DATABASE FOR REGISTRATION OF DNS AND DHCP RECORDS

Luděk Caha
Doctoral Degree Programme (3), FEEC BUT
E-mail: xcahal00@stud.feec.vutbr.cz

Supervised by: Tomáš Macho
E-mail: macho@feec.vutbr.cz

ABSTRACT

This article deals with design and implementation of a database system for registration of DNS and DHCP records. The records are classically edited as texts files. But this method is somewhat complicated and chaotic for use in wide computer networks.

The system is designed to control the correctness of entered values and prevent double counting in records. Another function of the system is the assignment of the access right for administrators. All administrators are allowed to access only the assigned part of the records. On the basis of the database content the system generates data files for DNS and DHCP servers.

The system is controlled over the web interface, which is provided by the Apache server. The Apache server must include support for the PHP scripting language. Data, which the system governed, are stored in MySQL database server.

1. INTRODUCTION

When operating extensive computer networks, a large amount of domain names and DHCP records must be administered. The records are usually saved in texts files, which are stored in DNS and DHCP servers. For a larger amount of records this way is chaotic and inconvenient. The designed database system is therefore used to simplify the administration of the records.

Among the main functions of the system is generating of text files for DNS and DHCP servers on the basis of the data entered in the system. When the data are entered, the system checks the data content and handles collisions, which may arise. Only authorized administrators are allowed access into the system, and can edit records in the assigned parts of the network only.

2. SERVER

One of the fundamental requirements for the designed system was accessibility over the web interface. This function is provided by the Apache HTTP web server http://httpd.apache.org. As the Apache server alone can provide operation of static web pages only, therefore the PHP module http://www.php.net was included into the Apache

server. This included module allows processing of the PHP scripting language, which provides required functionality of the system. As processing of scripts in the PHP language is performed on the side of the server, the scripts become independent on the web browser used by a client. Finally both programs are advantageously distributed under Open Source license.

The data are stored in the MySQL database system http://www.mysql.com. For most applications this system is distributed free of charge. The company MySQL AB sets the price according to the rule "if you distribute the software free of charge, we will do so as well, if you sell the software, we will sell it to you, too". Another advantage of the MySQL database system is that is can easily co-operate with the PHP scripting language. Communication between the MySQL and the PHP is implemented using the SQL language (Structured Query Language). The SQL language is considered to be the 4GL (Fourth Generation Language). That is because it is not algorithmic as the majority of other programs. It only says what is to be done, but not how to do it.

Fig. 1 shows the block diagram of the database system, with links between the individual parts of the system.

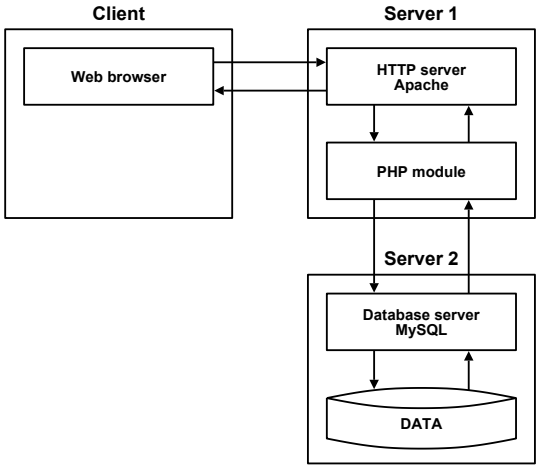

Figure 1: Processing of the dynamic web pages using the database.

3. THE DESIGN OF THE SYSTEM

The design of the database system consists of several parts. The more important parts are the design of the database model and the design of the program structure. These problems are dealt with in the following chapters of this article.

3.1. THE DATABASE MODEL

The database model is the basis of the whole system. The model defines the way, in which the data are stored in the database and how simple it will be to use them. The final database model is shown in Fig. 2.

The first important table in the database model is the table **Spravci**, which is used for login into the system. Another important table is **Segmenty**. In this table, the first login logged administrator can create a segment of the network. To this segments can be further assigned computers which are placed in table **Pocitace**. The administrator can create new administrators with lower authorization and assign them some segments. These new administrators will be allowed to edit only a computer in the assigned segments. Assignment

of authorization can be done with the table **Spravci_segmentu**. The administrators with the "root" authorization can edit all the segments in the system automatically.

Another table in the database is the table **Zony**. This table serves for definition of DNS zones. Then there is the table **DHCP,** which contains a list of the DHCP servers. For these servers lists of the computers are generated. The table **DHCP** contains only names of servers. The parameters of the servers are placed in tables **DHCP_CMU** and **DHCP_ISC**. The last table in the database model is the table **Texty**. This table is used to store the text, which is shown on the introductory page of the system. For example, its may be various notices on the system operation.

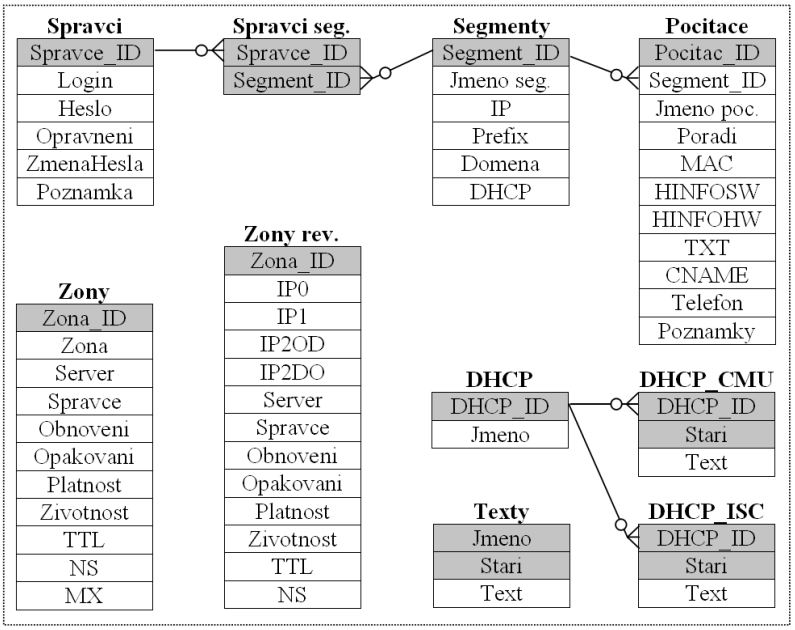

Figure 2: Database model of the system.
Note: The primary keys of all tables are marked with grey colour

3.2. PROGRAM

The design of the program structure is the most complicated part of the system. The main problem constitutes editing of a large number of similar tables and at the same time simple modification and expansion of the program. Those two requests are guaranteed only when all parts of the program are unique. This access on the other side implicates increased complexity of the program code.

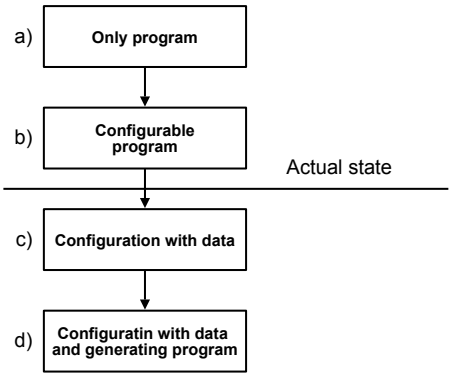

Figure 3: Development of program structure

In the course of the development of the system several variants of the program have been tested. Fig. 3 gives an overview of the possible solutions for program building. Point a) represents the approach which can be used for simple systems only. On the other hand, point d) applies to very complicated systems. The designed system can be found midway between these two limit points. More detailed explanation can be found in [1].

3.3. SECURITY OF THE SYSTEM

The implemented system contains a number of elements denying access to unauthorized administrators. One of these elements is authorization control in processing of SQL query, which protects the system from hackers who may attempt to send non-standard requests. Another protection element is denies access to some pages of the system to administrators with insufficient rights. Also it denies access to some directories on the server. To provide complete security it is necessary to protect the server by encryption to prevent wiretap of a password.

4. USE OF THE SYSTEM

The complete system and its varieties cannot be described within the scope of this article. Therefore, several functions have been chosen. A complete description can be found in [1].

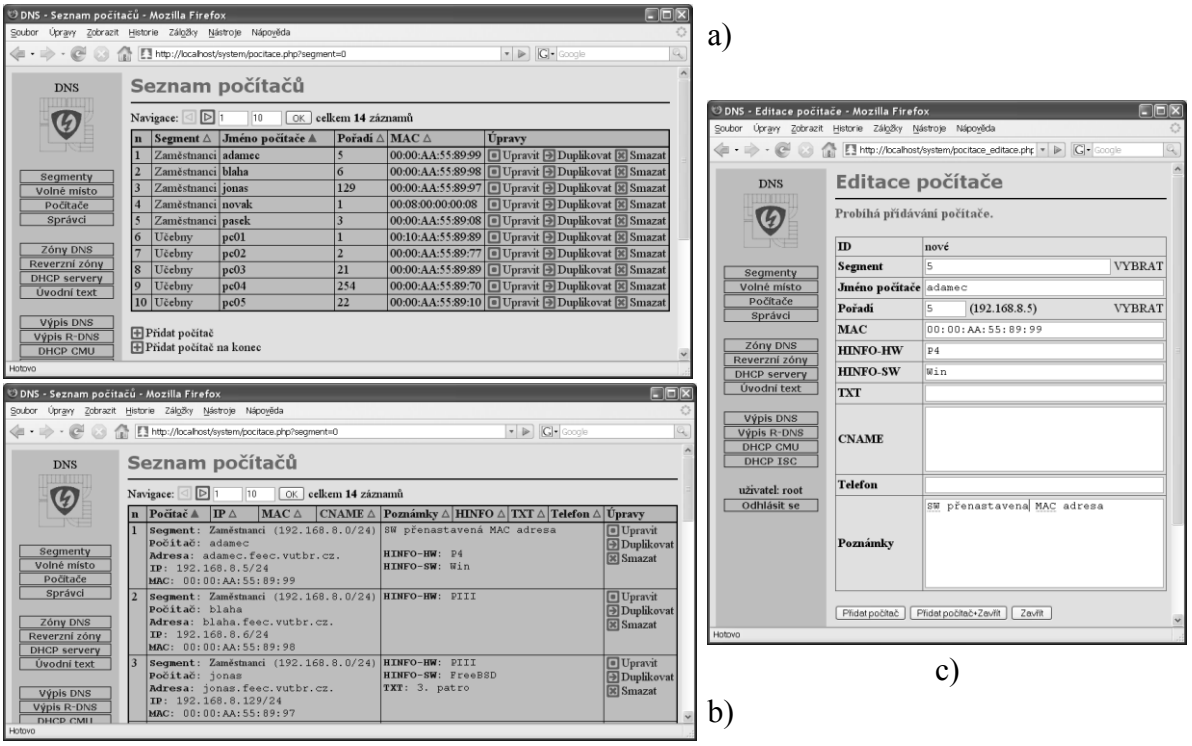

Figure 1: a) Illustration table with a list of computers. b) Extended table with a list of computers. c) Example of a window for editing records.

After the administrator logs in the system, a menu appears in left part of window, which provides access to individual functions of the system. First of all, it is access to the tables segments, computers and administrators. The number of shown options is also derived from administrator's authorization. For illustration, the table with the list of computers, which is shown in Fig. 4 a) has been chosen. Except the basic allow some tables allow switching to alternative representation. An example of alternative representation is shown

in Fig. 4 b). Above all tables, control elements are shown. These elements allow sorting of the records and navigation in the tables. In the right side of the tables there are links for record modification. An example of modification and adding of records is in Fig. 4 c).

Besides the above described tables the first group of the menu also includes a link to detector of free space. This detector finds and shows, in the defined region, a part in which no segments has not been created yet.

Another group of links in the menu serves for definition of direct and reverse DNS zones and also for definition of a list of DHCP servers, which can distribute addresses to the computers in the network. The last group in the menu serves for generating of data files for the above defined servers.

5. CONCLUSIONS

The designed and implemented database system allows creation of a list of computers in a network. On the basis of this list data files for the DNS and DHCP servers can be generated. Further the system allows management of a list of administrators. Each administrator can get authorization for a part of network, or for the whole system. The administrator with authorization for the whole system can edit segments in the network. To these segments he can add new computers with a lot of descriptive information. Finally, he can add new administrators and change their authorization and edit direct and reverse DNS zones and settings for the DHCP servers. Administrators with low rights can edit only the list of computers in the assigned part of the network.

More information about the system, with suggestions for its extending and improving is given in [1].

REFERENCES

[1] Caha, L.: Databázový systém pro evidenci DNS a DHCP záznamů. Brno: Brno University of Technology, Faculty of Electrical Engineering and Communication, 2007. 86p. Diploma thesis, Supervisor: Ing. Tomáš Macho, Ph.D.

[2] Welling, L., Thomson, L.: MySQL Průvodce základy databázového systému. Brno: CP Books, 2005. 256 p. ISBN 80-251-0671-3

[3] Dostálek, L., Kabelová, A.: Velký průvodce protokoly TCP/IP a systémem DNS. Praha: Computer Press, 2000. 436 p. ISBN 80-7226-323-4

[4] Droms, R., Lemon, T.: DHCP Příručka administrátora. Brno: Computer Press, 2004. 490 p. ISBN 80-251-0130-4

[5] Kosek, J.: Aplikace na Webu - Jak pracují databáze na Webu. [online]. 1999. [cit 2006-03-07]. Available on: http://www.kosek.cz/clanky/iweb/index.html

[6] Janovský, D.: Jak psát web. [online]. 2006. [cit 2006-04-28]. Available on: http://www.jakpsatweb.cz

[7] Hojtsy, G.: PHP Manual. [online]. [cit 2006-12-21]. Available on: http://www.php.net/download-docs.php

[8] MySQL AB: MySQL 5.0 Reference Manual. [online]. [cit 2006-12-21]. Available on: http://dev.mysql.com/doc/