Design and analysis of intelligent retrieval system for drilling data and completion data based on cloud platform

Yifan Zhao¹, Kuisheng Wang¹ and Lianguo Chen²

¹ School of Computer, Xi’an Shiyou University, Xi’an, Shaanxi, 710065, China
² Changqing General Drilling Company, Xi’an, shaanxi, 710018, China

*Corresponding author’s e-mail: kshwang@xsyu.edu.cn

Abstract. In the context of the accelerating cloud platform construction in the oil industry, this paper solves the problem of difficulty in drilling and completion data retrieval by classifying and sorting drilling data and completion data to construct an intelligent retrieval model for drilling and completion data. Based on the cloud platform environment, this paper studies the method of constructing an retrieval system for drilling data and completion data through the microservice architecture. This system provides a unified retrieve page for structured data and unstructured data of drilling and completion operations. By using the ElasticSearch full-text search engine to build a retrieval model, it provides users with fast, efficient, and convenient comprehensive query and retrieval services, and lays a foundation for the intelligent construction of drilling and completion platforms.

1. Introduction

Against the background of the rapid development of the petroleum industry, concepts such as the construction of digital oilfield[1], smart oilfield[2], and oil cloud platforms[3] were proposed, then the construction of digitalization, informatization, and intelligence in the petroleum industry continued to accelerate. Since petroleum development operations need to coordinate multiple operations and integrate multiple technologies[4], the oil operation data has the characteristics of distribution, multi-source and massiveness, which has caused the island problem of oil data. Therefore, the problems of data integration and retrieval in the construction of digital platforms need to be solved. With the sustained development of network storage technology and distributed technology, cloud platform construction has been further improved. By integrating and scientifically managing oil development data using a cloud platform and making full use of cloud platform technology to integrate and retrieve oil development data, it is possible to solve the difficulty of integration and retrieval of oil data.

Drilling and completion are two important links in oil development. The quality of drilling and completion directly affects the production capacity and economic life of the oil well, and is related to the reasonable development of the entire oil field. However, due to the lack of a unified data integration and retrieval platform, the drilling and completion data cannot be used efficiently, quickly, and conveniently, which restricts the intelligent development of drilling and completion operations. Therefore, in the context of petroleum data integration, based on the drilling and completion data integration platform, the drilling data and completion data are integrated into the cloud platform. Through the use of microservice architecture and retrieval technology for intelligent query and retrieval of structured data and unstructured information in drilling and completion operations, it provides users with fast, efficient and convenient query and retrieval services, which provides support.
for intelligent decision-making, data analysis and prediction of drilling and completion operations, then lays the foundation for the intelligent construction of drilling and completion platforms.

2. System architecture under the cloud platform

This system is deployed on a cloud platform. Since microservice technology can effectively solve the integration problem of various heterogeneous systems, it can flexibly use virtual containers to isolate a single application. And by using Docker or other technologies, it makes system automation integration, deployment, operation and maintenance work easier[5], so this system adopts a microservice architecture model. SpringCloud is a set of general tools built on SpringBoot. As a microservice development framework, it is very suitable for quickly building distributed systems[6]. This system uses the Spring Cloud microservices solution. Figure 1 shows the system architecture.

![System architecture diagram](image)

Service Discovery: This system uses Eureka components as the registration center to realize the service management function in the microservice architecture. The Eureka registration center performs service management on already started services, dynamically maintains the service registry through a heartbeat mechanism, and checks the node health information of the services.

Client Side Load Balancer and Circuit Breaker: This system uses Ribbon as the load balancing component. And this system uses Hystrix as the circuit breaker protection middleware of the distributed system to prevent the service avalanche effect caused by a single service failure.

API Gateway: As the middleware between the client and server services, the gateway is the only external entrance of the system. This system uses Zuul as a gateway filter to realize the routing, and authentication of the gateway.

Center of Configuration: This system uses the SpringCloud Config service for centralized configuration management. By combining Git and RabbitMQ, the service dynamically pulls the configuration file, which facilitates maintenance and release.

Microservices division: This system is divided into three services, named user service, data import and analysis service, and data retrieval service. User service mainly implements functions such as user login, user role authorization, and user data authorization. Data import and analysis service analyzes and stores the data in the data platform, and establishes an index to retrieve the data through the index. Data retrieval service retrieves structured and unstructured data from drilling and completion within the scope of user data authorization, and performs analysis and statistics.
3. System analysis and design

3.1. Data classification and collation
This paper classifies and sorts the drilling and completion data, and divides it into 10 types, namely basic information, oil and gas layer, wellbore, strata, drilling and completion operations, cementation and wellhead, perforation and stimulation, completion string, and formation testing. In the basic information category, according to business needs, the retrieval objects are divided into 4 categories, namely oil and gas fields, oil wells, wellbores and well pipes. By classifying the drilling and completion data, the object of comprehensive retrieval of drilling and completion data is determined.

3.2. System function design
According to the business needs of the drilling and completion data platform and the characteristics of the drilling and completion data, this paper has designed the specific functions of the system in detail. As shown in Figure 2, the basic functions of the system.

User service: User services mainly implement user management and data authorization and verification. By creating user roles, creating organizational management, and authorizing roles and assigning departments to users, the function of different users can be divided. Due to the strict confidentiality of oil drilling and completion data, when creating a user, the user needs to authorize the oil well-level data. When the user is authorized to drill and complete the well data of the well, the user can query the well information.

Data retrieval service: The system provides functions of comprehensive data retrieval, business data query and report query. The comprehensive data retrieval function is divided into data retrieval and document retrieval. Data retrieval uses keyword matching and fuzzy queries to retrieve all object data that meets the requirements and returns them to users according to different objects. Document retrieval is a full-text search of document and picture, by using keywords, returning documents and pictures to users. By using a common query method, oil well business data query can be quickly queried for a single well. Based on querying the completion data, the system count and analyse the data to generate reports.

Data import and analysis service: Data import is to import structured drilling and completion data, and establish an index for each oil well, and query through the index to improve the query speed. For the dynamic data of drilling and completion, when data is imported, it is necessary to analyze the unstructured data, and then establish the index to achieve fast query and analysis.

4. System implementation
Elasticsearch is a distributed, highly scalable, and real-time search and data analysis engine based on Lucene[7][8]. As a search engine in information retrieval system, Elasticsearch is highly scalable due to its distributed architecture design[9]. According to the specific needs of the business, this article
builds 3 types of indexes through the Elasticsearch search engine. The first is the static data index for drilling and completion. In this paper, objects such as oil and gas fields, oil wells, wellbores, and well pipes are indexed separately. Enter the query string on the comprehensive data retrieval page, use fuzzy queries or keyword matching methods to traverse the Elasticsearch index object, and obtain the retrieval results under different objects. The second type is the dynamic data index for drilling and completion, which uses a single oil well as an object to establish an index, and performs data retrieval and multi-dimensional analysis. The third type uses Elasticsearch to create an index structure for text and pictures. This system uses full-text search technology to retrieve document information.

4.1. Construction of json document index based on oil well entity

In the relational database, data query often requires multiple tables to be related, as shown in Figure 3. Due to the diverse sources of data for drilling and completion, the data tables are stored in multiple databases, which reduces the efficiency of data retrieval. Taking oil well objects as an example, by indexing oil well entities, the related queries of multiple tables are converted into index queries of the JSON document structure to speed up the retrieval efficiency, and the multi-dimensional analysis of oil well entities can be performed through the query conditions.

Using the oil well ID (IDWELL) as the primary key of the well entity, each record in the relational database table can generate a JSON document, which is defined as $d^n_i$, where $i$ represents the $i$-th table related to oil well information and $n$ represents the $n$-th record in the table. As shown in Figure 4, a row of records in the Wst_Wellheader table is converted to JSON format.

![Figure 3. Relational database table structure.](image)

![Figure 4. JSON format conversion.](image)

With $k$ representing different values of oil well ID, the oil well record in each table can be expressed as:

$$D_1 = \{\{k_1: d_{n_1}^1\}, \{k_2: d_{n_1}^1\}, \{k_3: d_{n_1}^1\}, \ldots, \{k_m: d_{n_1}^1\}\}$$

(1)

$$D_i = \{\{k_1: d_{n_i}^i\}, \{k_2: d_{n_i}^i\}, \{k_3: d_{n_i}^i\}, \ldots, \{k_m: d_{n_i}^i\}\}$$

(2)

Definition $S^i_k$, which represents all $d_{n_i}^i$ records of the $k$-th well in the $i$-th table which related to the well information. Then there are multiple well records with the same ID in the table, which can be converted into the JSON document shown in Figure 5.

For tables related to wells but without well IDs, such as the association between Wst_Job and Wst_Job_Repter tables, define $q$ as the associated primary key, then:

$$D_1 = \{S^1_k: \{q: d_{n_1}^1\}, S^2_k: \{q: d_{n_1}^1\}, S^3_k: \{q: d_{n_1}^1\}, \ldots, S^m_k: \{q: d_{n_1}^1\}\}$$

(3)

$$D_i = \{S^1_k: \{q: d_{n_i}^i\}, S^2_k: \{q: d_{n_i}^i\}, S^3_k: \{q: d_{n_i}^i\}, \ldots, S^m_k: \{q: d_{n_i}^i\}\}$$

(4)

The converted JSON format is shown in Figure 6.

![Figure 5. Same ID JSON conversion.](image)

![Figure 6. Father-son table JSON conversion.](image)
Definition \( W^i_k \), which represents all \( d^k_i \) records of the \( k \)-th well in the \( i \)-th table which related to the well information and its sub-table information. According to the above definition, convert \( D_1, D_2, ..., D_l \) to the following format:

\[
D_1 = \{ W^1_1, W^1_2, W^1_3, ..., W^1_k \} \tag{5}
\]

\[
D_i = \{ W^i_1, W^i_2, W^i_3, ..., W^i_k \} \tag{6}
\]

According to the formula

\[
D_1 \cup D_2 \cup ... \cup D_l = \{ W^1_1, W^2_1, W^3_1, ..., W^k_1 \} \cup \{ W^1_2, W^2_2, W^3_2, ..., W^k_2 \} \cup ... \cup \{ W^1_k, W^2_k, W^3_k, ..., W^k_k \} \tag{7}
\]

\[
D_1 \cup D_2 \cup ... \cup D_l = \{ W^1_1, W^2_1, W^3_1, ..., W^l_1 \} \cup \{ W^1_2, W^2_2, W^3_2, ..., W^l_2 \} \cup ... \cup \{ W^1_k, W^2_k, W^3_k, ..., W^k_k \} \tag{8}
\]
can be derived.

Define \( T_1, T_2, ..., T_k \) to represent sets \( \{ W^1_1, W^2_1, W^3_1, ..., W^k_1 \}, \{ W^1_2, W^2_2, W^3_2, ..., W^k_2 \}, \ldots, \{ W^1_k, W^2_k, W^3_k, ..., W^k_k \} \) respectively, which represents the Json index tree of wells 1 to \( k \).

From this, formula \( T \rightarrow \{ T_1, T_2, ..., T_k \} \) is derived. In summary, the Json index based on the entity of oil wells is shown in Figure 7.

In the above way, the static data and dynamic data of drilling and completion are indexed separately, and the table association query in the traditional database is transformed into the traversal of the Json document of the oil well entity. Constructs a Nested type index through Elasticsearch, by enter the search string to retrieve the data, and finally return the data object of the oil well entity. And through multi-keyword query, this system performs statistics and analysis on drilling and completion data to generate reports.

### 4.2 Construction indexes of unstructured data

For unstructured data such as texts and pictures in the drilling and completion data platform, index them through unified document metadata. The metadata structure is shown in Figure 8.

![Figure 7. Json index tree.](image1)

![Figure 8. File metadata.](image2)

Through the construction of the metadata index, Elasticsearch uses the tokenizer to segment words and establish an inverted index. Because the text information of drilling and completion contains a lot of professional terminology in the oil industry, the vocabulary of drilling and completion is established based on the use of IK word segmentation. By using a custom word segmentation, Elasticsearch performs more accurate word frequency statistics on text information. After the user searches for keywords, Elasticsearch searches the document through the reverse index, and returns the document object to the user according to the score. After that, the document data is obtained from the database through the object information in the document.
4.3. Data retrieval
This system provides users with a unified search interface, as shown in Figure 9. By inputting search fields, using keyword matching and fuzzy matching methods, the drilling and completion data is searched to obtain entity objects and text information that match the fields. By setting the query field and using the index constructed by the Json document tree, the system performs multi-dimensional statistics on the drilling and completion data, and finally generates reports, as shown in Figure 10.

![Figure 9. Comprehensive retrieval interface.](image1)

![Figure 10. Report interface.](image2)

5. Conclusion
This paper designs and analyzes the intelligent retrieval system for drilling and completion data under the cloud platform, describes the basic architecture of the system based on the cloud platform, introduces the basic functions of the system, and briefly introduces the implementation method of data retrieval in the system, then structures a retrieval model based on the drilling and completion data platform. The drilling and completion data retrieval system introduced in this paper provides users with a unified retrieval interface to retrieve structured oil well data and unstructured text, pictures and other data in the drilling and completion platform. By indexing the oil well entities, the retrieval speed of drilling and completion data is improved, and through multidimensional analysis and statistics of the oil well data, the system generates reports, which lays the foundation for the intelligent construction of drilling and completion platforms.

References
[1] LI, Hong, et al. Applications of Artificial Intelligence in Oil and Gas Development. Archives of Computational Methods in Engineering, 2020, 1-13.
[2] Li Jianfeng. Construction of Intelligent Petrochemical Industry: from Information to Intelligence. Oil Forum, 2020, 39(01):34-42.
[3] Du Jinhui, Shi Fugeng, Zhang Zhonghong, Wang Tiecheng, Ding Jianyu. Research and practice of Dream Cloud for petroleum exploration and development of PetroChina. China Petroleum Exploration, 2020, 25(1): 58-66.
[4] Song MengXin, Feng Mei. Construction and Application Practice of Petroleum Professional Software Cloud Platform. China CIO News, 2019(12):25-27.
[5] Li Bingyuan, Zhang Wenzhe, Zhao Chunxue, Chen Qiqi, Si Changzheng. Application of microservices in multi-platform integration of petrochemical enterprises. China Management Informationization, 2020, 23(09):80-81.
[6] Liu Bin. The Path of Micro-Service Transformation of the Guarantee System of Telecommunication Integrated Service Based on SpringCloud. Management & Technology of SME, 2019(11):90-91.
[7] Xu Weijie, Wang Ting, Xue Wan Ting. Design and Implementation of Search Engine Based on ElasticSearch. Think Tank Era, 2019(23):228+240.
[8] Coronel, Josue Balandrano, and Stephen Mock. Designsafe: Using elasticsearch to share and search data on a science web portal. Proceedings of the Practice and Experience in Advanced Research Computing 2017 on Sustainability, Success and Impact. 2017. 1-3.
[9] Praveen M Dhulavagol, Vijayakumar H Bhajantri, S G Totad. Performance Analysis of Distributed Processing System using Shard Selection Techniques on Elasticsearch. Procedia Computer Science, 2020, 167.