Vue. js." JavaScript Frameworks for Modern Web Development. Apress, Berkeley, CA, 2019. 523-539.
[10] DeqingLi, HonghuiMei, et al. "Echarts: A declarative framework for rapid construction of web-based visualization." Visual Informatics 2.2 (2018): 136-146.