MODELS IN DESIGNING DOCUMENT MANAGEMENT SYSTEMS

Kremena Arsova – Borisova[1], Veselin Christov[2]

[1] University of mining and geology "St. Ivan Rilski", 1700 Sofia, kremena.arsova@mgu.bg
[2] University of mining and geology "St. Ivan Rilski", 1700 Sofia, veso@mgu.bg

ABSTRACT. In all organizations, they use and accumulate large quantities of various documents - contracts, annexes, minutes, invoices, bank statements, mission statements and reports, advance reports, offers, customs declarations, bills of lading, references, proposals, orders, requests, applications, drawings, sketches, plans, decisions, notary deeds and others. Documents can be both paper and electronic. Typically, document management systems are used to handle a large quantity of documents. The article examines the different models in the design of these systems: information, functional, structural, the database system, etc. An experimental document management system is presented to test the models under consideration.

Keywords: document management system, DMS, design model

МОДЕЛИ ПРИ КОНСТРУИРАНЕ НА СИСТЕМИ ЗА УПРАВЛЕНИЕ НА ДОКУМЕНТИ

Кремена Арсова – Борисова[1], Веселин Христов[2]
[1] Минно геоложки университет „Св. Иван Рилски", 1700 София, kremena.arsova@mgu.bg
[2] Минно геоложки университет „Св. Иван Рилски", 1700 София, veso@mgu.bg

РЕЗЮМЕ. Във всички организации си използват и натрупват големи количества от разнообразни документи – договори, анекси, протоколи, фактури, банкови извлечения, командировъчни заповеди и отчети, авансови отчети, оферти, митнически декларации, товарителници, справки, предложения, заповеди, молби, заявления, чертежи, скици, планове, решения, нотариални актове и други. Документите може да са както на хартиен, така и на електронен носител. Обикновено при работа с големи количества документи се използват системи за управление на документи. В статията се разглеждат различните модели при конструирането на тези системи: информационен, функционален, структурен, на системата от бази от данни и др. Представена е експериментална система за управление на документи, при която се изпробват разгледаните модели.

Ключови думи: система за управление на документи, СУД, конструктивен модел

Introduction

Each organization has a real need for a document management system (DMS). DMS exist to organize, store and retrieve information accurately and effectively. They are designed to organize files and recordings in electronic form. They are a mean of providing more compact storage, universal access for retrieval, a higher level of data security and privacy. Its base features include [Björk, Bo-Christer, 2001]: acquisition, storage, classification, indexing, version editing, maintenance, use, document security. When documents are in a structured system, more complex processing procedures, such as tracking the life cycle of each document, are possible. They are designed to provide the data warehouse where documents are created, managed and stored for easier access by different users and departments within an organization [Löwnertz, K., 1998].

When designing such a system, it is necessary to plan carefully the various models that define its design and operation so that it is as efficient, easy and safe as possible. [Ralph H. Sprague, Jr, 1995].

An experimental document management system has been designed and developed, in which such design models have been implemented.

Design Models in DMS

In the process of planning and building a DMS it is necessary to consider and clarify a number of issues related to its functioning. The solutions found determine the design models embedded in the future EMS. Properly selected models should provide system evolution and scalability. Such models are:

System architecture. It determines whether the system will function as a separate local application for a single user or will serve a team of users.

The operating system file system is generally used as a local DMS, but in this case, the structuring of document storage and searching does not particularly support users. There is no further description of the documents, they cannot be grouped by additional signs, the life cycle and the movement of the documents cannot be traced. In some word-processing systems, for example, Microsoft Word has built-in local DMS components, but it is limited to documents created with this

word-processing system. In general, local DMSs are rarely created and used.

Collective DMSs can be organized in two major ways: distributed processing and client-server technology.

In distributed processing, a single user's work is stored on his computer and a system for searching, accessing, and moving documents between users is organized. This way of working is closer to the thinking of traditional users. On the other hand, it is very complex for implementation and maintenance, and creates a number of problems with the security of storage and authorized access to documents. The number of users is limited.

Client-server technology is a network architecture type. The administration is centralized. The server is a computer running dedicated server software and a network operating system that makes its resources available to other computers on the network. On the other hand, the client is a computer that accesses the server resources. The term client may also mean software programs accessing server programs. Access to resources is controlled through authentication and access rights. User accounts are created and stored on the server. The client is the active partner in the communication, and the server is a passive partner, waiting for requests, processing and returning a response. Each server can receive queries from a large number of clients simultaneously. The connection between the client and the server is accomplished through a specific protocol (a set of rules of the dialogue between the two partners).

Most DMSs are based on this architecture. Various variations are possible: the databases (DBs) for the documents are on the server and the documents themselves are on the client computers or the documents and the DBs are on the server. It is possible that most of the DMS is in the form of a client module of the user computers. This is a good solution for a document system limited within an organization. The disadvantage is that access to the DMS can only be done by computers where this module is installed. If access is needed from anywhere, then the solution is a Web based DMS, where it is all deployed on a server and accessed via a web browser.

Functional model. It describes the functions that the DMS implements. Depending on what documents the system is intended for, it may be specialized or general purpose. For example, a specialized DMS is the one that organizes the design documentation of a building organization. Such systems have some specific functions related to special operations with documentation such as selection of tender documents. The main functions of the general purpose DMSs include acquisition, storage, classification, indexing, version editing, searching, using, printing, sending to specific destinations, electronic signature stamping, security, document lifecycle; creation and maintenance of user accounts and user groups, user profiles, specific user access rights, personalized user interface, etc. A separate aspect of the functional model is the administration of the system.

Structural model. It defines the modules and subroutines of the software and DMS hardware requirements. This model is in direct connection with the chosen architecture and functional model. It also depends on the database system, the security model, the user and the information models of the system. In server DMSs, the technical characteristics and positioning of the server or servers, their connection to the computer network, the operating systems, the organization of the data warehouse, etc. are of particular importance. The choice of the programming language and database management system (DBMS) depend on the architecture, the structural and the functional model.

Structure of databases. Mostly, relational databases are used in the DMS. It is important to properly build the individual bases and their system organization: there must be no data excesses, the system has to be properly normalized and it is possibly to add new bases. It is necessary to maintain the appropriate data for the documents and the users and for who and when they used the documents.

To these design models can be added:

- An **information model** describing the movement of documents and user data inside the system;
- A **security model** covering all issues related to secure document storage, recovery from user errors and system crash and malfunctions, user accounts and profiles, organizing users in groups, access rights, user tracking, etc.;
- An **user model** of the system that defines the user interface, user personalizing, organizing collaborative work on the same document, etc.

Experimental DMS description

The Experimental DMS is implemented as a desktop application through a client - server architecture. Resources from Microsoft are selected to the realization. The Visual Basic programming language for Visual Studio 2010 and the .NET Framework 4.0 platform are used to implement the client part. For the implementation of the server side of the application, MS SQL Server 2014 Express and Transact-SQL are used. The Inno Setup compiler (installation software written on Delphi) is used to create the application's installation system.

For system implementation about the server 6 tables, 11 procedures, 4 functions and 326 rows of program code are made. With respect to the client part - 7 forms and 1490 rows of program code are produced. The main activities of the system are:

- Add a document to the system and allocate it to a specific directory - To add a document to the system, the user first describes the document by adding additional information about it - chooses its type; the directory in which it will be recorded; comments and labels, then indicates the required document and adds it;

- Use of function keyboard keys - The user is given the opportunity unless provided by the interface controls to use relevant function keys: INSERT function key - pressing its trigger event “click” on the Add button; Function key F2 - Pressing the button activates the event “click” on the Edit button; with the DELL function key - when it clicks, the event triggers the click of the Delete button; Function key ENTER - when pressed (in a text box) this triggers the radio button click event (if there is one above the text box);

- Searching and retrieving electronic documents (records) - In the submenu “Entering and correcting” of the menu “Data entry”, in the tab “Main data”, the user has the possibility to search and retrieve records according to the following criteria: Code - search is performed in whole or in part in the text field code; Document type - searching by document type selected from combo box; Folder - searching by folder name selected by the combo box; Document name - searches are made in whole or in part; Comment - search is made on a wholly or partially entered comment; Labels - searching based on user-tagged labels;

- Sorting - The system allows sorting of data;

- Generate sequential document type numbers - The system automatically generates a sequential number and adds it to the code corresponding to each document type. For example, the user adds a document type Archive (the code is ARH), and if no documents of this type are added to the system, then its ID will be ARH1. If there are already 10 documents of this type in the system, the newly added will be code ARH11;

- Link between documents - The system allows the user to add additional documents to one master document. For example, we have a document that contains a contract and several other documents that are annexes to this contract;

- Revisions (versions) - Allows you to store and manage various document revisions. This feature is useful for documents that require frequent updates. When adding a document, the system checks by its name if it exists in the system and if not, then by default add rev0 to its name. If a document with the same name exists when it is added to the system, its Rev + revision number for this document is added to its name;

- Opening a document - In “Correction and Input” submenu the system allows the user to open a marked from him document by clicking on “Path”;

- Show the size of the document and its file extension - The system provides information about the size and file extension of each document;

- System Administration and Security - Different levels of access may be required for different users (when correcting and deleting documents). The system allows a user to be restricted in their access to correct and delete data. In addition, a user with a lower access level cannot correct or delete a document added by a higher-level user. Access levels are set in menu “Users”;

- Remembering the last user who used the program - The system keeps information about the last user who used it and when it was rebooted, In this case it is not necessary to fill in the User field, but only its password.

Figure 1 shows a schematic diagram of the structural model of the experimental system client part, and Figure 2 illustrates the scheme of the database system.

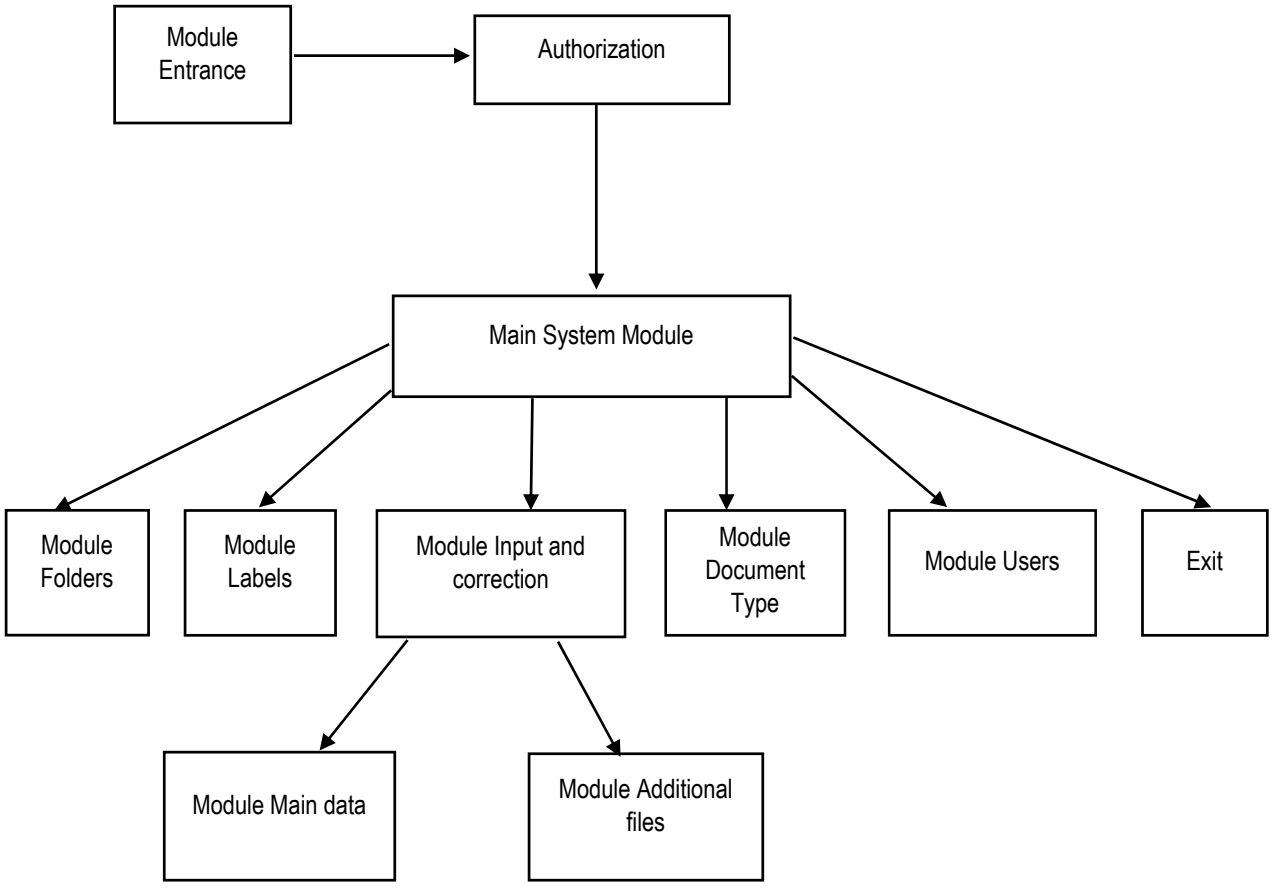

Fig. 1. Structural model of the experimental system client part

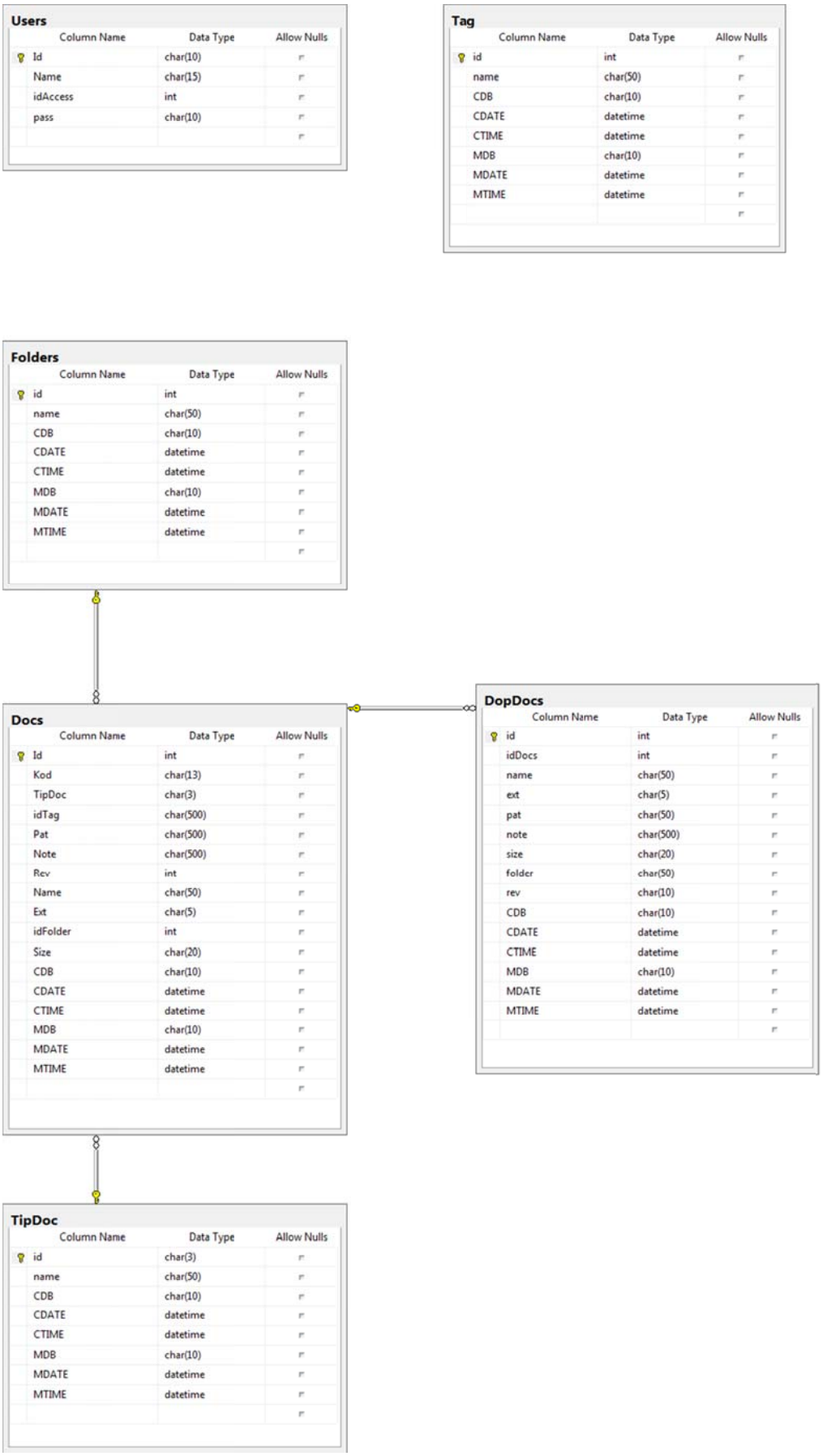

Fig. 2. Database System of the experimental system

Conclusion

Document management systems are an important part of the software of any organization that works with documents. Their quality largely determines the effective work of these organizations. Planning and developing such a system requires careful consideration and selection of the various design models that will be built into it. Such models are system architecture, functional model, structural model, database system, information model, system security model, user model, etc.

To illustrate this model approach, an initial version of an experimental document management system is being developed. It could be evaluated by:

- adding a metadata classification function;
- adding a feature that will prevent specific users from accessing certain menus and features;
- adding a search function using metadata;
- adding a keyword search feature;
- adding a semantic analysis function to determine a search connection;
- adding web functionality so that it can be accessed via mobile phones, tablets, etc.;
- giving an ability to upload multiple documents simultaneously.

References

Björk, Bo-Christer, 2001. Document management - a key technology for the construction industry, Information and Communications Technology (ICT) in the Practice of Building and Civil Engineering, proceedings of the 2nd worldwide ECCE Symposium, Espoo, Finland 6-8.6.2001, Association of Finnish Civil Engineers, Helsinki, Finland, pp. 35-43;

Löwnertz, K., 1998. Change and Exchange – Electronic Document Management in Building Design, Licentiate Thesis, Dept. of Construction Management and Organization, Royal Institute of Technology, Stockholm, Sweden;

Ralph H., Sprague, Jr., 1995. Electronic Document Management: Challenges and Opportunities for Information Systems Managers, MIS Quarterly, Vol. 19, No.1 (Mar.,1995), Management Information Systems Research Center, University of Minnesota.