Željko Eremić
Dragica Radosav

Distributed System for Automatic Improvement of Adaptive Website Navigation

Article Info:

Management Information Systems,
Vol. 7 (2012), No. 1,
pp. 025-031

Received 12 December 2011
Accepted 24 February 2012

UDC 007:005]:004

Summary

Communication with clients is now largely conducted through websites, whether it comes to business, scientific, or educational activities. Considering that a dynamic website contains a growing amount of information, the issue of efficient navigation through websites allowing quick access to desired content is imperative. Adaptive websites can customize their navigation based on knowledge gained from previous user behaviour. Discovering such knowledge is a process that engages significant resources, and it is convenient to have it organized by the use of one or more servers. This paper describes the architecture of distributed systems for improving navigation through a greater number of adaptive websites. High flexibility, efficiency and reliability of the performance, as well as saving user's time and effort, can be obtained as a result of this system's usage.

Keywords

adaptive website, navigation, web usage mining

1. Introduction

A variety of commercial, scientific or educational activities are performed using the advantages of the Internet. Designers of dynamic web sites are faced with increasing challenges when it comes to organizing large amounts of information in such a way that includes quick access to relevant information and does not require too much user effort. Designers' vision of optimal navigation may not match the vision of users' optimal navigation. Also, over time, the needs of users change, which is another reason for using web usage mining techniques, in order to discover the knowledge on the basis of previously reported activities of users.

According to Bošnjak (2008) "As in all data analysis problems, data is the most important issue. The correct interpretation of input data in great deal influences the success of the analysis process itself." (p. 14)

According to Bošnjak (2010), "There are three constituents of web mining: content mining, usage mining and structure mining. As most web servers keep logs, the most common data sources are Web access logs (clikcstream data)." (p. 32)

Adaptive websites can enhance navigation based on previous user behaviour models. In practice, the process of discovering knowledge requires considerable resources and, it is desirable to have it relocated on a special server which would be dealing with only this kind of processing. A system containing multiple servers could provide conditions in which such processing would be performed faster and more reliably than the processing performed on a single server. The advantages of the web site's usage, whose improvement of navigation is automated after initial setup, are very clear. The goal is to obtain the architecture of the distributed system for already mentioned optimization, which is flexible, easily scalable, fault and dismissals tolerant, and appearing as a compact unit to customers.

2. Adaptive Websites

Adaptive websites are dynamic websites characterized by the possibility of changing its organization or display based on the model of previous user behaviour. In this paper, changes of navigation are expected based on the analysis of previously recorded users' activity. Log files, as described in Pamnani & Chawan (2010) and Markov & Larose (2007) can be important source of information, if user requests for webpages and files are considered under users' activities. Web usage mining is defined in Srivastava, Cooley, & Deshpande, (2000) as a "process of applying data mining techniques to the discovery of usage patterns from web data" (p. 12). So it is possible to obtain patterns of user behaviour from the contents of log files using web usage mining techniques. Conclusions can be drawn out of these patterns regarding the connection that needs to be established between previously unrelated web pages or documents. Improved navigation in adaptive web sites is most frequently used by setting hyperlinks between web pages where necessary.

3. Previous Studies

One of the earliest works that set creating adaptive web sites as a challenge is presented in Perkowitz & Etzioni (1997). The idea of a new dimension in improving the navigation of adaptive Web sites that takes into account Waypost documents is described in Bathumalai (2008). According to Bathumalai (2008), "Therefore, our work aims to explore this possibility and presents a way to reduce the number of clicks required to find a target document by identifying wayposts, which can act as navigational shortcuts based on frequently travelled users' paths. Wayposts are intermediate documents in a path that could act as a significant guide for users to find the target document." (p. 43-44)

New proposals relating to adaptive web sites are considered in Eremić, Radosav, & Markoski (2010), their contribution to efficient data access in education is discussed in Eremić & Radosav (2011), while the importance of ranking shortcuts leding from a single document to other documents is presented in Eremić (2011a). A proposal of a system architecture for automatic improvement of adaptive website navigation, which is the starting point for this paper, is presented in the article Eremić (2011b).

The architecture, shown in Figure 1, uses a single server that serves multiple client web sites on web servers. Database management system (DBMS) containing one base for each client is placed on that server. Directories where log files of customers are kept waiting to be processed and the location for storing the processing results are placed on the server's hard disk. The control database DB_Common keeps all the configuration data of all clients, as well as information relating to the server activities.

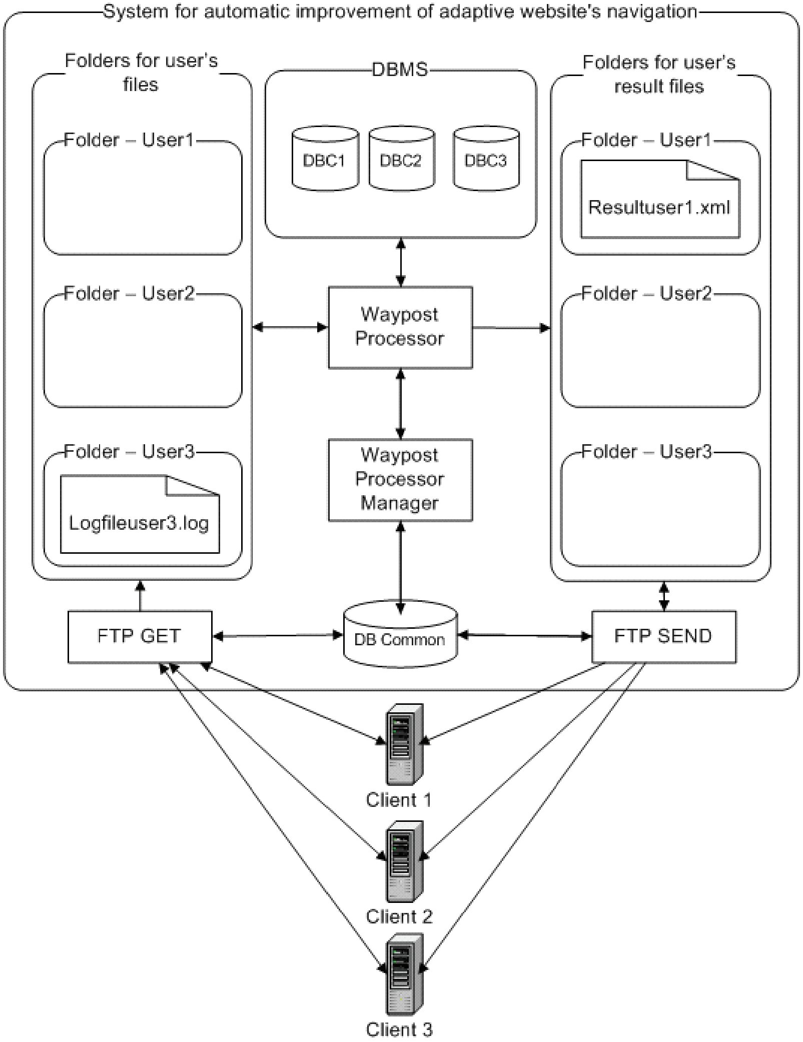

Figure 1 An architecture of a system for automatic improvement of adaptive website navigation (according to Eremić 2011b)

Two processes — "FTP GET" and "FTP SEND" — serve to perform the verification of entrusted tasks in a certain period of time (.eg. every second). When the time comes for downloading log files, the "FTP GET" process, based on the configuration parameters of particular client, downloads log files from a web server of the current client using the FTP protocol. Similarly, when "FTP SEND" determines that the conditions for sending the results to the client are fulfilled, results which were not obtained are sent to the client's web server using FTP protocol. "Waypost Processor" is a process that performs the optimization itself of adaptive website navigation which uses the log files as an input, and gives a list of the proposed shortcuts between documents which are ranked in accordance with the estimated contribution regarding the saving of time and effort of the user as an output. Waypost Processor Manager is a process that periodically checks for log files on pending and, in case there are any, it checks the "Waypost Processor" availability for a new task. If it is available, the task is assigned to the processor and if not, the task is set to be on hold until the "Waypost Processor" is available again for a new task. Usage of multiple servers and multi-core processors which could support multiple processes of the "Waypost Processor" is specified as the direction of further development of this architecture.

4. Distributed Architecture of the System

According to Radosav (2011) "Distributed systems consist of a set of independent computers connected by computer networks and equipped with software support for distributed systems. The software support enables computers to coordinate their activities and share system resources - hardware, software and data. Users of a distributed system should notice only one integrated system, although it may be implemented by using many different computers in different locations." (p. 138)

According to Radosav (2011), there are six basic characteristics that should be ensured in order for the system to be distributed: resource sharing, openness, concurrency, scalability, fault tolerance and transparency.

An architecture of distributed system proposed in this paper is given in Figure 2. The architecture consists of a master server (Server0), while other servers are used for processing, and there can be more than two of them (server1, server2, ...). The adaptive websites whose navigation needs to be improved are hosted on the web servers that are considered for the clients of this distributed system. Number of web servers is not limited, and each of them should allow FTP access and account that enables downloading log files and accepting xml files with the results of improvements in specific folders.

The data which are considered to be data processing inter-results as well as data processing results are stored in the corresponding database tables. What is characteristic of their relationship is that the amount of data used for obtaining the data processing results is much larger than the data representing the data processing outputs. In fact, when the data processing is performed, it is necessary to preserve a small part of data which contains the data processing and key inter-results. A set of necessary data processing inter-results and data processing results is called AWMinDataSet in this article. The existing results can be upgraded when there is a need for a new log file processing using AWMinDataSet and data from log files. In that way, unnecessary data storage is avoided.

Server0 is the main server, which communicates with clients on the one hand, and with the servers for processing on the other. It has a database called DB_Common, which contains the data needed for communication with the client web server (FTP account passwords, directories with log files, folders to store the results...) in the table "ClientSettings". There is a table called "ServerPool", containing information of the tasks intended for the system in this database. Both of these tables are described in more details in Eremić (2011b). There is also a table called ServerSettings, containing details of the servers for data processing in this architecture. "FTP GET" process periodically checks the "ServerPool" table and when it finds that it is time to download log files from a web server, it performs the downloading of the file, places it in the provided folder and records successfully performed task or, if the task hasn't been successfully completed, it tries to do it again in the next time period. Similarly, "FTP SEND" periodically checks for the resulting XML file to be sent, which is specified in the "ServerPool" and placed in the appropriate folder. If the conditions are fulfilled, the resulting XML file which adaptive Web site reads and uses to improve its own navigation is sent to the web server using the FTP protocol. DB_common also contains a set of necessary inter-results and final results for each client (AWMinDataSet). The process called

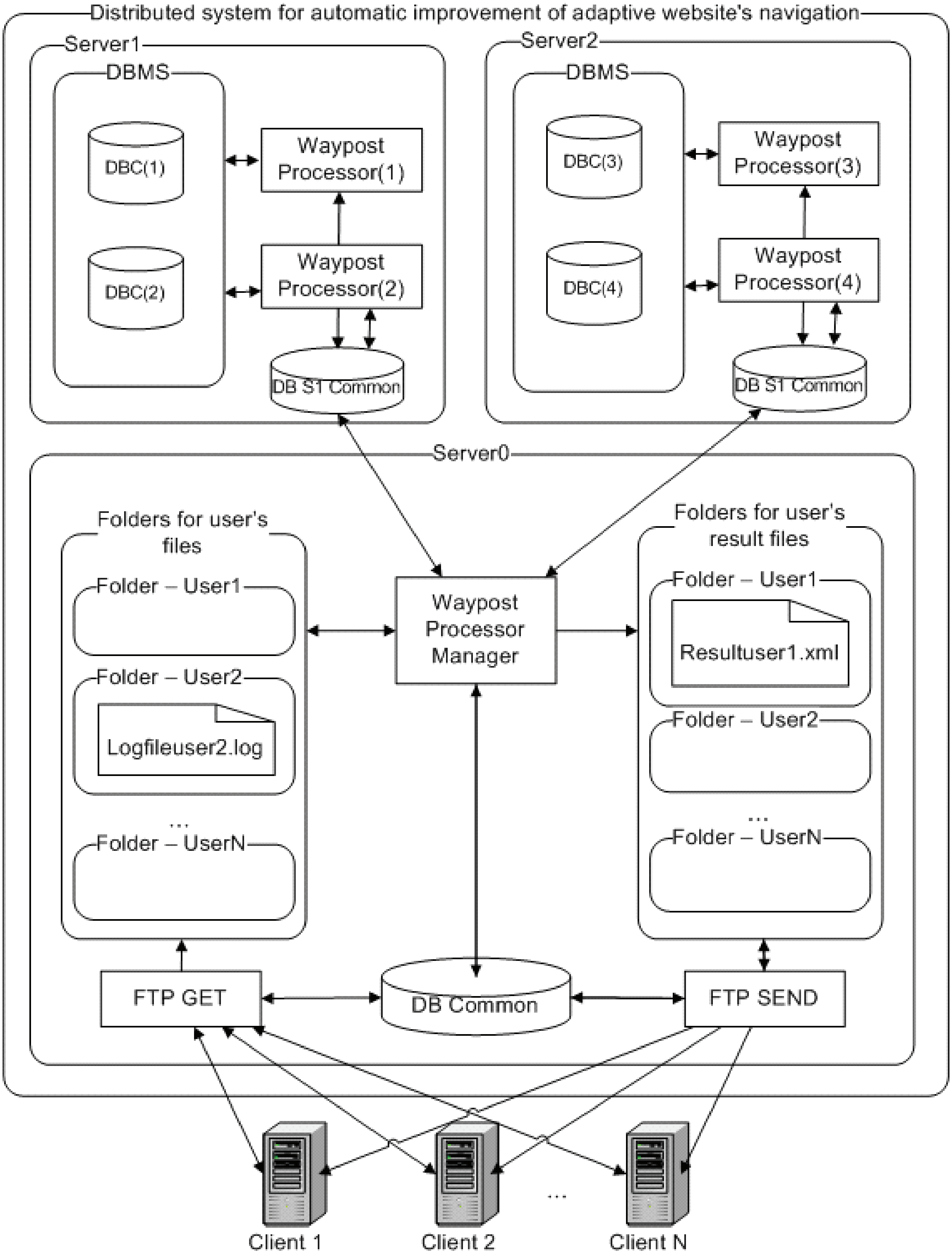

Figure 2 An architecture of distributed system for automatic improvement of adaptive website's navigation

Waypost Processor Manager is used for managing the work of other servers. When the process determines that there are log files in the table DB_Common downloaded from one of the clients and waiting to be processed, it attempts to assign the task to any available server.

Servers used for data processing are computers preferably with multiple cores containing the DBMS (Database Management System). Each process operates with its own database, where the databases are identical in structure but vary in information they contain. A single process and its related databases deals with the optimization of the adaptive website. Database called DB_Sx_Common base (x = 1,2, ...) is used for storing data sent between Waypost Processor Manager process and Server0 , and it is also used for storing processes that are on the server waiting to be processed.

Waypost Processor Manager records the data about capacities of all the servers used for processing (CPU speed, number of cores) and the current availability of all servers in the table "ServerPool". When a new task needs to be assigned, the Waypost Processor Manager selects the least occupied server and assigns the task to it based on the data in the "ServerPool" table, and information about server capabilities (Table ServerSettings). Allocation of the tasks is done in the following way: AWMinDataSet is loaded into DB_Sx_Common along with all the additional data required for the initial processing. The process which is currently available periodically checks DB_Sx_Common database where it finds a new task, takes it, performs and stores the results again in the DB_Sx_Common database. Waypost Processor Manager periodically accesses DB_Sx_Common server's databases and if it finds final processing results, it takes those results and places them in appropriate folder on Server0 in the form of an XML file. Meanwhile, it is recorded in the DB_Common table that the task has been performed, which means that the results are ready to be sent to the client's web server.

The following describes the system in terms of the six characteristics of distributed systems based on Radosav (2011):

- Resource sharing – necessary data for completion of tasks are sent from Server0 to other servers, then the processing results (data) are downloaded from the processing server and sent to Server0. The application for navigation improvement and database structure are identical on all servers used for processing;
- Openness is defined as expanding possibility in different ways. This system can be expanded simply by assigning new servers for processing which need to be registered in the "DB_Common" on Server0, and then to wait for Waypost Processor Manager to assign the tasks when the conditions are fulfilled for it;
- Concurrency – There could be multiple servers for processing, with one or more processors that may have one or more cores. Assigning new tasks is done depending on the resource availability. It is possible that, for example, demanding SQL queries are run using multiple CPUs if they are available and accessible. In this way, the execution time of some demanding tasks is significantly reduced;
- Scalability – There is no limit for adding new resources or changes in existing resources. The addition of a new server for the processing is done by simply connecting the new server to an existing network system and its registration in the database DB_Common on the Server0. Also, if there is a need, for example, for having more powerful processor on an existing server, the only thing left to be done after the hardware change is to register the existence of new capacity in the database DB_Common on Server0;
- Fauilt tolerance - the system continues to operate even if there is a cancellation of some activities or components. If, for example, downloading log files or sending final results fails, new attempts are made. Or, if any of the servers fails, Waypost Processor Manager assigns the tasks to available servers. Server0 can use RAID technologies with multiple disks in order to avoid data loss. It can also use multiple network cards to provide greater security in communication;
- Transparency - the system components are hidden from the end user (adaptive web site on the web server). The client perceives the system as a whole which it communicates with, and has no knowledge of certain parts of the system or any occured error or corrections of errors that appeared within the system.

The architecture of distributed system for improvement of the adaptive website navigation using log files is shown in the figure 2. This architecture shows a simple situation where there are two servers for processing, and each server has a dual-core processor. This architecture can be extended by adding new servers or adding new and more powerful processors.

5. An Example

Illustration of the mentioned system will be provided through the description of a small system that has two processor servers, where each server has one dual-core processor (an example is based on Figure 2). First server has cores coded as 1 and 2 and another has cores coded as 3 and 4. There are three clients over which the optimization of navigation is performed every day by downloading the log files related to the previous day. Looking at "ServerPool" database table on the date of 10th of June 2012 at 19:03:15, it can be seen that all the files that were successfully downloaded on the date

9th of June 2012 were processed (after the process is completed the value 0 was entered in the field called Processing) and successfully sent to client's Web servers (The first three rows). The resulting file, which refers to the first client (row 4), is ready to be sent but it is not sent yet, and FTP SEND process periodically attempts to send it, and after the successful sending it enters the value 1 in the same line in the Send field. Line 5 indicates that the processing of the log file of another client is currently being performed on core 1 of processor 1, which is located on the first server. Line 6 indicates that the processing of log file of the third client is currently being done on the core number 3 of CPU 2 processor (the core with the code 3 is actually the first core of processor on a second server). The last three rows indicate that for the date of 11th of June 2012, log files are scheduled for all three clients. When the log file from the first listed client on the date of 11th of June 2012 is downloaded, the value 1 will be recorded in the line number 7 in the field called Get.

Configuration data, such as name of the base, the FTP user names and passwords, and any necessary parameters related to the client are stored for each customer in the Table 2 - "ClientSettings".

Data about currently active servers, processors and cores are stored in the table 3 - "ServerSettings". If adding or removing new servers needs to be done, it's registered here, so that Waypost Processor Manager is able to asses which server is the most suitable for assigning the new task.

6. Conclusion

A distributed architecture of system for improvement of multiple adaptive website navigations based on the usage of log files has been presented in this paper. The proposed system includes a server which, on the one hand, communicates with the clients and on the other, distributes the tasks to available servers for processing. Servers used for processing may contain different numbers of processors and can be quickly and easily added and removed from the system. This system has all six characteristics that a system should possess to be called distributed according to Radosav (2011). An example of three database tables' content from main database for control of task allocation is listed.

Table 1 Server Pool

Id	Client	Get	Time	Processing	Result	Send
1	Client 1	1	09 Jun 2012 19:01:00	0	1	1
2	Client 2	1	09 Jun 2012 19:02:00	0	1	1
3	Client 3	1	09 Jun 2012 19:03:00	0	1	1
4	Client 1	1	10 Jun 2012 19:01:00	0	1	0
5	Client 2	1	10 Jun 2012 19:02:00	1	0	0
6	Client 3	1	10 Jun 2012 19:03:00	3	0	0
7	Client 1	0	11 Jun 2012 19:01:00	0	0	0
8	Client 2	0	11 Jun 2012 19:02:00	0	0	0
9	Client 3	0	11 Jun 2012 19:03:00	0	0	0

Table 2 Client Settings

Id	Client	DB Name	FTP Username	FTP Password	Params
1	Client 1	DBC(1)	Client 1usr	Client 1pwd	...
2	Client 2	DBC(2)	Client 2usr	Client 2pwd	...
3	Client 3	DBC(3)	Client 3usr	Client 3pwd	...

Table 3 Server Settings

Id	CPU	Server	Speed	Description
1	1	Server1	2.16 GHz	...
2	1	Server1	2.16 GHz	...
3	2	Server2	2.16 GHz	...
4	2	Server2	2.16 GHz	...

The proposed distributed system allows clients to experience the system as a whole, while they don't have to know anything about composition and properties of the system. The system itself is easily scalable, fault -tolerant, resistant to failures of individual components, and has the possibility of optimal use of available components.

References

Bathumalai, G. (2008). *Self adapting websites: mining user access logs*. Unpublished master's thesis, The Robert Gordon University, Aberdeen.

Bošnjak, S., Marić, M., & Bošnjak, Z. (2008). Choosing a Collaborative Filtering Algorithm for e-Commerce. *Management Information Systems , 3* (1), 11-15.

Bošnjak, S., Marić, M., & Bošnjak, Z. (2010). The Role of Web Usage Mining in Web Applications Evaluation. *Management Information Systems , 5* (1), 31-36.

Eremić, Ž., Radosav, D., & Markoski, B. (2010). Mining User Access Logs to Optimize Navigational Structure of Adaptive Web Sites. *11th IEEE International Symposium on Computational Intelligence (CINTI 2010)* (pp. 271-276). Los Alamitos: Institute of Electrical and Electronics Engineers (IEEE).

Eremić, Ž., Radosav, D. (2011). *Adaptive web sites in the function of information access improvement in education.* Paper presented at the Information technology and development of education – ITRO 2011, Zrenjanin.

Eremić, Ž. (2011). *Ranking of shortcuts of adaptive web sites in the function of efficient information access*. Paper presented at Technology, Informatics and Education – for learning and knowledge society – 6th International Symposium, Čačak.

Eremić, Ž. (2011). *Ranking of shortcuts of adaptive web sites in the function of efficient information access*. Paper presented at II Naučno – stručni skup sa međunarodnim učešćem: PREDUZETNIŠTVO, INŽENJERSTVO I MENADŽMENT, Zrenjanin.

Markov, Z., & Larose, D. (2007). *Data Mining the Web: Uncovering Patterns in Web Content, Structure, and Usage.* New Jersey: John Wiley & Sons.

Pamnani, R., & Chawan, P. (2010). *Web Usage Mining: A Research Area In Web Mining.* Retrieved December 20, 2011 from RIMT: http://www.rimtengg.com/iscet/proceedings/pdfs/database/73.pdf

Perkowitz, M., & Etzioni, O. (1997). Adaptive Web Sites: an AI Challenge. *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence* (pp. 16-21). San Francisco: Morgan Kaufmann Publishers.

Radosav, D. (2011). *Software Engineering* (2nd ed.). Zrenjanin: Technical Faculty Mihajlo Pupin.

Srivastava, J., Cooley, R., & Deshpande, M. (2000). Web usage mining: discovery and applications of usage patterns from web data. *SIGKDD Explorations , 1* (2), 12-23.

Željko Eremić

Technical College of Applied Sciences in Zrenjanin
Đ. Stratimirovića 23
23 000 Zrenjanin
Serbia
Email: zeljko.eremic@gmail.com

Dragica Radosav
University of Novi Sad
Mihajlo Pupin Technical Faculty
Đ. Đakovića bb
23 000 Zrenjanin
Serbia
Email: radosav@tfzr.uns.ac.rs