WEB SEARCH ENGINE

Liberios VOKOROKOS, Anton BALÁŽ, Branislav MADOŠ
Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice,
Letná 9, 042 00 Košice, E-mail: liberios.vokorokos@tuke.sk, anton.balaz@tuke.sk, branislav.mados@tuke.sk

ABSTRACT

Searching an information is essential part of the Internet today. There are many algorithms and methods for effective web search engines, where the results are more or less relevant. Web pages for better visibility and usability use SEO techniques which improve final query results. The result of this work is full-text web search engine for images and texts. Aim of this work is to make introduction to knowledge of the web search engines and provide own design of web search indexing architecture which will be used for futher educational process.

Keywords: *internet, web search, data processing, indexing, database*

1. INTRODUCTION

Information on the Internet have a different types and nature and their number is constantly increasing. As the increasing quantum of information become enormous, it becomes overwhelmed internet space of different kind of information.

One of the most important characteristics of the Internet is speed, in case, speed of information transmission. The information is not in one place, but spread over the Internet. The hundreds of millions of websites located around the world, whose number is growing every day, it is necessary to find relevant information. Without the search engines it becomes almost impossible [12]. Many websites contain a quantum of information that otherwise could not be indexed without full-text search engines. Each user wants to find information as efficiently as possible and get the most relevant result, so classic catalogs in the world of computers are not enough to search the Internet. Developers are forced to develop more sophisticated and efficient searching tools [6].

Main search engines are proprietary systems that are difficult to use for education process. Aim of this work is to make introduction to knowledge of the web search engines and provide own design of web search indexing architecture which will be used for futher educational explanation.

2. WEB SEARCH ENGINE

Search engines can be generally divided into 4 groups [8]:

1. Full-text search engines - these search engines create their own index of data obtained from the web pages using special software called a robot. The most famous full-text search engines are Google, Yahoo or Bing.
2. Catalog Search engines - the search results does not matter from the text page itself, but the keywords which are basic information about this site - Google, Guntenberg.
3. Hybrid search engines - these web search engines are combining full-text and catalog search engines - Yahoo,Google
4. Meta search engines - these search engines do not crawl the web sites and they do not index data, but they are using more search engines at once - Metacrawler, Dogpile.

3. ARCHITECTURE OF WEB SEARCH ENGINES

The designed system requires an extensive database of information that are maintained and updated automatically by robots. Manually maintained database is mainly used by online catalogs. To be able for the user to receive the best and most relevant information, it is necessary to evaluate the importance of the web pages. For these purposes the assessment tools and algorithms are used. Every major search portal has its own logic of how to assess the weight of words and their own websites [4]. Each web search engine has its own architecture. A general architecture Fig. 1 is dividing the search into two processes:

1. The first process is information gathering:
   - download documents
   - text operations
   - indexing
   - processing of links
2. The second is the search process, which consists of:
   - query formulation
   - query processing
   - return the results to the user interface

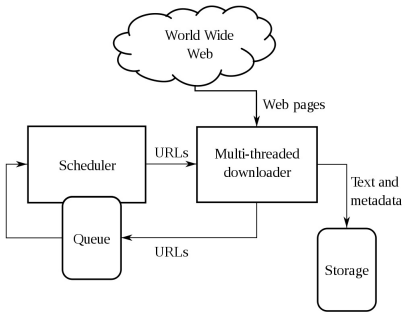

Fig. 1 Architecture of web search engines

3.1. Web Robot

A robot is a computer program for automated activities on the Internet. Its task is by using the hyperlinks on websites to visit all the places on the Internet. After receiving the required information, a robot passes through the next web site. Because this program works in a cycle, after some time it returns back to already searched site to identify any changes. While browsing hypertext addresses a web graph is created. It ensures that the robot looks for only certain hyperlinks and is not returning several times to visited website [7]. A robot can search a web pages in following ways:

- search to the depth - a robot, while browsing a website, uses already founded links as next links to browse.

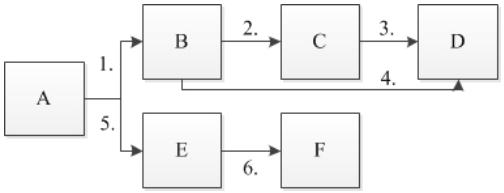

Fig. 2 Search to the depth

- search to the width - a robot stores founded links at the end of a list with web pages

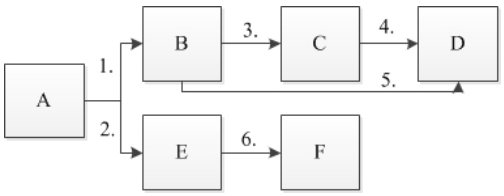

Fig. 3 Search to the width

- random searches - web pages from a list are searched randomly or by other defined matter

3.2. Indexing

Indexing algorithm is designed to assign keywords to data using the text from website. Any word appearing in the index also contains information about which parts of the text Internet site is located. Indexing takes into account the different meanings of the text, update time, size, and the like. To improve the results of searching of the lists, these attributes are modified during indexing [9]:

- size of the characters
- stemming and lemming are used
- stop words are erased

There are 3 formal models of indexing:

- Boolean model is based on set theory and is given a set of terms. This model does not assign any weight to words. It stores only whether a given word is in the text or not, what is indicated by values 1 or 0 Tab.3.2.

Table 1 Boolean model

	Word 1	Word 2	Word 3
Site A	1	1	0
Site B	0	1	0
Site B	1	0	1

- The vector model can be called as algebraic model, because documents and queries are represented as a vector in n-dimensional space. Terms weight in the document is calculated according to their number in the document. The advantage is the search speed, therefore this model is most often used in conjunction with its modifications.
- Probability model is based on probability theory. Bayesian theory is used for the similarity in the documents, what are often used in spam filtering.

3.3. Organize of Search Result

Ordering of search results is important part of web search engine, which by using of evaluation algorithms provides relevant sort of search results. A well known algorithm is the PageRank from Google. This revolutionary algorithm evaluates websites on behalf of back links based on what the importance is calculated. The maximum value is 10, and this value has only a web search engine Google [2].

3.4. Google PageRank

PageRank is Google's algorithm to measure the relevancy of websites. Algorithm assumes that if one website refers to the other, expresses the importance of its. First describe the algorithm and the calculation is to work together Google founders - Larry Page and Sergey Brin at work [2]. The latest version is only slightly modified, but no fundamental change in the calculation:

```
PR (Pi) = (1-d) / N + d.E(PR (Pj) / C (Pj)
```

where :

- `PR (Pi)` - PageRank of the ith site
- `d` = the value (0-1),
- `P1, P2, ..., Pn` are all the pages in the index and therefore N is the number of pages
- E = sum of all the `Pj` of `M (Pi)`, where `M (Pi)` is the set of all pages linking to that page `i`
- `C (Pj)` = number of links to the j-th site

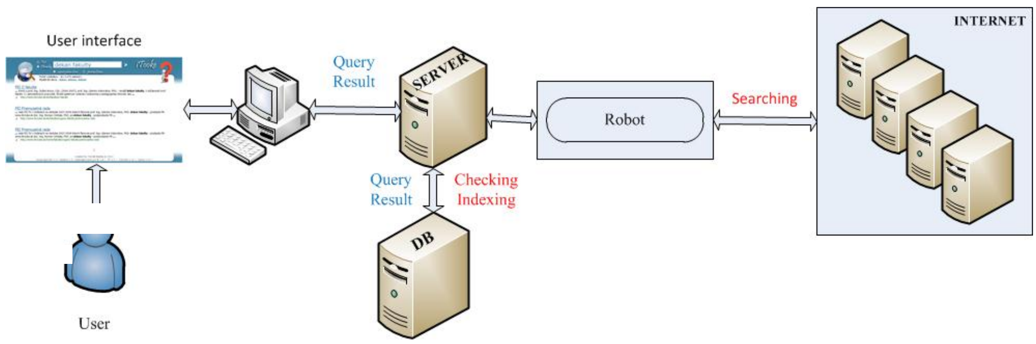

Fig. 4 Designed architecture

4. WEB SEARCH ENGINE IMPLEMENTATION

Designed architecture of the full-text search system is based on standard methods and techniques. It searches the text of the indexed web pages or pictures according to their name. It works with a relational database utilizing MySQL InnoDB kernel. Web search engine is designed to be able to search data across the Internet, or only one page. Therefore, it can be used as a search engine to one specific website. The system consists of 3 parts:

- database
- robot
- user interface

4.1. Database

The database consists of 9 tables Fig. 5 conserving all the necessary data to correct the activities of the entire scanning system. It also contains 4 triggers, which play a role in deleting outdated records.

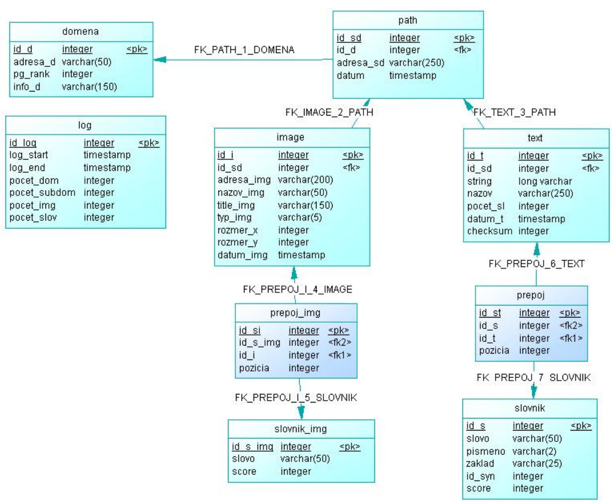

Fig. 5 Database model

Description of the tables:

Table 2 Dababase of designed web search engine

Table	Title
domena	information obtained on the website domain
image	information about founded images
log	records and information about the work of the robot
path	information about the URLs to finded domain
prepoj	prepoj / prepoj-img - keeps indexes of each words to texts or pictures
slovnik	slovnik / slovnik-img dictionary - keep all the words founded in the text or image name
text	text strings from web site

4.2. Robot

The task of the robot is to browse sites within the database in order to obtain new data and references.The robot uses information about domains from the database of the top level domain registries like SK-NIC for sk top level domain. Parser is a part of the robot, in this case represented by SimpleHtmlDom script, which is choosing only the necessary data from each site, which are subsequently checked. If a list exists, it is compared to its timeliness. If the record is other than the one already presented in the database, then updates are provided to the record. Activities of the robot repeats in cycle until they have searched every page of the database or the search in the configuration file was not set otherwise.

Simplified operation of the robot is described in the following steps :

1. during startup, the start of robot's logging activity is initialized
2. number of database domains is discovered and then the next domain is selected.
3. PageRank and domain information are updated.
4. number of addresses referring to the domain is discovered and next domain is selected.

5. parser returns data obtained from a given address.
6. if addresses were parsed, they are checked and updated, followed by a save.
7. if images were parsed, additional information gathering is underway.
8. subsequently, images are checked and saved.
9. if the text was parsed, next steps are conformity checking, saving or updating.
10. update of the number of addresses for that domain.
11. if not all addresses have been checked, point 5 continues, otherwise point 12
12. update of the number of domains.
13. if not all domains have been checked, the process continues in point 3, otherwise point 14
14. completion of logging activities and the robot itself.

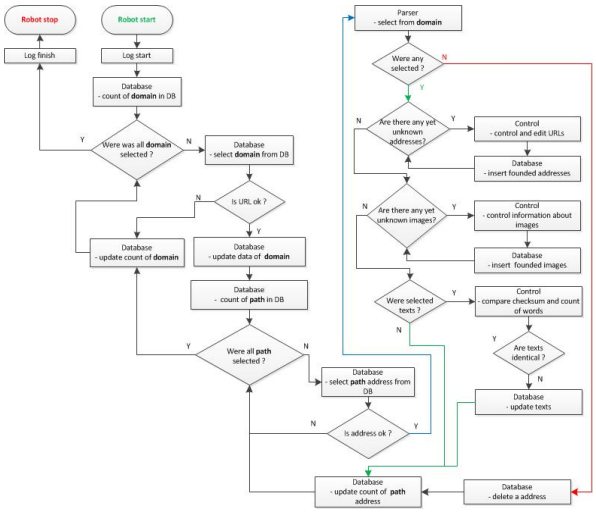

Fig. 6 Web robot flowchart

4.3. Browsing URLs

Designed robot works on the search methods in width. The robot searches the web page and found links saves finally. This means that the newfound addresses the robot gets up after you've checked those are in order before them.

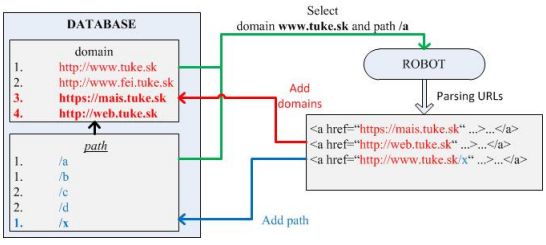

Fig. 7 Browsing URLs

4.4. Indexing Words

The text is divided into individual words, which in turn control the cycle. Control of words involves the removal of those words which contain characters other than alphabets and numbers. Valid character is @, which identifies the word as mail address. If words have passed, put them into the table. Subsequently, the routing tables will create a record of that word in the text so far and at position.

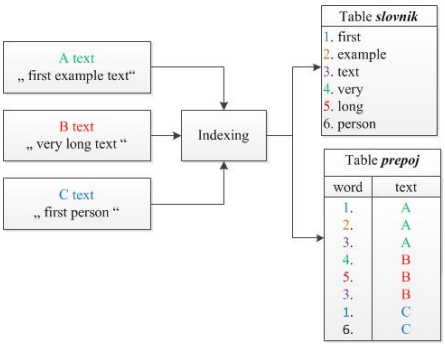

Fig. 8 Words indexing

4.5. User Interface

The user interface is a connection between users and databases. It is designed so that the operation is simple and transparent. The user does not need to manually adjust search options, it is a purpose of the search engine itself, which tries to select and sort the most relevant results Fig. 9.

The user has access to search settings for text or images combined with the possibility of finding at least one keyword or phrase. The search engine is optimized for all commonly used web browsers like Internet Explorer, Opera, Mozilla Firefox and Google Chrome. The search engine can accept and fully operate with words, without or even including special characters. It is not case sensitive. The system includes advanced features to adapt and find similar words with quite the same base word.

Special search function is limitation of the search results for a particular domain. This function is solved using the command `d:` followed by the domain name and then keywords Fig 9.

5. CONCLUSION

The goals of this work was to design a full-text web search engine. System could be further expanded to new functions and features. Providing extensions as lemming and stemming. At the same time complement the options to search for synonyms for specified keywords.

ACKNOWLEDGEMENT

This work was supported by the Slovak Research and Development Agency under the contract No. APVV-0008-10 and VEGA 1/0026/10.

Fig. 9 User interface

REFERENCES

[1] N. ADAM. Single input operators of the df kpi system. *Acta Polytechnica Hungarica*, 7(1):73–86, 2010.

[2] E.G. COFFMAN, Zhen LIU, and Richard R. WEBER. Optimal robot scheduling for web search engines, 1997.

[3] Michael DONOSER, Horst BISCHOF, and Silke WAGNER. Using web search engines to improve text recognition.

[4] Monika R. HENZINGER and Craig SILVERSTEIN Rajeev MOTWANI. Challenges in web search engines, 2002.

[5] ADAM N. DANKOVA E. JAKUBCO, P. Distributed computer emulation: Using opencl framework. pages 333–338, Smolenice, 2011.

[6] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web, 1999.

[7] Arash RAKHSHAN, Arash RAKHSHAN, Lawrence B. HOLDER, and Diane J. COOK. Structural web search engine. In *In FLAIRS Conference*, pages 319–324, 2003.

[8] Fabrizio SEBASTIANI and Consiglio Nazionale Delle RICERCHE. Machine learning in automated text categorization. *ACM Computing Surveys*, 34:1–47, 2002.

[9] J. TALIM, Z. LIU, Ph. NAIN, and E. G. COFFMAN. Controlling the robots of web search engines. In *IN SIGMETRICS/PERFORMANCE, 2001*, pages 236–244, 2001.

[10] Prasannaa THATI, Po hao CHANG, and Gul AGHA. Crawlets : agents for high performance web search engines. In *Proceedings of the 5 th IEEE Conference on Mobile Agents*, LNCS 2240, pp. 119-134, 2001.

[11] Liberios VOKOROKOS. *Digital Computer Principles*. Typotex, Budapest, 2004, p. 232, ISBN 9639548 09 X.

[12] Clement YU and Weiyi MENG. Search engine. In *IN ENCYCLOPEDIA OF DISTRIBUTED COMPUTING*. Kluwer Academic Publishers, 2000.

Received October 4, 2011, accepted December 28, 2011

BIOGRAPHIES

Liberios Vokorokos (prof., Ing., PhD.) was born on 17.11.1966 in Greece. In 1991 he graduated (MSc.) with honours at the Department of Computers and Informatics of the Faculty of Electrical Engineering and Informatics at Technical University in Košice. He defended his PhD. in the field of programming device and systems in 2000; his thesis title was ”Diagnosis of compound systems using the Data Flow applications”. He was appointed professor for Computers Science and Informatics in 2005. Since 1995 he is working as an educationists at the Department of Computers and Informatics. His scientific research is focusing on parallel computers of the Data Flow type. In addition to this, he also investigates the questions related to the diagnostics of complex systems. Currently he is dean of the Faculty of Electrical Engineering and Informatics at the Technical University of Košice. His other professional interests include the membership on the Advisory Committee for Informatization at the faculty and Advisory Board for the Development and Informatization at Technical University of Košice.

Anton Baláž (Ing., PhD.) was born in Sobrance, Slovakia, in 1980. He received the master degree in Informatics in 2004 from Faculty of Electrical Engineering and Informatics, Technical University of Košice. In 2008 he received PhD. in area of computer security. Since 2007 he is working as professor assistant at the Technical University of Košice.

Branislav Madoš (Ing., PhD.) was born on 20.5. 1976 in Trebišov, Slovakia. In 2006 he graduated (MSc.) with distinction at the Department of Computers and Informatics at the Faculty of Electrical Engineering and Informatics of the Technical University of Košice. He defended his PhD. in the field of Computers and computer systems in 2009; his thesis title was ”Specialized architecture of data flow computer”. Since 2010 he is working as a professor assistant at the Department of Computers and Informatics. His scientific research is focusing on the parallel computer architectures and architectures of computers with data driven computational model.