

The construction of repository for livestock breeding based on mobile technology

Xiao-Bin Qiu, Hong-Qian Chen, Jun Qiao and Nan Zhou
Network Center, China Agricultural University, Beijing, China
Email: qxb@cau.edu.cn

Raising livestock and poultry is a high professional requirement, thus a repository is of great significance for the precipitation and sharing of knowledge about raising and disease prevention. Due to the characteristics of this kind of repository and the current development of mobile Internet, this system uses a mobile development framework named PhoneGap. This framework can support Android, iOS, Windows Phone and other mobile operating systems. It is convenient for developing and it shows high performance like Native app. Hadoop is a distributed storage with high extensibility and is used to store the unstructured data, i.e. images, documents and so on. Each query takes only 40 ms when the number of concurrency is 1,000 in the test. It can meet more requirements because the system is extensible. The test shows that the system can not only achieve the convenience of mobile operations, but also can also meet the needs of high concurrency.

Keywords: Livestock; Repository; PhoneGap; Hadoop.

1. Introduction

Repository is an infrastructure of academic research for the collection, conservation, management, retrieval and utilization of research output. Its role is mainly reflected in centralized management, long-term preservation, sharing and improving the academic influence and prestige of knowledge production. Being recognized and used by more and more organizations, the construction of repository has become the bottleneck of its development. Compared with construction abroad, the quality of our repository is uneven, the standard specification is not perfect, there is a lack of sharing methods and so on[1]. In the livestock breeding industry, the number of farmers in rural areas has increased rapidly in recent years, but the level of farming is uneven, so the demand for Repository of Livestock Breeding is high. With the rapid development of mobile Internet, great changes have taken place in the collection and way of use. Almost every farmer has a smart phone, and mobile repository is more flexible, timely and convenient to provide technical support and services. Based on these characteristics, this paper proposes the construction model of this

kind of repository for livestock breeding with mobile Internet technology [1][2][3][4].

2. Key Technology

2.1. Mobile Development framework

The mobile operating system mainly includes Android, iOS, Windows Phone and so on at present. Java is the development language for android and objective-c is for iOS. Web technology includes HTML5, CSS3, Javascript, etc. The mature development framework mainly has PhoneGap, Titanium and so on. While PhoneGap is a framework that can support Android, iOS, Windows Phone, etc. It encapsulates many system level APIs into Javascript APIs, developers can use the Javascript APIs to handling operations such as global positioning, accelerating, managing address book, sounding, vibrating, etc. It loads the application by WebView. A lot of developing framework is optimized based on the PhoneGap, for example Worklight, appMobi, WeX5, etc. The performance is close to the Native app. Titanium is another important one, it has a built-in Javascript engine, and compiles some UI components into Native Code, so it displays high efficiency and high speed in loading pages, but it cannot be moved directly to other platforms to be used [5][6][7].

2.2. Distributed Storage System

Distributed storage system is a kind of storage system that scatters the data in many independent devices. In traditional storage system, all the data is stored on the servers. This poses as a bottleneck for performance and focus is centered on reliability and safety, thus preventing the system from meeting the needs of large-scale storage applications. On the other hand, in a distributed storage system, it is an extensible system architecture that shares memory load over multiple storage servers and locates storage information with position servers. Not only does it improve the reliability, availability and accessing efficiency of the system, but it is also easy to extend.

Current distributed storage systems include Hadoop published by Apache, GFS by Google, TFS by Taobao, QFS by Tencent, Haystack by Facebook, and so on. GFS is one of the most famous storage systems. HDFS, MooseFS, MogileFS are based on GFS. TFS, QFS and Haystack architecture is similar, also known as FS Blob (File System Blob) [8].

3. System Architecture

The system architecture is made up of two parts: The client for the display and operation, including uploading the collected information for the server, obtaining the basic categories of data from the server and storing it on the local database. The other is the collecting and handling the repository data by the server, and feeding the results the client according to the demand.

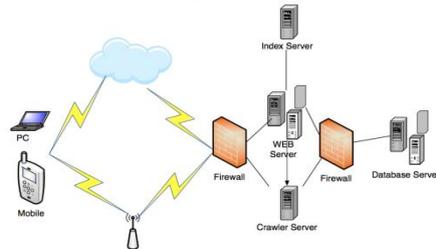


Fig.1. System architecture diagram

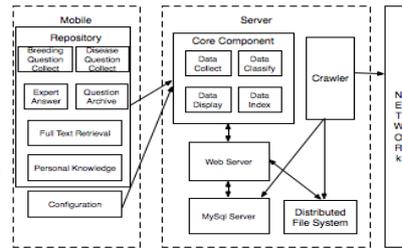


Fig.2. Design Diagrams of Repository

As shown in Fig.1, the client retrieves the information of the repository and submits the collection information to the server by a mobile phone or PC via 3G, 4G or Wifi. The Server contains three parts. The first part is Web Server that interacts with the client, including information inquiry and submission. The second is Web Crawler that collects the related repository information from network and then saves it in the databases. The last is the TRS server that creates index for the mass data, and provide efficient retrieval services to client.

4. System Design and Implementation based Mobile

4.1. System Design

The system is composed of two parts, mobile terminal and server. The mobile terminal realizes the information displayed and question submitted. The server can collect and classify the data from network automatically, handles the request and feeds it back to the mobile in real time. As shown in fig.2.

4.2. Mobile Terminal

The mobile terminal mainly integrates a knowledge portal that displays all information, so the mobile can learn about new archival information at any time, getting functions efficiently from the portal and finding the required information quickly and accurately. The knowledge portal provides users with a macro perspective and places the main information on the best position, so we use and query the information conveniently. The main functions of the portal include

collecting questions, expert answers, archiving questions, full text retrieval, personal knowledge, and so on. The contents include breeding methods, breeding experience, disease controlling, disease preventing and curing, etc. The part of question collection uses two models, they are online and offline. When wireless signal (3/4G or Wifi) are all good. The data transmits and saves on the server in real time, otherwise the system can save the data into the local database, and upload to the server after the signal is better. In the section of expert answers, you can get the status of question and query the expert answers at any time. At present the main mobile operation system includes Android, iOS, Windows Phone, Symbian, BlackBerry OS. According to our sampling survey, more and more farmers are using iPhones, but the number of android users is more than 80 percent, so the development based on android is the key point of this project. This Paper takes Android as an example to introduce the realization of the question collection module. In order to provide better compatibility with future applications in other operating systems, this system uses the mobile framework of PhoneGap that can support iOS, Android and other operating systems.

The general process of question collection is, opening the camera – taking a picture - describing the problem – submitting. If the wireless signal is good, the question will be submitted to the servers, or saved locally, then automatically uploaded when a Wifi signal is detected, or manually uploaded when 3/4G is good. The core code is as follows:

```

Camera mCamera01 = Camera.open();
Camera.Parameters parameters = mCamera01.getParameters();
parameters.setPictureFormat(PixelFormat.JPEG); /* Set Photo format JPEG
*/
mCamera01.startPreview();//Preview photo
mCamera01.takePicture (shutterCallback, rawCallback, jpegCallback);
ConnectivityManager
conn=getSystemService(CONNECTIVITY_SERVICE); NetworkInfo info =
connectivity.getActiveNetworkInfo();
SharedPreferences sp =
getSharedPreferences("Datadefault",MODE_PRIVATE);
SharedPreferences.Editor editor = sp.edit();//Get edit object
editor.putString("keyname","");
editor.commit();//submit
sp = getSharedPreferences("Datadefault",MODE_PRIVATE);
String readstring = sp.getString("keyname","if no default value");
HttpClient httpclient = new DefaultHttpClient();

```

```
String urlStr = "http://xx/upload"
HttpPost httpPost = new HttpPost(urlStr);
HttpResponse response = httpClient.execute(httpPost);
Finally, check the value returned and verify that it is successful.
```

4.3. Service Server

The service server is based on the web server, the core components include data collection, data classification, data display and data index. The core components realize the data collection, classification and construction index, then displays the result on the PC or mobile according to the needs of users. The system background is based on the Phalcon architecture, using PHP to implement. The background implements information processing, crawler crawling, index establishment, etc, and display through Html5. The main contents include:

(1) Collection and Presentation of Data. In this function, users submit the questions about breeding & disease, through words or pictures, then the server saves the data in the database, saves the picture in the distributed storage system, then saves the picture's address in the database. At the same time, the system can provide real-time issues to submit, deal with the situation, if the problem has the answer, it will be sent to the client through message. Since the data displays mainly through HTML5, it can be shown in Andriod, iOS, etc., and even in WeChat, because it is less dependent on the mobile platform.

(2) Establishment of Data Index. The key to the establishment of Chinese data index lies in the word segmentation, and the good or bad of word segmentation is in great relationship with the word segmentation algorithm and thesaurus. This system uses Language Model + CRF. The CRF is the best program in the Chinese word segmentation competition, and the Language Model is also very good when the word frequency statistics is correct. Compared with CRF, its effect of word segmentation is stable, and easy to adjust. Once the word segmentation gives errors, you can add new words to correct word segmentation results. Therefore, the word segmentation strategy of the system is to first identify new words by CRF, and then use Model Language.

(3) Knowledge Crawler. It is a kind of supplement to the repository collection of information, which can collect the data of similar subject in a short time. The crawler of the system is based on the Scrapy framework, implemented by Python. In order to satisfy the requirements of the crawler, the distributed cluster design is used. A better performance server is used for the Master, which is responsible for the management of URLs, and it places all the next level URL of current pages into a queue, then ensures that the URL queue data has removed

the duplicate data. The Master is responsible for scheduling the queue, any other crawler servers Slaves. When a Slave completes a crawler mission, it applies for a new crawler task, and the URL is accessed in the Redis.

4.4. Testing

In the testing, we deploy two virtual machines, which are configured for 8G memory and 8 CPUs. One for web the server and one for database server. ApacheBench is used to perform stress tests. We select three parameters in the most important parameters; transactions per second, average transaction response time and current network status. [9] Shown as Tab. 1:

Tab. 1 Result of Stress Test

CN	TPR	NRPS	NF
100	143.461	6.97	4.18
200	60.075	16.65	9.98
300	59.551	16.79	10.07
400	90.79	11.01	6.6
500	61.521	16.25	9.75
600	69.809	14.32	8.59
700	48.034	20.82	12.48
800	48.225	20.74	12.43
900	41.852	23.89	14.33
1000	42.691	23.42	14.05

CN: Concurrent number

TPR: Time per req (ms)

NRPS: Number of responses per second

NF: Network flow (Kb/s)

The test results show that when the number of concurrent number is increased to 1000, each request time is about 40ms, the number of requests per second is up to 23, the indicators can well meet the needs of repository. Because the nginx is used for load balancing, so it can expand easily. When an index exceeds the standard, it can directly increase the number of servers so as to improve concurrent performance.

5. Conclusion

With the rapid development of mobile Internet, repository based mobile has obvious advantages in daily use. It meets the requirements of livestock breeding, especially in rural areas. This paper not only puts forward the system architecture, for the overall design of this kind of repository platform, but also

gives the concrete realization based on the related technologies of mobile Internet. Through this repository, we can submit the questions about livestock breeding or diseases real-time, and receive relevant answers in a timely manner. However, there are still a lot of work to be solved in the aspect of security, reliability and accuracy of repository.

References

1. Xiaohong Qiu, Yin Jiang. Analysis of the Present Situation and Characteristics about the Content Construction of Institutional Repositories in China and the USA [j]. *Documentation, Information & Knowledge*. 2015(3):113~119.
2. SUN Zhong-fu, DU Ke-ming. Perspectives of Research and Application of Big Data on Smart Agriculture. *Journal of Agricultural Science and Technology*. 2013,15(6):63~71.
3. Qiao Huan, Jiang Ying. Present and thinking of the content construction of Institutional Repository[j]. *Library Work And Study*. 2011(12):67~70,119.
4. Yan Xiaoping. Design and Research of Credit Bank Information Platform based on Cloud Architecture [j]. *Distance Education in China*. 2014(5):66~71.
5. Shang Minghua, Qin Leilei. Risk Information Collection System of Wheat Based on Android Mobile [j]. *Transactions of the Chinese Society of Agricultural Engineering*. 2011,27(5):178~182.
6. LI Ru-ping, XU Zhen-yu. Internet of Things System based on ASP.NET and Android Platform[j]. *Journal of Jiamusi University(Natural Science Edition)*. 2015,33(5):710~713.
7. Liu Yingcheng. The Design of Smart Phone-Based Tour Guide System[j]. *Computer Applications and Software*. 2014,31(11):248~250.
8. Li Tie, Yan Cairong. Optimization of Small File Storage and Accessing on Hadoop Distributed File System[j]. *Journal of Computer Applications*. 2014,34(11):3091~3095,3099.
9. Yang Yi-jun, Huang Da-qing. Research and Development of Automated Performance Test Tool for Android Smartphone [j]. *Journal of Computer Applications*. 2012,32(2):554.