EPJ Web of Conferences **68**, 00004 (2014)
DOI: 10.1051/epjconf/20146800004

Electronic Document Management Using Inverted Files System

Derwin Suhartono, Erwin Setiawan, and Djon Irwanto

Computer Science Department, Bina Nusantara University, K.H. Syahdan 9, Jakarta, Indonesia

Abstract. The amount of documents increases so fast. Those documents exist not only in a paper based but also in an electronic based. It can be seen from the data sample taken by the SpringerLink publisher in 2010, which showed an increase in the number of digital document collections from 2003 to mid of 2010. Then, how to manage them well becomes an important need. This paper describes a new method in managing documents called as inverted files system. Related with the electronic based document, the inverted files system will closely used in term of its usage to document so that it can be searched over the Internet using the Search Engine. It can improve document search mechanism and document save mechanism.

1 Introduction

Rapid technology development affects on many aspects in many different fields. Fields of information management should be important as it is needed for all people and even for institutions. The issue how to make information well saved is one of the problems. Much information has lost as the way to manage them is not well enough. Document is a physical or digital representation which contains information and is designed as communication tools. Managing physical document is not easy. It is caused by many factors such as its location has to be well defined, human errors such as forgetting where to put it and so on. That is why various forms of documents are encouraged to be moved into digital or electronics documents.

This trend can be seen from the data sample taken by the SpringerLink publisher in 2010, which showed an increase in the number of digital document collections from 2003 to mid of 2010 [1]. SpringerLink itself is an international publisher of journals and e-books. There are about 25,400 journals have been reviewed in the early of 2009, which collectively issued about 1.5 million articles per year. The number of articles published in digital form continues to be increased by 3% per year, in addition to an increasing number of journals are also linearly at 3.5% per year. This fact can be seen in figure 1.

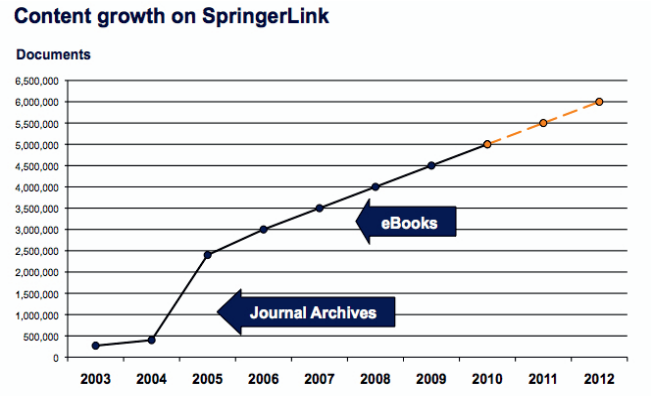

Figure 1. Increased number of digital data documents [1]

By seeing the figure 1 above, the number of digital documents continued to increase over time, resulting in the emergence of the need for a system with the application of technology to a more thorough search for documents based on content contained in the document. Thus, an inverted files system is used to manage the electronic documents.

2 Related Works

Document has a specific syntax and structure which is usually determined by the application or the person who create it. The document also has semantic, which is determined by the document author. In addition, it may also have a representation of the style associated with it, which determines how it should be displayed or printed. A document can also have information about the document itself or called as its metadata. The syntax of

the document can express the structure, presentation style, semantics, or even an external action as depicted in figure 2.

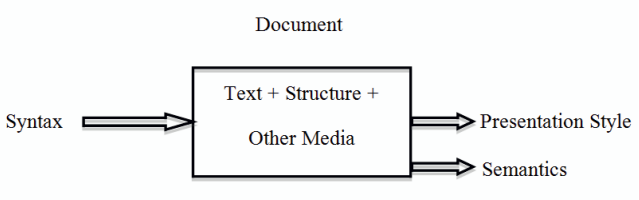

Figure 2. Document Characteristics [2]

Some benefits of digital document:

a. Digital documents can be sent and quick to be transferred
b. It does not require print media such as paper or another
c. It can be transferred to the print media at any time
d. Easily to be indexed, in contrast to the document which uses the print media, several digital documents can be indexed into one to facilitate retrieval of the documents.
e. Can be used in many platforms

By using documents in digital form, it makes possible to distribute a document through web site.

Inverted File is a word-oriented mechanism used for indexing a text collection to speed up the search documents job [2]. Inverted files range from a simple list of any alpha-numeric sequence in a set of documents/pages which are indexed along with the overall documents identification in sequential term, for the list of languages that are more complex, tf/idf weights, and pointers which indicate where a term always appears. The more complete the information in the index, the better the search results.

Examples of inverted file with its weight are described as follows:

Term 1: R1, 0.3; R3, 0.5; R6, 0.8; R7, 0.2; R11, 1
Term 2: R2, 0.7; R3, 0.6; R7, 0.5; R9, 0.5
Term 3: R1, 0.8; R2, 0.4; R9, 0.7

The first line means that the weight of term 1 is 0.3 in Record 1, 0.5 on Record 3, 0.8 on Record 6, 0.2 on Record 7 and 1 on Record 11. Other lines are read by using the same way. Boolean operations with term weight can be explained as follows:

- To query with OR, the higher weight of the record that contains the query term is used as the similarity between the query and the document. The returned list is sorted in similarity of great value to the small.
  Example:
  To query (term 2 OR term 3), we have:
  R1 = 0.8, R2 = 0.7, R3 = 0.6, R7 = 0.5, R9 = 0.7,
  Hence its output sequence will be R1, R2, R9, R3, R7

For queries by using AND, lower weight among the records that match the query.

Document Indexing and Retrieval

Indexing documents is an important step in the retrieval of text information [3]. Through the indexing process, relevant information about a collection of documents are processed and stored in a format that enables easy and quick access during retrieval. Indexing speed up the retrieval process because it would be faster to find a match in the index rather than looking at a whole document.

Indexing is generally done only in 5 main steps, namely:

- Markup & format removal
  Elimination of special format and the entire markup tags in the document
- Tokenization
  Separation of words either from sentences, paragraphs or pages, into a single word token.
- Filtration
  Determination of the term which will be used to represent the document so that it can describe the contents of the document, and to distinguish from other documents in the collection of existing
- Stemming
  Returns every term in the form of the root / foundation of a word
- Weighting
  Assigning weights to the term

Document Retrieval which is generally refers to Information Retrieval on the field of research is a computerized process to produce a list of documents relevant to the request of the Inquirer through a comparison between the requests from the user to the index which is automatically generated from the textual content of existing documents the system [4]. Figure 3 explains the information retrieval process.

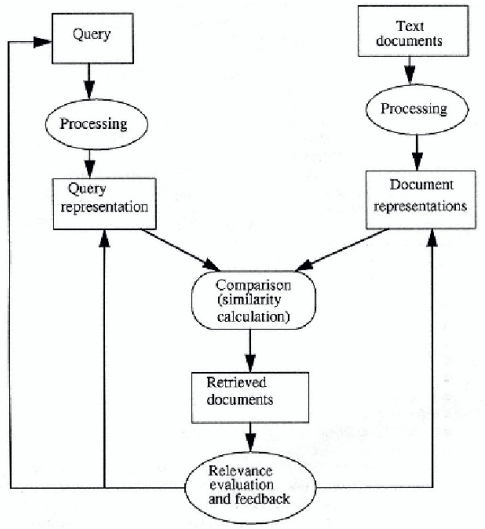

Figure 3. Information Retrieval Process [5]

The basic idea is as follows [2]:

- Given a user query, there is a set of documents containing exactly relevant documents only. Then we refer to this document as the ideal answer set. Because we are given the description of the ideal answer set, we will have no problem in retrieving the document. So, we can think that the query process here is a process specification of properties of an ideal answer set.

3 Inverted Files System

The static pruning of inverted lists, discarding at index construction time entries that are heuristically judged to be unlikely to affect ranking has been proposed. While successful at reducing index size, the method is less effective and less efficient than the dynamic pruning methods [6]. Another ordering variant is described where, within each inverted list, documents are ordered according to the number of times the document is highly ranked by a training query. The savings yielded are not as high as for impact-ordering and large numbers of training queries are required [7]. Many different types of index have been described but the most efficient index structure for text query evaluation is the inverted file [8].

Continuing the previous research, inverted files system for the information retrieval is constructed. Each token has a list of documents containing the token. Documents list is stored in document container formed a double linked list as seen in figure 4. Each document has attributes that are extracted automatically by the system. There are three main models that are used to save object document to the database, which is Token Model, Document Record Model, and Document Model. Token Model is used to model token contained in the document. Document Record Model models the document storage in the form of double-linked list, while Document Model models the document itself.

Searching is conducted by matching the document root of the token to the stored one in the database. If it matches with the root word of the keywords searched, then the whole document object inside the object records document the token is taken to be processed and displayed.

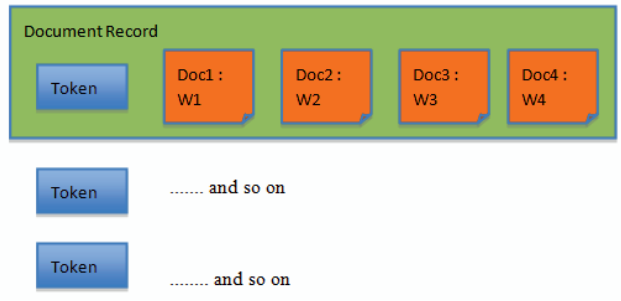

Figure 4. Form of the Inverted Files

This research not only uses an inverted file in the mechanisms, but a system is designed that regulates the inverted files. The regulatory process here includes what will be done while doing insert inverted files, delete inverted files, query execution, log manager, settings, information resources, and indexing.

Because the system on Inverted Files System is too large, we split into several packages in accordance with the process conducted by each package. The relationship between each package is described in the use case description of Inverted Files System as in figure 5.

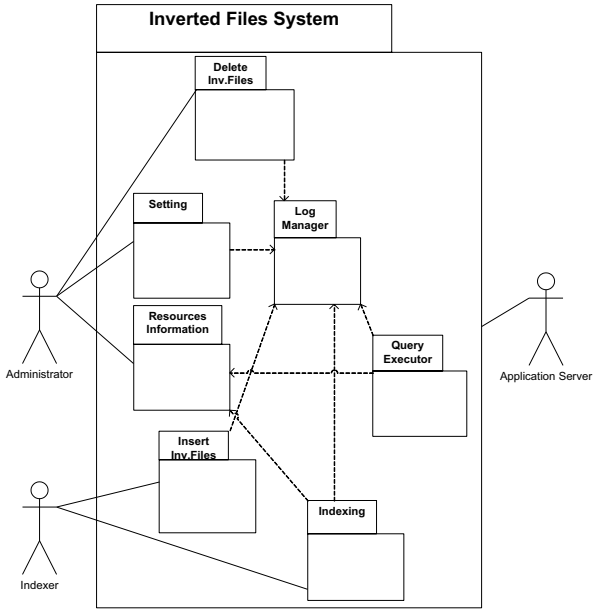

Figure 5. Inverted Files System

1. Delete Inverted Files
   - Admin select the document in which is going to be removed.
   - The system stores the selected document to the list.
   - Admin activates the delete function
   - System do the parsing process of the document list which is going to be removed to the token collection
   - System search for token position of deleted documents in inverted files
   - The system delete the documents that is related to the tokens associated with the deleted documents in inverted files
   - System performs re-weighting to each token

2. Setting
   - Admin sets up the database files on the ideal match the application made
   - Admin determines the IP Application Server.
   - System makes arrangements for shelter HTML document to be stored in database
   - System configures the storage folder location of log files

3. Indexing
   - Indexer does indexing by processing results of the first stage of indexing and calculates the weight of each token.
   - Indexer does database sorting of the inverted file appropriate with the weight for each inverted file.

4. Insert Inverted Files
   - System performs insertion of inverted files which is directly done by saving the document at the end of the linked list.
   - System reads the size setting of the inverted files and metadata. Based on these metadata, system determines the file name of the new database file. Furthermore, the inverted file will be created by system. System updates the metadata that is associated with this auto-extend process.
   - System calculates the size of inverted files that exist in the database. This calculation is done every time the system is about to perform the insertion process to ensure sufficient capacity of database for storing the data.
   - Admin creates inverted files database as a place to accommodate the document

5. Query Executor
   - The system executes query by performing query low level parsing
   - System performs documents retrieval

4 Results and Discussion

By using the application that has been constructed, the performance of our system is tested. Insertion to the document is done in object form (.odb extension). Figure 6 show the detail of insertion process to the inverted files system.

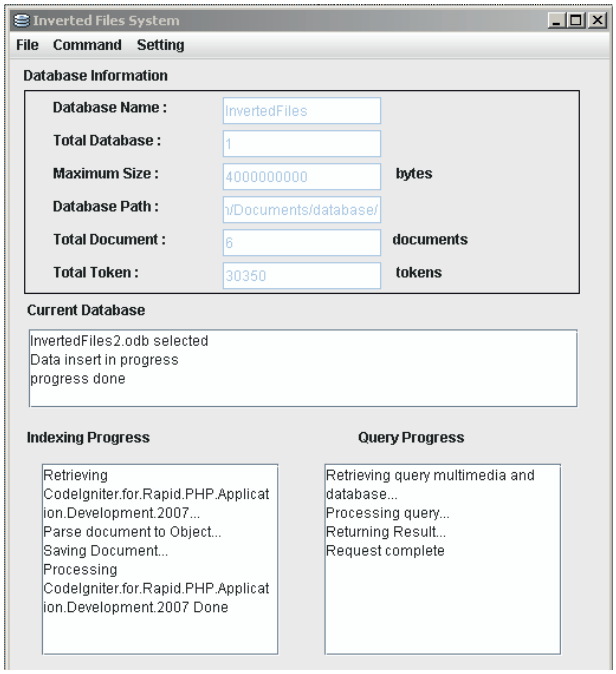

Figure 6. Manage Inverted Files System

The document is then being indexed. From one document form, it will be parsed into a compilation of token. This token is also saved into the database. The process of indexing is depicted in figure 7.

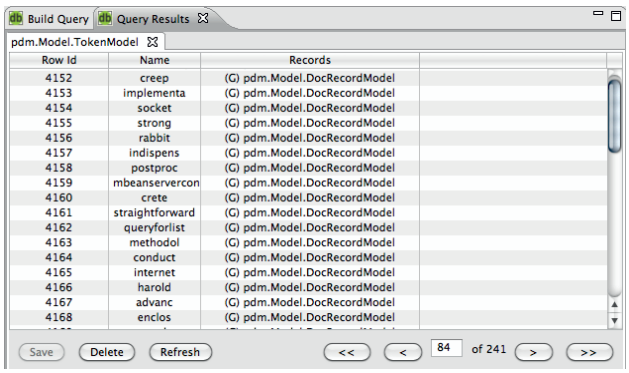

Figure 7. Token Result from Indexing Process

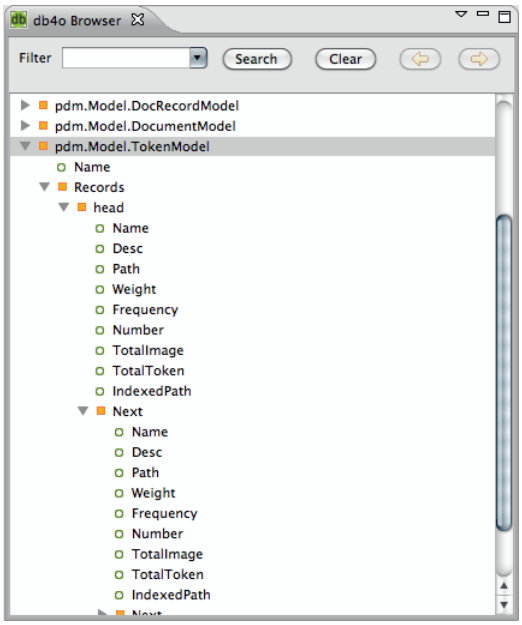

Figure 8. Object Storage Structure on Database

After using search engine, measurement between the relevance of search results need in accordance with the keywords from document which has been entered to the database is calculated.

In this testing, a search engine which has been constructed before is used. It has been implemented in Library and Knowledge Center Bina Nusantara University. The result of the searching process is presented in table 1.

Table 1. Data Set Used for Document Searching

Document Category	Document Qty
software engineering	5
network	2
information retrieval	4
image processing	3
database	2
Others	30

As the data sample, 23 text documents with multiple categories of documents are used. They are software engineering, network, information retrieval, image processing, database and other data. Measurement is performed by using recall and precision. Recall is the ratio of number of relevant documents received by number of relevant documents in the collection, while precision is a ratio of number of relevant documents received by total number of documents received. Table 2 presents the data of recall and precision ratio.

Table 2. Measurement Result of Data Relevancies

Keyword	relevant	irrelevant	recall	precision
project and management	5	0	1	1
networking	1	2	2	0.33
document retrieval	3	0	1.33	1
image	3	2	1	0.6
database	1	4	2	0.2

The test is conducted by using key words relating to the categories of documents sample. It gives a result that the number of relevance document almost the same as the number of documents in the category. It shows us that the document search result is quite relevant to the documents to be searched.

5 Conclusions

Despite of how to save or retrieve, electronic/digital documents need a good database management. By using this inverted files system, it can be concluded that:

1. Inverted files system by using object oriented database is much more effective to be used compared with relational database. It is because the documents are being saved in form of object, not in a table
2. To manage documents in a big quantity, inverted files system is suitable. The important additional management process is by forming an indexing and retrieval mechanism collaborated with inverted files system

References

[1] Moore, W., MacCreery, N., and Marlow, M., *Usage Measurements for Digital Content*, SpringerLink. (2010)
[2] Baeza-Yates, R. and Ribeiro-Neto, B., *Modern Information Retrieval*, Addison Wesley, USA (1999)
[3] Neve G., and Orio, N., *Indexing and Retrieval of Music Documents Through Pattern Analysis and Data Fusion Techniques*, Universitat Pompeu Fabra (2004)
[4] Liddy, E., Document Retrieval, Automatic, Encyclopedia of Language & Linguistics (2005)
[5] Lu, G. *Multimedia Management Database Systems*. Artech House Inc., Boston, London (1999)
[6] Carmel, D., Cohen, D., Fagin, R., Frachi, E., Herscovici, M., Maarek, Y.S., and Soffer, A., *Static Index Pruning for Information Retrieval Systems*. Proceedings of the 24th Annual International Conference on Research and Development in Information Retrieval. New Orleans, LA 43-50. (2001)
[7] Garcia, S., Williams, H. E., and Cannane, A., *Access-Ordered Indexes*. Proceedings of the Australasian Computer Science Conference. Dunedin, New Zealand. V. Estivill-Castro, Ed. Australian Computer Society, 7–14. (2004)
[8] Zobel, J. and Moffat, A., *Inverted Files for Text Search Engines*. ACM Computing Surveys, Vol. 38, No. 2, Article 6. Australia (2006)