Express Supervision System Based on NodeJS and MongoDB

Li Liang
School of Computer Science
Communication University of China
Beijing Key Laboratory of China in
Security&Protection Industry
Beijing, China
18810361527@163.com

Ligu Zhu, Wenqian Shang,Dongyu Feng, Zida Xiao
School of Computer Science
Communication University of China
Beijing Key Laboratory of China in
Security&Protection Industry
Beijing, China

Abstract—**Aiming at the functional requirements of the Express Supervision System, This paper discusses the advantages of using AngularJS to build the front-end framework, the advantages of using NodeJS to construct the back-end Web server, and the performance advantages of storing data based on MongoDB. This paper focuses on the storage solutions of using MongoDB to store large data and the statistical analysis solutions based on MapReduce. This paper argues on how to build Web services that meet the requirements of large data visualization based on NodeJs.**

Keywords—Data storage; Statistic analysis; NodeJS; MongoDB;

MapReduce

I. Introduction (*Heading 1*)

With the advent of the Internet+ era, online shopping has become an indispensable way of shopping in daily life. According to China online shopping market in 2015, China online shopping market in 2015 has reached 3.8 trillion yuan. As of 2016, the number of online shopping in China has reached 350 million. Hundreds of millions of online shoppers have generated a huge number of courier pieces. In 2015,the number of delivery has reached 20.5 billion. The average number of bills generated per day is 60 million. It is foreseeable that in the future, these figures will grow exponentially.

The data produced in Express is a very valuable digital resources, which includes an important indicator measuring the level of social economy in China. The data information from the Express have an important influence on the analysis of industry trends and the safety supervision. Therefore, the study of the data from Express has an important role in guiding the distribution of resources and standardizing the rules in Express. At present, there are a large number of off-line data in the security system accounting for about 400TB memory, which has reached 0.2% of the global production of printed materials(200PB). The data mainly includes the bill information, the state information and the branch information, which is the important sensitive data of the Express industry. Therefore, establishing a monitoring platform achieving the management of the off-line data, has an important role in the prediction of the industry trend, the optimization of industry organization and safety supervision.

II. Key technologies

A. AngularJS

AngularJS is an open source JavaScript library maintained by Google to help a single page application run. Its goal is to enhance browser-based applications with MVC mode (MVC) capabilities, making development and testing easier.

B. MongoDB

MongoDB is a powerful, flexible, high performance, easy expanded way of data storage. It's a document-oriented database, not a relational database, is NoSQL.[1] The so-called document-oriented, use a more flexible "document" replace the "line" concept in original relational database. The document can be the value of array, documents and other complex data model. And the key of document are not pre-defined and will not be fixed. One of the main ideas of designing the MongoDB is that the operations that can be handed to the client are transferred from the server to the client, such as the generation of object id and other operational solutions. MongoDB as a general-purpose database, in addition to the ability to create, read, update, delete data, but also provides a series of unique features continue to expand. [2]

- MongoDB supports generic secondary indexes, allowing multiple quick queries, providing unique indexes, composite indexes, geospatial indexes, and full-text indexes.
- MongoDB supports "aggregated pipelines". Users can create complex aggregations from fragments, and automatically optimize them through the database.
- MongoDB supports a time limited set that applies to data that will expire at some point, such as a session. Similarly, MongoDB also supports fixed-size collections for storing recent data, such as logs.

978-1-5090-5507-4/17/$31.00 ©2017 IEEE
ICIS 2017, May 24-26, 2017, Wuhan, China

- MongoDB supports a very easy-to-use protocol for storing file and file metadata.

MongoDB does not have some common features in relational databases, such as connecting queries and generating lines in order to get better scalability. Because these two functions in de distributed system is difficult to use. One of MongoDB's primary goals is to provide superior performance, MongoDB can dynamically fill the document, but also pre-allocation of data files to take advantage of additional space in exchange for stable performance. MongoDB uses as much memory as the cache, trying to automatically select the correct index for each query. In short, MongoDB in all aspects of the design is to ensure his stable performance. Although, MongoDB is very powerful and tries to preserve many of the features of a relational database, it does not pursue all the features of a relational database. Whenever possible, the database server will handle the generation and logic to the client to achieve. This compact design is one of the reasons MongoDB can achieve such high performance.

C. NodeJS

Node.js written in C ++ language, is a JavaScript operating environment. Node.js is a JavaScript runtime environment. Node.js uses the Google Chrome V8 engine for good performance, and also provides a lot of system-level APIs, such as file operations, web programming, and so on. The JavaScript code on the browser side is subject to various security restrictions at run time, and the operation of the client system is limited. Node.js uses event-driven, asynchronous programming, and designed for network services. Node.js design ideas take the event-driven as the core, it provides the vast majority of APIs that are event-based, asynchronous style. Take the Net module as an example, where the net Socket object has the following events: connect, data, end, timeout, drain, error, close, etc. The developer using Node.js needs to register the corresponding callback function according to its business logic. These callback functions are executed asynchronously, which means that although these functions appear to be registered sequentially in the code structure, they do not depend on the order in which they appear, but rather wait for the corresponding event to fire. The important advantage of Event-driven and asynchronous programming is that make full use of the system resources. The implementation of the code without waiting for a certain operation to complete, and the limited resources can be used for other tasks. This design is very suitable for back-end network service programming, which is the goal of Node.js. In server development, concurrency request processing is a big problem, and blocking functions can lead to the waste of resource and the delay of time. Through event registration, asynchronous function, developers can improve the utilization of resources, and performance will also improve. From the supported module provided by Node.js, we can see that many of the functions, including file operations, are executed asynchronously, which is different from traditional languages. In order to facilitate server development, Node.js' network modules are particularly large, Including HTTP, DNS, NET, UDP, HTTPS, TLS, etc., developers can build a Web server on this basis.

III. SYSTEM ARCHITECTURE AND TECHNICAL SOLUTIONS

A. System Architecture

The Express Supervision platform is divided into four main levels: data integration, data storage, data analysis and data application layer at the vertical level. The figure below shows the specific hierarchical structure layout of Express Supervision platform.

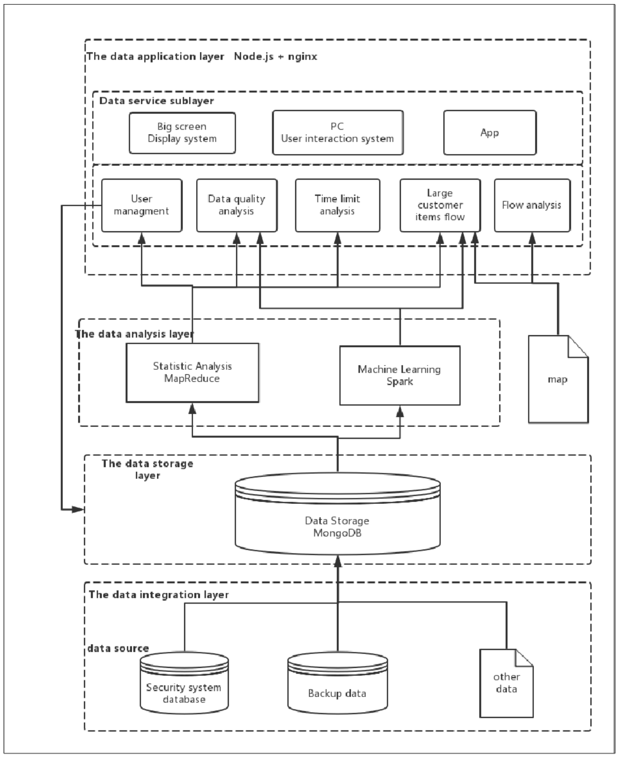


Fig. 1. System Architecture

In the data integration layer, we use real-time or dynamic method to export 11g data from the oracle database. At the same time, using the recovery method to export the backup data and using the data cleaning tools to achieve the data preprocessing. In the end, importing the data into MongoDB.

In the data storage layer, using the MongoDB to store data. Due to its fragmentation and weak consistency, MongoDB can ensure the speed of access. Coupled with the structure of its document storage, users can more easily access to the data.

In the data analysis layer, the system use the MapReduce to achieve the statistics analysis and Machine Learning to complete large data processing and analysis.

In the data application layer, Using Node.js + Nginx cluster, based on the standard restful cloud service interface to provide data services, the big data through the visualization has been presented in the front end.

B. Data Storage Solutions

The goal of the data storage solution for the express data is to meet the needs of storage, backup and high-performance read and write of large data (including historical data and real-time data) in the express delivery industry. It is designed to provide efficient data for data analysis and the basis for the protection of the performance.

a) Overview of the Overall Architecture: For the huge and complex data in Express industry, combined with the hardware performance and technical route requirements. The server cluster is divided into: analysis and application server group, Mongo DB storage server group. Analysis and application server group consists of two SX204-12 large data all-in-one machine, and each machine is equipped with four dual server nodes, which achieve the data analysis and processing, storage node routing and configuration control. Mongo DB storage server group consists of eight SX206-12 large data all-in-one machine, each large data one machine configuration 6 single server node, which achieve the data distributed storage[3]. It is shown in the below pictures:

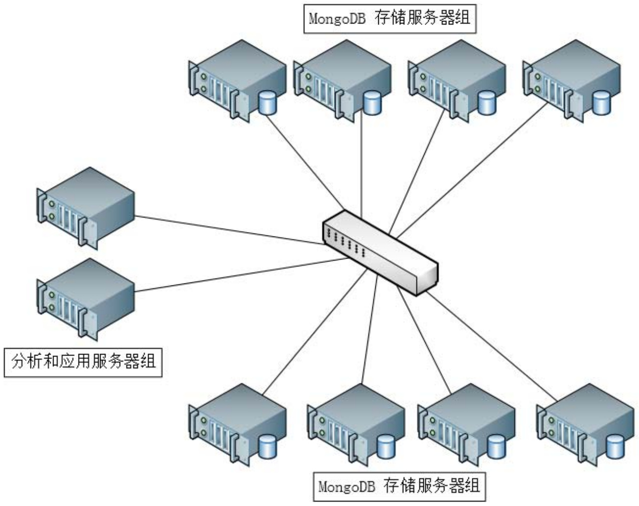


Fig. 2. System Architecture

b) MongoDB Distributed Storage Architecture:

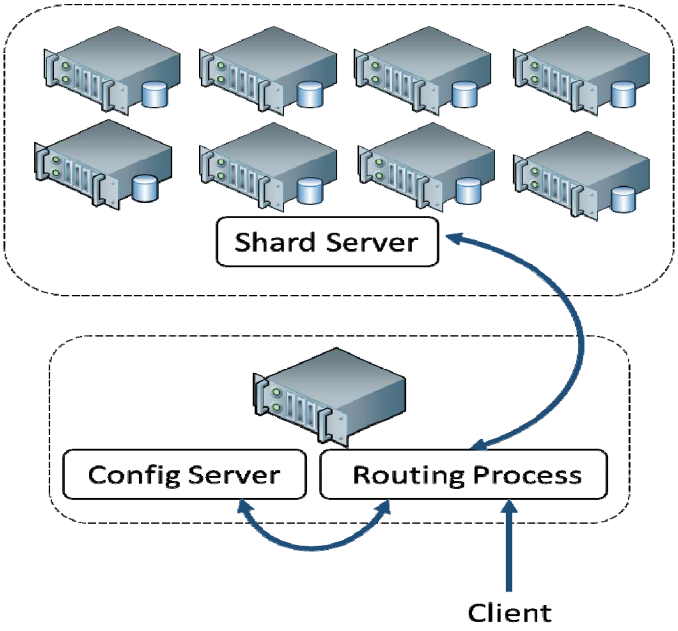


Fig. 3. MongoDB Distribute Storage Architecture Model

Figure 3 shows the MongoDB distributed storage architecture model, the specific components are as follows:

Shard Server: For the characteristics of express data, as well as robustness and efficiency of MongoDB distributed storage, the 48 nodes of storage server group are divided into 16 slices, with 3 copies in each slices(master, slave and arbitration node). Using the above configuration can be better to avoid a single point of failure of the host, but also automatically switch from the master and slave copy, taking into account the performance and efficiency[4].

Routing Process: As the routing service on the entire database storage speed and read and write performance will have a greater impact, it will be allocated for the three high-performance server nodes, responsible for managing the fragment. The client is routed through the front-end route so that the entire cluster looks like a single database. The client application can be transparently used. The Routing Process does not store data from the Configuration Server.

Configuration Server: these 3 nodes reused by routing server are responsible for storing the entire cluster configuration information, that is, data and fragmentation of the corresponding relationship between slices.

c) Configuration Settings:

Chunks: In order to avoid Chunk is too small, resulting in frequent movement of stored data, combined with the actual size of the data, we divided by the size of 256MB.

Data Block: According to the daily generation of 100 million data, each data is about 0.5KB. We calculated that a single set need to set 200 data blocks.

Data Organization: The data is stored in the physical storage node according to the size of the data block and

Chunks by using the date key as the key value and the label item is mapped. The efficiency of the data operation is improved evenly.

Index: In order to improve the efficiency of data manipulation and management, the data is indexed with key values that uniquely label data.

d) Database Structure

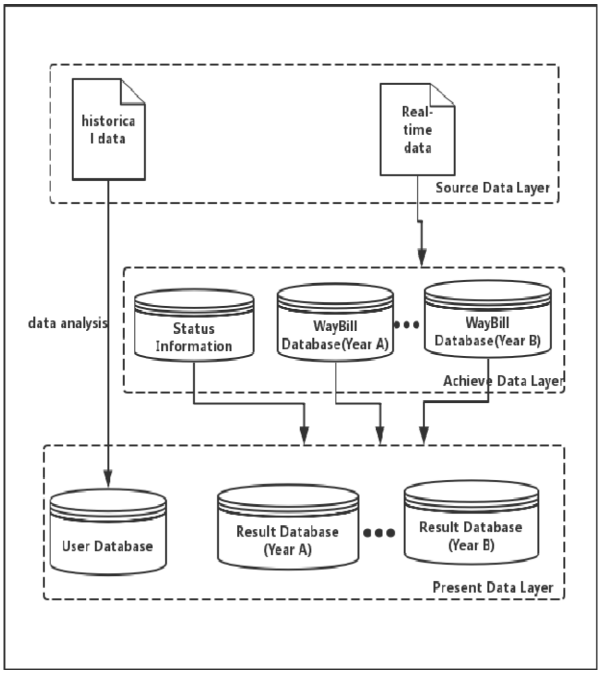


Fig. 4. Database Structure

Source Data Layer: All historical information is stored in the historical database, limiting a single set to a maximum of 100 million documents. Real-time data will be imported into the database, after its preprocessing, generating state information and waybills data temporary tables, which are into stored in a specific set after preprocessing.

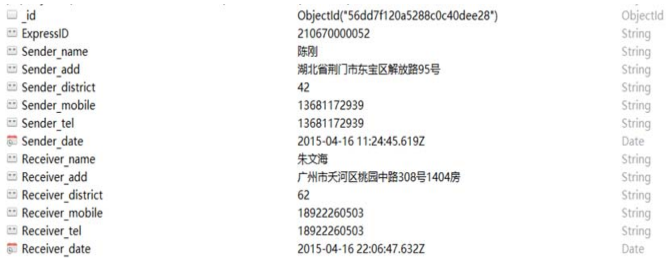
Achieve Data Layer: The information in the state information database is only used as the auxiliary information of the next stage data analysis. Therefore, it is not necessary to do preprocessing in the information storage process, and the storage method of the waybill database can be stored. In accordance with the year of the waybills, a waybill database is maintained every year.

Present Data Layer: The data of covering all time range(Including historical data, real-time data collection), and the user-related information are processed and stored in the database, limiting the individual collection to a maximum of 100 million (100 million) documents.

C. Statistics Analysis Solutions

Mongo DB is one of the most popular large data storage platforms, and it can be used to provide source data as the bottom storage layer of cloud computing technology, such as Spark, Hadoop, Pig, Hive, Drill and other computing framework. At the same time it provides the aggregation pipeline, MapReduce features to support the calculation of large data, statistics, classification and other needs. Aggregation pipeline and MapReduce are collectively referred to as aggregation operations in MongoDB.

a) Aggregation Pipeline: Aggregation Pipeline is a newly introduced aggregation framework for MongoDB 2.2, which follows the Unix-style data processing pipeline. The output is a result of the aggregation after that the documents in the collection enter the multi-level pipeline[5]. The first step in the processing pipeline may contain a plurality of components, including filtering, projecting, grouping, sorting, limiting, skipping and so on.



_id	ObjectId("56dd7f120a5288c0c40dee28")	ObjectId
ExpressID	210670000052	String
Sender_name	陈刚	String
Sender_add	湖北省荆门市东宝区解放路95号	String
Sender_district	42	String
Sender_mobile	13681172939	String
Sender_tel	13681172939	String
Sender_date	2015-04-16 11:24:45.619Z	Date
Receiver_name	朱文海	String
Receiver_add	广州市天河区桃园中路308号1404房	String
Receiver_district	62	String
Receiver_mobile	18922260503	String
Receiver_tel	18922260503	String
Receiver_date	2015-04-16 22:06:47.632Z	Date

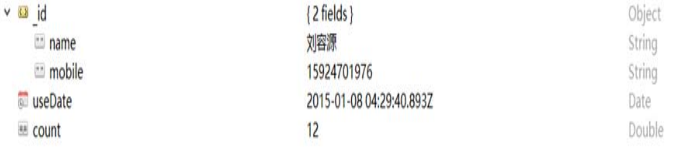
Fig. 5. Waybill Information

Figure 5 shows a waybill document that stores fields such as Receiver_name, Receiver_mobile, Receiver_date, and so on. The goal is to count the number of different users (differentiated by the Receiver_name + Receiver_mobile combination field) and the first use of time. The processing flow is divided into three steps, preprocessing, statistics and result output.

Preprocessing: Preprocessing: First, according to the Receiver_date field, using the sort command $sort of MongoDB to sort the data in the collection at the first stage of the pipeline. During this period, you can use indexes to sort quickly, and the result of this step is to provide an ordered data set for calculating the earliest time[6]. Otherwise it will be slower to sort after other processing steps and take up a lot of memory.

Statistics: Using the mongo DB grouping command $group to group the documents according to the specific fields(id). Specifically, specifying the Receiver_name + Receiver_mobile combination field as the _id field, which will identify different users. In this case, you also need to count the user's earliest time of using and the number. The earliest time can be queried using the $ first command, and $ first is more efficient in the sorted data set in the corresponding field. The number of users is passed through the $sum operator, which adds the value to the calculated result for each document in the group[7]. In the application case, setting value to 1 which can be used as a statistic value.

The output: the final step is to output the result, and the result document can be written to the target collection via the $ out operation, and the operation can also bypass the Mongo DB document size limit. The problem with the target set is described as follows: When the target set does not exist, the collection is created, which is visible after the end of the aggregation; when the target collection already exists, the $ out operation automatically overwrites the original collection.



∨ _id	{ 2 fields }	Object
name	刘容源	String
mobile	15924701976	String
useDate	2015-01-08 04:29:40.893Z	Date
count	12	Double

Fig. 6. User Information

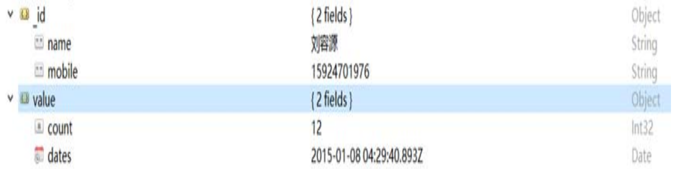
:

Where _id field records the user ID field, userDate records the user's earliest time, and count is the total number of times the user's use.

b) MapReduce: MapReduce is software architecture model proposed by Google for large-scale data sets (greater than 1TB) of the parallel operation. Concepts "Map" and "Reduce", and their main ideas are borrowed from the functional programming language, as well as from the vector programming language borrowed features. MongoDB implements MapReduce using the same processing paradigm. In the map function, the values in the current document are mapped to the key-value pair and emit; the reduce function takes the arguments and merges the values of the same key into a document. By dividing the calculations into parallel add operations, MapReduce can be to achieve large-scale data computing functions[8].

For example, we assume that the goal is to obtain the total amount of shipments of different users and the time of the first use. First implementing the map part, the Receiver_name, Receiver_mobile are taken as a key to group, and setting the value of {'count': 1, 'dates': [this.Receiver_date]}. The count field is the number of users, and the dates array stores the user time data. The format of the data is used to ensure the consistency of the values and reduce the time array.

In the reduce phase, the value of the current key is counted. Count field can be a simple summation. After the dates array merges the time data from the current values into an array, it need to be reassigned the value of the smallest time in the array. The other optimizations are sorted using the sort key of the map, which reduces the number of operations[9]. Finally, our calculations are shown in Figure 7.



∨ _id	{ 2 fields }	Object
name	刘容源	String
mobile	15924701976	String
∨ value	{ 2 fields }	Object
count	12	Int32
dates	2015-01-08 04:29:40.893Z	Date

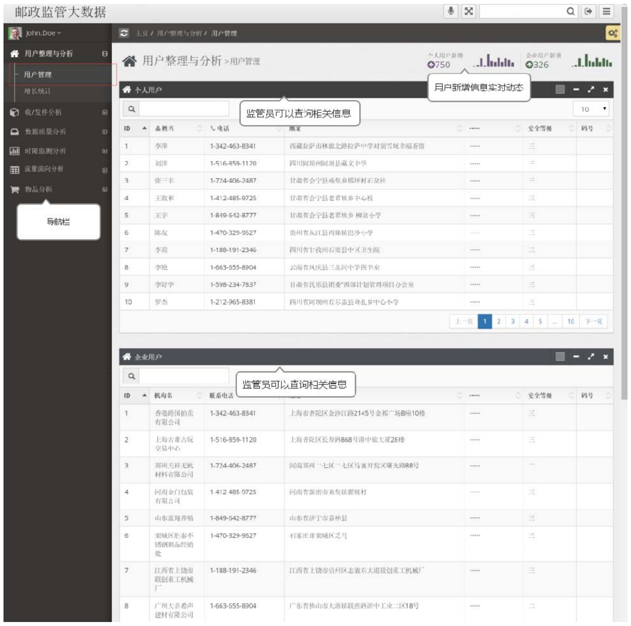
Fig. 7. User information

IV. SYSTEM DISPLAY

A. User Query

The supervisor can search for the relevant information and data in the user database. The user database information is mainly divided into individual users and institutional users. You can browse and access information about individual users and institutional users. Personal user database information items include: name, address, telephone, ID number, code number, etc. Institutional user database information items are divided into: organization name, address, telephone, organization code, code number and other information.

The user's statistical data can be obtained by the three steps of the above data aggregation processing. The result is shown in the following figure



ID	姓名	电话	地址	······	安全等级	码号
1	李萍	1-342-463-8341	西藏拉萨市林廓北路拉萨中学对面雪域幸福茶馆	······	三	
2	刘洋	1-516-859-1120	四川阿坝州阿坝县藏文中学	······	三	
3	张三丰	1-724-406-2487	甘肃省会宁县杨集乡那坪村石家社	······	三	
4	王政和	1-412-485-9725	甘肃省会宁县老君镇乡中心校	······	三	
5	王宇	1-849-542-8777	甘肃省会宁县老君镇乡 柳家小学	······	三	
6	陈友	1-470-329-9527	贵州省从江县丙妹镇巴沙小学	······	三	
7	李霞	1-188-191-2346	四川省甘孜州石渠县中区卫生院	······	三	
8	李艳	1-663-555-8904	云南省凤庆县三岔河中学图书室	······	三	
9	李好学	1-598-234-7837	甘肃省民乐县团委"西部计划管理项目办公室	······	三	
10	罗杰	1-212-965-8381	四川省阿坝州若尔盖县降扎乡中心小学	······	三	

ID	机构名	联系电话		······	安全等级	码号
1	香港跨国拍卖有限公司	1-342-463-8341	上海市普陀区金沙江路2145号金祁广场B座10楼	······	三	
2	上海古董古玩交易中心	1-516-859-1120	上海普陀区长寿路868号港中旅大厦28楼	······	三	
3	郑州天祥无机材料有限公司	1-724-406-2487	河南郑州二七区二七区马寨开发区曙光路88号	······	一	
4	河南金门包装有限公司	1 412 485 9725	河南省新密市来集镇郡城村	······	三	
5	山东蓝翔养殖	1-849-542-8777	山东省济宁市嘉祥县	······	三	
6	栾城区胜泰不锈钢制品经销处	1-470-329-9527	石家庄市栾城区之号	······	三	
7	江西省上饶市联创重工机械厂	1-188-191-2346	江西省上饶市信州区志敏东大道联创重工机械厂	······	三	
8	广州大音希声建材有限公司	1-663-555-8904	广东省佛山市大沥镇联滘路沥中工业二区18号	······	二	

Fig. 8. User Query Page

B. Statistic Chart

a) Regional User Statistics: Users can browse the real number of users in the provinces dynamically. Users can also see the number of national, provincial and municipal users and the corresponding trends. On the histogram, the user understands the number of users in each province by the depth of the color and the situation. Through the realization of the visualization tool, you can also choose to select a province to browse to the province each city each county's new user statistics.

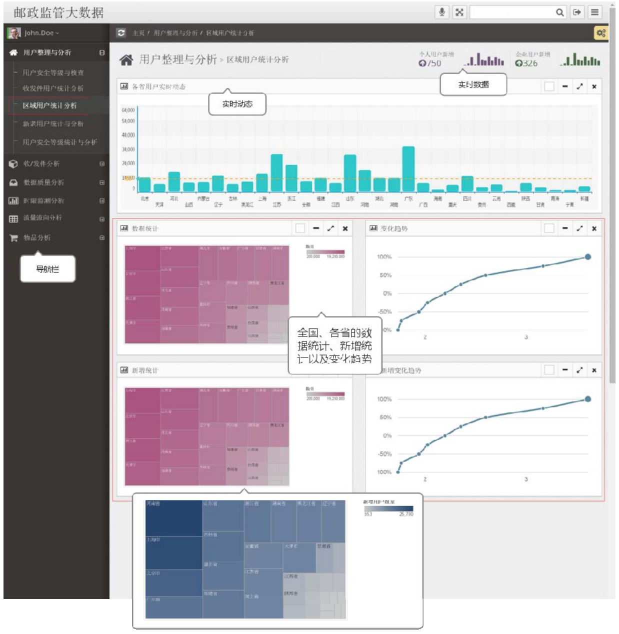


Fig. 9. Regional User Statistics

b) User Safety Level Statistics: With the addition of dynamic users can understand the users increased by real time, and you can choose to browse to different levels of security by the number of users statistics and changes in the situation, so that different levels of user data statistics can become more intuitive and clear. Users can also see the growth of different security levels of users, as well as the proportion of all different levels of users.

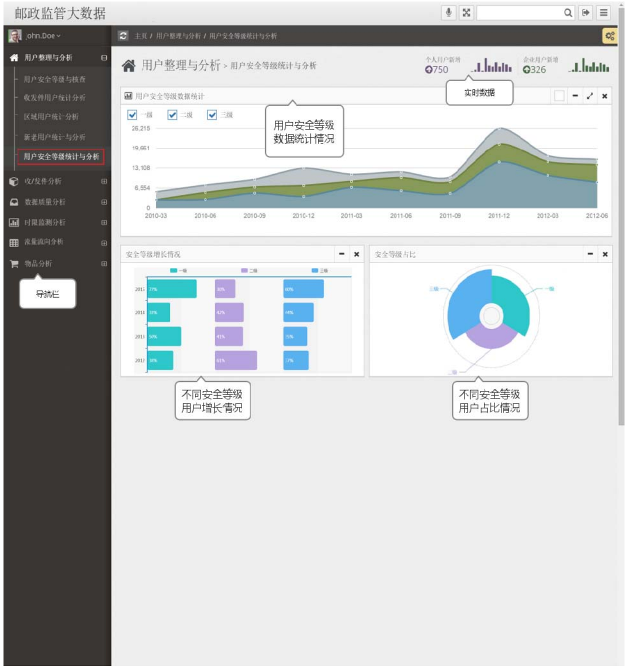


Fig. 10. User Safety Level Statistics

V. CONCLUSION

This paper introduces the construction of the Express Supervision System. First it introduced the key technology used in system construction. Then it mainly introduces the system architecture, the data storage solutions based on MongoDB database and the statistical analysis solutions based on MapReduce. Finally, this paper presents user query and statistical analysis modules for the Express Supervision System. In the future, we will use the method of machine learning to optimize the statistical analysis method and get a more accurate analysis result.

ACKNOWLEDGMENT

This paper is partly supported by "Key Cultivation Engineering Project of Communication University of China (Project number: 3132016XNG1606 and 3132016XNG1608)", "Cultural technological innovation project of Ministry of Culture of P.R.China (Project number: 2014-12)", and partly supported by "The comprehensive reform project of computer science and technology, department of science and Engineering". The research work was also supported by "Chaoyang District Science and Technology Project (CYXC1504)".

REFERENCES

[1] Y., G., et al. Analysis of data storage mechanism in NoSQL database MongoDB. in 2015 IEEE International Conference on Consumer Electronics - Taiwan. 2015

[2] V., A. and M.R. C. MongoDB and Oracle NoSQL: A technical critique for design decisions. in 2016 International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS). 2016.

[3] Zhu, Q. and P. Zhou, The System Architecture for the Basic Information of Science and Technology Experts Based on Distributed Storage and Web Mining. 2012. p. 527-530.

[4] Kookarinrat, P. and Y. Temtanapat, Analysis of Range-Based Key Properties for Sharded Cluster of MongoDB. 2015. p. 1-4.

[5] Bonnet, L., et al., Reduce, You Say: What NoSQL Can Do for Data Aggregation and BI in Large Repositories. 2011. p. 483-488.

[6] Hong, T.P., et al., Efficient data preprocessing for genetic-fuzzy mining with MapReduce. 2015. p. 88-89.

[7] Arora, S. and I. Chana, A survey of clustering techniques for big data analysis. 2014. p. 59-65.

[8] Nabavinejad, S.M. and M. Goudarzi, Faster MapReduce Computation on Clouds through Better Performance Estimation. IEEE Transactions on Cloud Computing, 2017. PP(99): p. 1-1.

[9] Pol, V.V. and S.M. Patil, Implementation of on-process aggregation for efficient big data processing in Hadoop MapReduce environment. 2016. p. 1-5.