MATEC Web of Conferences 22, 01025 (2015)
DOI: 10.1051/matecconf/20152201025
© Owned by the authors, published by EDP Sciences, 2015

A study on the unstructured music database

——Taking the Bo people's music and its music iconography database as an example

Yutong Liu*
College of Music and the Performing Arts, Yibin University, Yibin, Sichuan, China

ABSTRACT: An unstructured music iconography data system constructed by key technologies like Dublin Core, Lucene technology and MVC framework is studied in this paper. Results indicate that the traditional directory tree and the existing indexing and searching tools are severely insufficient in the organization and management of the massive unstructured data. Relevant documents can be searched effectively and rapidly through the index established by <keywords, files> provided by BeFS. Key technologies, such as Dublin Core, Lucene technology and MVC framework, can be applied to the construction of the enormous unstructured database of music and image resources. The database system test can be divided into two links, functional test and performance test. The test results of the Bo people's music and image database system obtained through the tested design scheme indicate that the performance of the system is relatively high and able to satisfy the concurrent access of massive data with excellent user experience.

Keywords: unstructured data; image database of the Bo people's music; Dublin Core; Lucene technology; MVC framework; system test

1 INTRODUCTION

Long Wen (2009) pointed out that information is generally divided into structural data and unstructured data [1]. Structural data refers to numbers and symbols whereas unstructured data refers to texts, images, sounds, and so on. The Bo people's music iconography database is a thematic database with objects of the Bo people's music and musical images of the Bo people's cliff paintings in the Gong County of Sichuan province. This paper studies the unstructured data resources, aiming at exploring key technologies and system test methods applied in the construction of the thematic database.

Many people have made efforts in studies on the unstructured database and the thematic data of music. Teng Teng (2015) pointed out that the big data of music can not only "predict the future" but also speculate and dig the music resource utilization deeply in the Internet era [2]. Haiyuan Wu (2014) discussed the construction mode of the northeastern folk music database with such social functions as modern education and resource sharing [3]. Peishun Ye (2015) studied two basic routing strategies and the heuristic P2P searching strategy of the unstructured P2P network, analyzed seven kinds of heuristic searching strategies of the unstructured P2P network, made comparisons of these strategies, and discussed the implementation mechanism as well as merits and demerits of each strategy [4]. Ying Yang (2013) constructed a data warehouse model based on unstructured data with XML and a video retrieval platform of the unstructured database [5].

Studies of the above scholars suggest that music and image resources are massive unstructured data. This kind of database can be only established in a small range. Traditional methods have certain difficulties in dealing with massive unstructured data. Based on previous studies, this paper analyzes the applicability of Dublin Core, Lucene technology and MVC framework in constructing a system of massive unstructured music and music iconography data so as to explore effective construction methods and test means for the database system.

2 UNSTRUCTURED DATA ORGANIZATION

With the development of science and the progress of the society, the popularization of network technologies is now in a state that cannot be surpassed. Disadvantages of the file database become gradually prominent against this background. The most severe disadvantage lies in the aspect of semi-structured and unstructured data processing. As for the organization and management of unstructured data, there are such methods as file directory tree, index and retrieval, semantic system, and so on [6]. In order to study the unstructured thematic database of music class, this paper first analyzes the three methods of organization and management mentioned above so as to provide a theoretical basis for the research object.

*Corresponding author: liuyutong123@163.com

Article available at http://www.matec-conferences.org or http://dx.doi.org/10.1051/matecconf/20152201025

2.1 *The method of file directory tree*

Data and information are frequently stored and organized in the form of files inside the computer. The organization and management of data [7] can be also regarded as the organization and management of files. At present, the directory tree method is most commonly used in the structure of file management. This method classifies and manages files in accordance with path names of files. The advantage is that users can set up a path name through the understanding on the file content and store the file in a certain path accurately.

The path name of a file includes the semantic meaning and the function of the physical address. It means that users carry out the logic management through file path names whereas the operating system carries out semantic block operations on files through file path names [8]. The management efficiency of the directory tree method becomes lower with the continuous growth of file data. When the number of files becomes massive, the traditional directory tree method has the following disadvantages:

1) Files can be only classified into one category. With the refinement of the classification, it becomes more and more difficult to select the exclusive category for each file. Although a file can be classified into multiple categories through the linking scheme, the scheme cannot satisfy the practical usage of users perfectly. In the meantime, it brings the complexity of reclassification. The link fails to reflect corresponding changes of the source file when the source file is moved or deleted.

2) Lots of detail information of files will be lost if there is no good naming method. For example, the naming of music files MP3 fails to include all the information, such as the singer, the genre and the rhythm.

3) Related file sets must belong to the same subset; otherwise the relationship cannot be expressed.

4) A user must remember the full name of a file in the file retrieval. However, this memory is quite rare in practical applications.

When a large amount of unstructured data is managed through the traditional method of file directory tree, users are easily caught in a dilemma that they want to classify files in detail but they fail to remember absolute pathnames marking the detailed classification of files.

To sum up: the traditional method of file directory tree is not applicable in the management and organization of massive unstructured data.

2.2 *Index and retrieval*

Index first appeared in the bibliography system. In this sense, index refers to a system guide for items contained in a literature collection or the extraction of a concept from the literature collection [9]. These items or extracted concepts are expressed with articles of the known or explained checkable sequence. Due to the advent of the computer, the index technology gets its rapid development in modern times, especially the index technology of the database system.

“Index”, as a term of database, refers to the sequence of data in a database in line with a specific domain (or attribute). The specific domain is called the key domain [10]. The corresponding indexing service is to extract information from data according to the index and then provide users with it after organizing and analyzing the information effectively. Indexing technologies that are similar to reading and consulting the catalog are adopted in the database so that the required content can be rapidly looked up without scanning the whole database. In the practical application process, users are unable to remember absolute paths of all data information. But they can provide certain features of data information required by the scanning. Users also hope that the small amount of feature information can be utilized to reduce the data file set so as to find the data required by themselves quickly. At present, most file systems provide users with relevant retrieval tools by establishing some specific indexes, such as UINX/Linux [11], Windows, Google Desktop [12], Yahoo, Desktop [13], Windows Desktop [14], and so on. The index of a search engine is different from the index of a database. But they have the same function. The index of a search engine not only provides the interface between unstructured data and users but also provides technical supports for users to find data rapidly and accurately [15, 16]. The diagram of the index of a search engine is shown in Figure 1.

The advantage of existing indexing and retrieval tools is that users are able to find the required data rapidly with some simple operations. But there are certain defects, because indexes are predefined or obtained from analyses on the absolute paths of files. Both the accuracy and the usability cannot meet the requirements of massive unstructured data usage.

2.3 *Method of semantic file system*

Many institutes have developed indexing methods based on attributes in order to get rid of the defects of hierarchical cultural directory tree [17]. BeFS provides the method of indexing a file set through attributes. Users are able to find relevant files with keywords through the indexing structure established by <keywords, files>.

The semantic file system [18] normally uses <classification, value> to provide files with retrievable mappings. The classification is also called file’s attribute, which can be obtained through user input or other methods, such as the full text analysis and data extraction of file paths. Once the attribute is established, users are able to set up the virtual folder of this attribute and all files containing this attribute can link to this virtual folder. If there is an inheritance relationship between two attributes, the parent attribute can be expressed in the form of virtual subfolders.

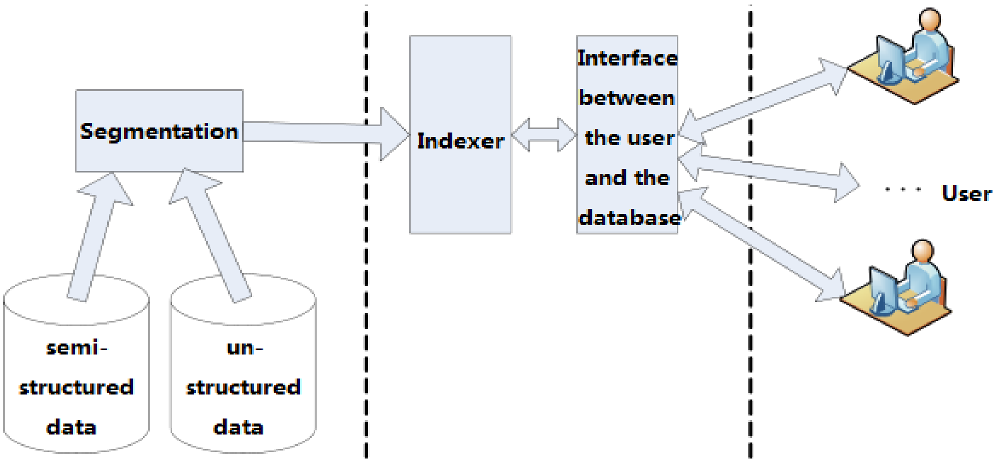

Figure 1. Indexing diagram of the search engine

Table 1. Dublin Core [24]

Category	Names	Explanation
Descriptions of resource content	Title	Resource name
	Subject and Keywords	Themes and keywords of resource content
	Description	Brief content of resource
	Source	Source information of resource
	Language	Language used for resource description
	Coverage	Time and space involved in resource
Descriptions of intellectual property	Creator	Resource creator
	Publisher	Resource publishing organization
	Contributor	Other contributors
	Rights	Resource copyright information
Descriptions of external attribute	Date	Date and time of resource generation or effective data and time
	Type	Resource type and category
	Format	Data format of resource
	Identifier	Identifying information assigned to resource in line with relevant regulations
	Relation	Reference to related resources

3 KEY TECHNOLOGIES OF ESTABLISHING THE UNSTRUCTURED MUSIC DATABASE

From the resource quantity, the Bo people's music and its music database belong to a massive unstructured database. This paper discusses the key technologies of constructing a highly extendible general unstructured database system based on the unified data storage standard, aiming at carrying out a study on the unstructured music database. The music database system discussed in this paper is based on the Dublin Core Metadata. It formulates the metadata standard of resource unification by combining the characteristics of the Bo people's music and its music iconography resources. After the resource encapsulation of the metadata standard, the key technology Lucene can be used to create an index in data retrieval and MVC framework can be used to process interactions between operations and data.

3.1 *Dublin Core Metadata Set*

Dublin Core is a brief directory model used for identifying electronic resources [22]. It is formulated by the international Dublin Core Metadata Initiative (DCMI). It is the common core standard that should be followed by all Web resources [23].

Dublin Core only defines and describes the minimum core set of resources. It consists of 15 items of core metadata that are provided in Table 1.

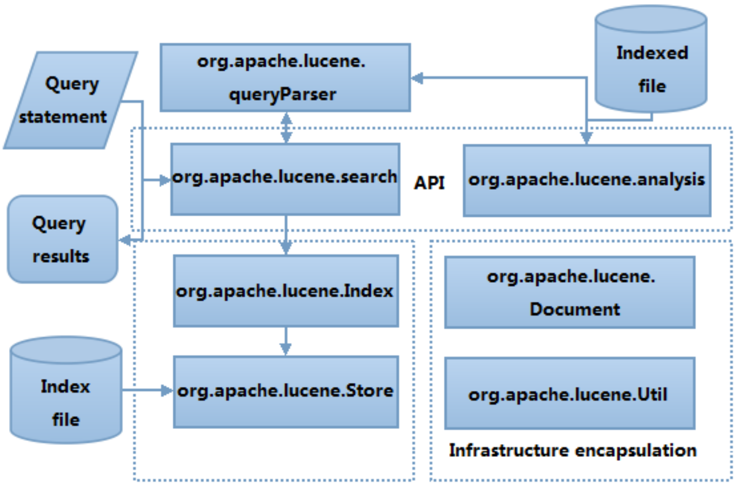

Figure 2. Structure chart of Lucene [28]

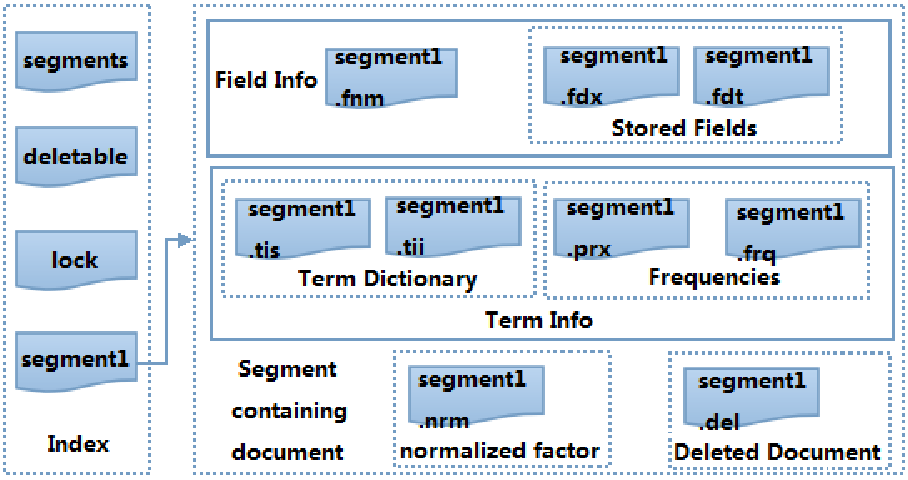

Figure 3. Structure chart of Lucene index

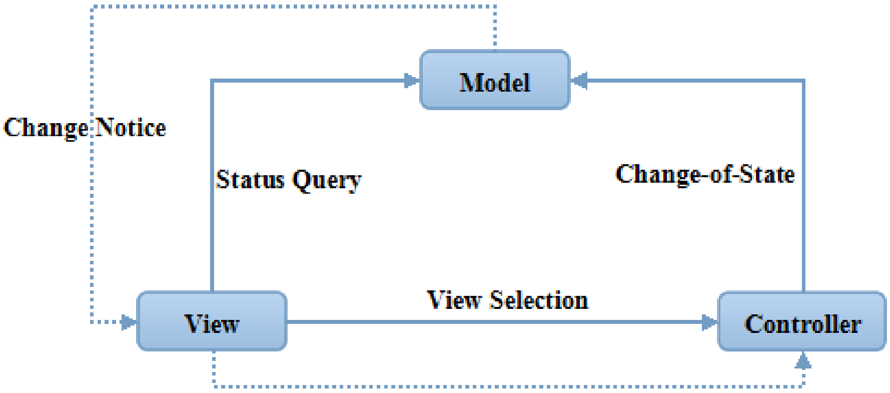

Figure 4. MVC model structure

Functions of metadata like retrieval, positioning and evaluation are the collection of attributes related to content and formats of digital resources [25]. Metadata has the function of electronic resource directory, which can be used to represent the content of electronic resources and other attributes. With the increase of electronic resources, the significance of metadata in its organization becomes greater and greater. On the one hand, metadata is able to store such information as formats and content of electronic resources. On the other hand, metadata provides exact storage locations of electronic resources [26]. The design principle of Dublin Core is that the core metadata set features extendibility, the minimum scale and definite significance. Four principles are followed in terms of design, including the simplicity of construction and maintenance, generally accepted grammar, multiple language culture and strong extendibility.

3.2 *Lucene technology*

Lucene is an open source project of the Jakarta project group of Apache software foundation. It is a full-text search engine toolkit with the function of data indexing and full-text retrieval [27].

Lucene source code consists of seven components, namely query analyzer, language analyzer, lexicon, data storage, index and retrieval. The organizational structure is shown in Figure 2. Lucene index contains multiple files that are respectively stored in groups according to the segment. Indexed files of the same group have the same filenames but different expanded-names [28]. The structure of Lucene index is shown in Figure 3. Each file of the segment contains domain set information and item set information. There is a corresponding relationship between indexed files and information files, which means that information files are sequenced in accordance with indexed files. In addition, the corresponding relation between domain set information and item set information can be also realized by domain record numbers of domain record files.

Normalized factor files and deleted files are respectively used in the scoring and ordering mechanism of indexing results and pseudo delete operations. Both of the two provide basic functions. These files constitute the index information of a segment [29]. The whole retrieval process of Lucene is shown below [30]:

1) Create a textbox and save all the information that might be retrieved by users. Determine the text model of the index. The text model is an information format approved by the system. Upheavals should be avoided after the determination.

2) Create an index according to the data text. In this process, correct indexing methods should be correctly selected in line with the retrieval system scale.

3) Retrieval. Users firstly submit a retrieval request and then the system analyzes the request. Carry out relevant processing through text operations.

4) Filter and sequence the indexing results in line with certain rules and then return the results to users. A Lucene retrieval operation is thusly completed.

3.3 *MVC framework*

MVC (Model View Controller) is a model of the program development and design. It realizes the separation of business logic and data display [30], improves the maintainability, transportability, extendibility and reusability of programs, and reduces the development difficulty of programs.

The MVC framework includes three cores, namely Model, View and Controller. The structure is shown in Figure 4. As the main part of the MVC application program, Model consists of a business logic module and a data module. View is the interface of users' interactions, consisting of jsp and html. Controller processes nothing, which only receives requests from users and invokes Model and View to fulfill users' requests. Struts2 framework is the most popular framework in MVC framework field. The structure of the Struts2 controller is shown in Figure 5. It is composed of a scheduler, an interceptor and a service controller. The operating principles are: (1) The scheduler selects corresponding interceptor groups for preprocessing after receiving an http request from the page and then invoke the corresponding service controller; (2) the service controller processes requests by calling the service logic layer structure; (3) return the results to the interceptor group and then return to the page for rendering after the processing.

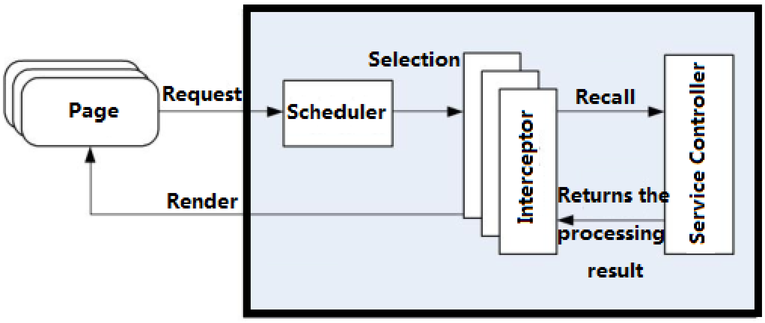

Figure 5. Structure of the Struts2 controller

Table 2. Manual test cases of the unstructured music database system

No.	Test_01		No.	Test_02		No.	Test_03	
Name	Functional test of subsystem		Name	Functional test of classification management		Name	Functional test of item management	
Purpose	Function realization of subsystem		Purpose	Function realization of classification management		Purpose	Function realization of item management	
ID	Function list	Test procedures	ID	Function list	Test procedures	ID	Function list	Test procedures
Test_0101	Global search	“music”-->“index”	Test_0201	New database	“database”-->“subclass management”-->“new-built”-->“store”	Test_0301	New items	Click “new-built item” navigation Tab, choose a category without subclasses, fill in items needed to be input, click “store”
Test_0102	Advanced search	“title”-->“bronze drum”-->“musical instruments”-->“indexing”	Test_0202	Database deletion	“database”-->“subclass management”-->“delete”			
Test_0103	Advanced search	“title”-->“bronze drum”-->“musical instruments”--> “random”-->“indexing”	Test_0203	New subclass	Choose a parent class containing subclasses, click “new-built”-->“store” after the display of subclasses	Test_0302	Item input	“input item”-->“input item table”-->“choose file”-->“upload”
Test_0104	Single category query	“article”-->“title”--> “bronze drum”-->“searching”-->“literary theory”-->“title”-->“bronze drum”-->“searching”	Test_0204	Subclass deletion	Choose a parent class containing subclasses and delete those needed to be deleted after the display of subclasses			
Test_0105	Single category query	“article”-->“add search criteria”-->“swf”-->“searching”	Test_0205	New field	Choose a category without subclasses, click “new-built”-->“store” after the display of relevant configuration fields	Test_0303	Attachment table input	“input item”-->“input attachment table”-->“choose file”-->“upload”
Test_0106	Detailed information browsing	Click one of the above searching structures, enter the page for detailed resource information, and click the attachment.	Test_0206	Field deletion	Choose a category without subclasses, click “delete” after the display of relevant configuration fields	Test_0304	Attachment input	“input item”-->“input attachment”-->“choose file”-->“upload”

4 SYSTEM TESTING OF UNSTRUCTURED MUSIC DATABASE AND THE TESTING RESULTS ANALYSIS

The system of the Bo people's music and its music iconography data is composed of a global resource search module, a resource classification management module and a resource item management module. The global resource search module has two major functions, creating Lucene index and realizing the global search. The resource classification management module is mainly responsible for the database and category management and field configuration of the database so as to ensure the system's universality and high extendibility fundamentally. The resource item management module is responsible for metadata input and management of all music resources. It contains three sub-modules, namely new item, item input and item operation. Problems like music iconography data processing [32] and numbering [33] of the system of the Bo people's music and its music iconography data are no longer discussed in this paper. The testing of the database is accomplished in the Internet environment. Music resources, system engineering and database are deployed on the same server. The operating system of the server is CentOS of Linux and the operating system of the client host is Windows7. System accesses are conducted through the browser. Interfaces, links and forms of the system are tested so that applications and researches of unstructured music database can be provided with theoretical guidance.

4.1 *Test content design*

Objectives of functional tests are provided below:

1) According to the UI design, test whether the system interface and the design are consistent and whether the system is operable, reasonable, artistic and coordinated. Tests mainly include navigation test, form test and overall interface test.

2) Test whether all links and functional buttons in each page skip to the target page in line with the design and whether link pages exist or not so as to make sure that all links are valid and correct.

3) Test character strings, numerical values, required items and keywords of the input content so as to test whether parameters of the submitted forms are correct or not and whether the response of the critical value is correct or not.

According to the above test objectives, functional tests can be accomplished with examples of manual tests. Specific testing processes of the portal subsystem, the classification management module and the item management module are provided in Table 2.

Objectives of the performance test are presented below:

1) Provide the statistics of performance parameters, including response speed, handling capacity, and so on.

2) The portal subsystem contacts with users directly. The database is frequently operated due to the large number of users. Therefore, retrieval pages have the largest load in the system. The performance test of the single category query page of the portal subsystem is carried out with JMeter. Specific parameter configuration of the performance test is shown in Table 3.

4.2 *Test results analysis*

Functional test results analysis:

The database system is tested in accordance with the functional test cases, results of which are shown in Table 4.

Table 3. Parameter configuration of JMeter

Concurrent parameter configuration	Thread number/Cycle index of RTS	100/5
	Thread start time	0
Required parameter configuration	Server/Port number	Localhost/8080
	Request mode	/fund/admin/project!list.action
	Path	GET

Table 4. Functional test results of the portal subsystem of the unstructured Bo people's music and its music iconography database system

Test items	Test methods	Test results	Results explanations
Interface test	Interface style	Excellent	Unified styles and colors
	Compatibility	Good	The compatibility of IE6/IE7 is inferior to that of IE8 and higher versions. The layout of the input box is perfect. But there will be disorders in the case of incompatibility.
	User experience	Excellent	The layout is rational. Results of the page are divided in line with functions and locations of operating buttons are unified.
Link test Form test	Artificial operating test cases	Medium	Testing cases and the development process of the system are conducted simultaneously. Bugs are managed by Redmine. Newly discovered bugs are already repaired and testing cases are verified.

Table 5. Aggregated data of the performance test of the portal subsystem

Parameter numbers	Parameter categories	Parameter values
1	Sum of samples	500
2	Average response time of requests	182ms
3	90% response time is smaller than this value	580ms
4	The shortest response time	43ms
5	The longest response time	375ms
6	The percentage of request errors	0
7	Handling capacity	276.90/second

It can be known from Table 4: The test of the database system is in good condition; the system meets the design requirements; interface styles and color unification are excellent; users have good experience; functional logic can be realized correctly; the compatibility is good except that for IE6/IE7.

Performance test results analysis:

Aggregated data of the performance test of the portal subsystem obtained under the JMeter parameters configuration in Table 3 is listed in Table 5.

It can be known from Table 4 that the system is still able to guarantee short time and high handling capacity with heavy loads.

To sum up: the performance of the unstructured Bo people's music and its music iconography database system designed in this paper satisfies the requirements of the database construction as well as concurrent accesses of massive data with extremely excellent user experience.

5 CONCLUSIONS

Based on the explanation of the unstructured data organization and management, this paper analyzes the key technologies of establishing the unstructured Bo people's music and its music iconography database system and carries out functional tests and performance tests on the database system established by the key technologies. Conclusions are drawn as below:

1) The traditional method of file directory tree is not applicable in the management and organization of massive unstructured data. Existing indexing and retrieval tools cannot be fully applied in the management and organization of massive unstructured data. Relevant documents can be searched by users effectively and rapidly through the index established by <keywords, files> provided by BeFS.

2) Dublin Core, Lucene and MVC framework can be regarded as the key to the construction of the unstructured database system.

3) The unstructured database system is composed of a global resource search module, a resource classification management module and a resource item management module.

4) The test of this kind of database system should be divided into two parts, functional test and performance test. Functional tests include three aspects, namely interface style, link test and form test. Performance tests mainly involve response time and handling capacity.

5) The thematic database system of the Bo people's music and its music iconography constructed by key technologies like Dublin Core, Lucene and MVC framework has a relatively high performance. It allows concurrent accesses of massive data and has extremely excellent user experience.

REFERENCES

[1] Wen, L. 2009. XML and unstructured data management, *Computer Knowledge and Technology*, 06(05): 1306-1308.

[2] Teng. T. 2015. Application and future of traditional music big data, *Journal of Xinghai Conservatory of Music*, (138): 61-65.

[3] Wu, H.Y. 2014. An exploration on the database retrieval of northeastern folk music, *Library Research*, (16): 51-54.

[4] Ye, P.S. 2015. A study on search strategies of the unstructured P2P network, *Journal of Yulin University*, 25(2): 37-40.

[5] Yang, Y. 2013. A study on the application of unstructured database in video retrieval, *Science & Technology Information*, 12(9): 73-74.

[6] Zou, B. 2008. Research and Implementation of the Organization of Massive Unstructured Data. *Wuhan: Huazhong University of Science and Technology*.

[7] Lev Novik & Irena Hudis. 2006. Peer-to-Peer Replication in WinFS, *Microsoft Research*, Technical Report, 29(5):4-6.

[8] Craig A. N. Soules, Gregory R. Ganger. 2004. Toward automatic context-based attribute assignment for semantic file systems, *Carnegie Mellon University*, 4(1):12-67.

[9] Wang, C. 2006. A Study on the Implementation of the Database System Retrieval Structure. *Beijing: China Machine Press.*

[10] J. Zobel. & A. Moffat. 2006. Inverted files for text search engines, *ACM Computing Surveys*, 38(2): 346-359.

[11] Udi Manber & Sun, W. 1994. GLIMPSE: a tool to search through entire file systems, Winter USENIX Technical Conference. *USENIX Association*, 4(2): 23-32.

[12] Google Desktop. Http: //desktop.google.com/zh/features.html.

[13] Yahoo Desktop. Http: //desktop.yahoo.com/index.html.

[14] Windows Desktop Search. Http: //www.microsoft.com/windows/desktopsearch/.

[15] Fazli Can, Ismail Sengor Altingovde. & Engin Demir. 2004. Efficiency and effectiveness of query processing in cluster-based retrieval, *Information Systems*, 29(8): 697-717.

[16] E. W. Brown, J. P Callan. & W. B. 2000. Crof. Fast incremental indexing for full-text information retrieval, *20th VLDB Conference*, 34(7):68-90.

[17] Wan, J.C. & Lu, L. 2002. *The Principle, Composition and Application of Software Architecture*. Beijing: Science Press, pp: 122-320.

[18] David K. Gifford, Pierre Jouvelotl, Mark A. Sheldon & James W. O'Toole, Jr. 2006. Semantic file systems, MIT Laboratory for Computer Science, 24(41): 6-29.

[19] David K. Gifford, Pierre Jouvelot, Mark A. Sheldon. & James W. O'Toole Jr. 1991. Semantic file systems. *ACM Symposium on Operating System Principles, Asilomar, Pacific Grove*, CA: ACM Press, 25(5): 13-25.

[20] B. Gopal. & U. Manber. 1999. Integrating content-based access mechanisms with hierarchical file systems, *Symposium on Operating Systems Design and Implementation*, ACM, 27(13): 265-278.

[21] S. Sechrest. &M. McClennen. 1992. Blending hierarchical and attribute-based file naming, *International Conference on Distributed Computing Systems*, 43(22): 572-580.

[22] Weibel S, Kunze J, Lagoze C, et al. 1998. Dublin Core Metadata for Resource Discovery, Internet Engineering Task Force RFC, 2413-2415.

[23] Stuart Weibel. 1997. The Dublin Core: a simple content description model for electronic resources, *Bulletin of the American Society for Information Science and Technology*, 24(1):9-11.

[24] Shigeo Sugimoto, Thomas Baker, Stuart L. Weibel. 2002. Dublin Core: Process and Principles. Digital Libraries: People, Knowledge and Technology, 25-35.

[25] Liu, Y.M. & Liu, S.H. 2000. The Dublin Core Metadata and its application, *Information Science*, 18(6): 572-574.

[26] Zhang, M. & Zhang, X.L. 2000. The development of Metadata and its formats, *Journal of the Sichuan Society for Library Science*, 12(2): 63-70.

[27] Zhou, D.P. & Xie, K.L. 2007. Lucene search engine, *Computer Engineering*, 36(18): 95-96.

[28] Li, W. & Li, L. 2003. Web search engine and the full-text retrieval technique, *Information Science*, 32(5): 558-560.

[29] Shen, Z., Jiang, B.L. & Chen, Y., et al. 2004. A summarization of the full-text retrieval, *Computer Science*, 15(5): 61-64.

[30] Li, Y.C. & Ding, H.F. 2010. Research and application of the full-text retrieval of Lucene, *Computer Technology and Science*, 20(2): 12-15.

[31] Yao, X.W. & Wang, X.M. 2005. The realization of an MVC framework based on the design mode, *Computer Age*, 25(6): 21-22.

[32] Y. Liu. 2014. Cultural dimension of musical iconology based on graph clustering, *Energy Education Science and Technology Part A*, (12): 1863-1870.

[33] Liu, Y.T. 2014. The cultivation of practical consciousness in music iconography study--taking the study on music images of the Bo people's cliff paintings as a case, *Journal of Guizhou University* (Art), 28(1): 117-124.