

F. H. Pashayev, D. Sc. in Engineering, Head Researcher, e-mail: pasha.farhad@gmail.com,
Institute of Control Systems of the Azerbaijan National Academy of Sciences,

S. E. Pashayeva, Teacher, e-mail: pasayevasevinc5@gmail.com,
Nakhchivan State University,

H. T. Najafov, Tutor, e-mail: hasan_nacafov@mail.ru,
"Nakhchivan" University,

B. A. Suleymanli, Dissertant, e-mail: bahram.suleymanli@gmail.com
Institute of Control Systems of the Azerbaijan National Academy of Sciences

Operating Algorithms for Folders and Files

The paper presents tools for storing data of large files when creating a database of a search information system, technological corporate computer networks, various project evaluation systems and other systems and algorithms for managing such files. It is noted that these systems and many other systems face similar problems when processing large text files. In search systems developed using a database of guidance documents and various databases, in electronic document management systems, the problem of saving files in separate folders remains relevant. The need to solve similar problems arises during the creation of systems for the electronic evaluation of projects submitted to various contests and competitions. Here, a database of projects submitted for the competition, databases of experts are developed, a knowledge base and other necessary data for evaluating the authors of projects are created, the files associated with these databases are stored, an intelligent analysis of the data is provided. When developing technological corporate computer networks and systems like these networks, the software must support and supply maintenance personnel with the necessary operational and technical documentation. It is necessary to create an interface for the maintenance personnel, which will ensure their quick learning of the system. The software interface should provide access to operational documents and technical documents on basic hardware. The software interface should provide the options of selection, recording and reading of documents, as well as their correction if necessary. For this purpose, the paper covers the advantages of creating same-level folders by the software and storing large files in such folders. To simplify the writing of software codes, such folders and the program folders of the software are to be stored in one folder. Program fragments for accessing such folders and files are given.

Keywords: guidance documents database, technological corporate networks, project evaluation systems, same-level folders, automatic reading of folders

Ф. Г. Пашаев, д-р техн. наук., гл. науч. сотр., e-mail: pasha.farhad@gmail.com,
Institute of Control Systems of the Azerbaijan National Academy of Sciences,

С. Е. Пашаева, преподаватель, e-mail: pasayevasevinc5@gmail.com,
Nakhchivan State University,

Г. Т. Наджафов, преподаватель, e-mail: hasan_nacafov@mail.ru, "Nakhchivan" University,

Б. А. Сулейманлы, соискатель, e-mail: bahram.suleymanli@gmail.com,
Institute of Control Systems of the Azerbaijan National Academy of Sciences

Алгоритмы работы с папками и файлами

Представлены инструменты для хранения данных больших файлов при создании базы данных поисковой информационной системы, технологических корпоративных компьютерных сетей, различных систем оценки проектов и других систем и алгоритмов работы с такими файлами. В статье отмечается, что эти системы и многие другие системы сталкиваются с похожими проблемами при работе с большими текстовыми файлами.

В системах поиска, разработанных с использованием базы директивных документов и различных баз данных, в системах управления электронными документами задача сохранения файлов в отдельных папках остается актуальной. Необходимость решения аналогичных проблем возникает во время создания систем электронной оценки проектов, представленных на различные конкурсы. Здесь разрабатываются база данных проектов, представленных на конкурс, базы данных экспертов, создается база знаний и других необходимых данных для оценки авторов проектов, организуется хранение файлов, связанных с этими базами, проводится интеллектуальный анализ представленных данных. При разработке технологических корпоративных компьютерных сетей и систем, подобных этим сетям, программное обеспечение должно поддерживать и обеспечивать обслуживающий персонал необходимыми эксплуатационными и техническими документами. Необходимо создать интерфейс для обслуживающего персонала, который обеспечит ускоренное освоение системы. Интерфейс программного обеспечения должен обеспечивать доступ эксплуатационным документам и техническим документам по основным техническим средствам. Интерфейс программного обеспечения должен обеспечить выбор, запись и чтение, а также корректировку документов при необходимости. Для этой цели даются преимущества создания с помощью программного обеспечения папок одного уровня и хранения больших файлов в таких папках. Для упрощения написания программных кодов предусмотрено хранение таких папок и папки с программным обеспечением в одной папке. В статье приведены программные фрагменты для обращения к таким папкам и файлам.

Ключевые слова: база данных директивных документов, технологические корпоративные сети, системы оценки проектов, папки одинакового уровня, автоматическое чтение папок

1. Introduction

It is known that the detailed information on the documents included in the database of the search information system and the organizations sending these documents are collected in the Directive Document Database [1, 2]. The large files included in the system are written to the Data Base. The detailed information about the documents included in the Data Database of the search information system of the Directive Databases and the organizations sending these documents is kept as follows:

- Registration number of the document — ID;
- The type of the document;
- The sender organization;
- Source of the document;
- The date of entry of the document;
- Executive Officer;
- Brief content of the document. The document's summary or abstract may be in the document itself. It can be created by special software only if not available [3—5];
- The keywords in your mouth. These can also be found in the document itself. If not, can be created by special software;
- The status of the work on the document;
- The deadline for the last application;
- Information about the last applicant;
- Period of execution of the document;
- The address of the physical folder where the original is stored. Special folders are created to keep information accompanying the original documents and documents because of possible being large size. Physical folders are also created to keep up-to-date about the current status of the document and its associated data. The address of these folders and files must be stored in the Database.

Except for the above attributes, other attributes may also be included in the Database, depending on the issues that are necessary. For example:

- List of Executives;
- A list of sections that are intended to perform work on the document, and so on.

As you can see from the list, the system is intended to create custom folders to store large files or files.

The same problem is observed during the creation of many other systems. Examples of such systems are Technology Corporate Computer Networks and various Project Evaluation Systems. The same approach can be used to create information bases for these or other systems. When creating these networks, it is essential that the data storage and retrieval details of the files contained within this network and the addresses of the files stored on the network are stored in the database.

2. Problem statement

The article provides for the creation of such folders, the names, dates of creation, and other attributes of files that are collected in these folders, automatically generating algorithms for reading and arranging arrays within software tools. It is suggested that the fragments of the created algorithms are implemented in DELPHI programming language.

3. Solution of the problem

Keeping files in equal level folders

It is known that the software that generates multiple files, enabling them to store, archive, and maintain a large number of files first of all should solve the problem of storing these files. Analysis of

the software of various information systems shows that one of the ways to solve this problem is to create folders with the same folder as the folder on which the software is stored and to store the files in those folders. For example, we can specify the sequence of the following folders:

```
Y_Q_A
  Q_A_Proq
  Q_A_Dat
```

The Q_A_Proq and Q_A_Dat folders are included in the Y_Q_A folder and are the same level folders. The software can be written to the first folder, while the second folder contains files with "dat" extension, which stores code from analogue dialers. As you can see from the Delphi fragment, the codes to write are written to the AR_BT array and the number of bytes written to the F_NN integer-type variable.

File to be written The Q_A_Dat folder can be written as '.. \ Q_A_Dat \' as it is identical to the application folder. The name of the file has been adapted to the folder name, which has been added to the name of the toolbar and was obtained by the datetimestostr (now) team. Due to the datetime type that can be used in the future, the dots are replaced by dashes. All of these statements were made in the following DELPHI program fragment.

```
str_r: = ' ' + datetimestostr(now);
for v_ij: = 1 to length(str_r) do begin
```

```
if str_r[v_ij] = '.' then str_r[v_ij]: = '-';
end;
fil_name:= '..\Q_A_Dat\' + 'S_DAT04' + str_r + '.dat';
AssignFile(tof, fil_name);
Rewrite(tof, 1); //.. Record size = 1
BlockWrite(tof, AR_BT, F_NN, NumWritten);
CloseFile(tof);
```

The program fragment can be organized as a procedure (subprogram) to be used repeatedly. The name of this subprogram can be written by Write_Proc (Write_Proc).

In this section, str_r and fil_name are string-type variables. The Str_r variable is added to the file name by typing the current date and time. In the fil_name variable, the name of the file is generated according to the above. This name contains the name of the Q_A_Dat folder with the same software as well as the current time and date on the file name. ToF is a variable file type and is used to work with.dat files. The Ar_BT array is an array of bytes of type and the code to be written to the file is written to the array as mentioned above. The F_NN variable is a integer-type variable and saves the number of bytes to be written to the file.

As a result of repeated requests, the files in the folder can be written as in Figure 1.

The advantage of the specified structure is that the Y_Q_A folder can be placed on any logical disk or any folder with folders inside it.

Depending on the issues that the local information system faces and solves the corporate, technological network, other folders of the same level may also be created. For example, the "Intelligent Monitoring System for latent period offailures for Compressor Devices"-has some of the key software technical passports, maintenance papers and other essential software software placed in the "doc" folder [6–10]. Here's also the Operator.doc file, which was created to guide the system operator. A system information screen has been created to access the documentation collected here (Figure 2, see the second side of the cover).

Documents written in the "Doc" folder are shown in Figure 3.

As mentioned above, the information search systems like the directive document database information retrieval system create folders to keep large volumes of documents.

S_DAT04	01.12.2017 0-00-31	01.12.2017 0:00	DAT File	16 KB
S_DAT04	01.12.2017 0-02-14	01.12.2017 0:02	DAT File	16 KB
S_DAT04	01.12.2017 0-03-57	01.12.2017 0:03	DAT File	16 KB
S_DAT04	01.12.2017 0-05-40	01.12.2017 0:05	DAT File	16 KB
S_DAT04	01.12.2017 0-07-24	01.12.2017 0:07	DAT File	16 KB
S_DAT04	01.12.2017 0-09-07	01.12.2017 0:09	DAT File	16 KB
S_DAT04	01.12.2017 0-10-50	01.12.2017 0:10	DAT File	16 KB
S_DAT04	01.12.2017 0-12-33	01.12.2017 0:12	DAT File	16 KB
S_DAT04	01.12.2017 0-14-16	01.12.2017 0:14	DAT File	16 KB
S_DAT04	01.12.2017 0-15-59	01.12.2017 0:16	DAT File	16 KB
S_DAT04	01.12.2017 0-17-42	01.12.2017 0:17	DAT File	16 KB
S_DAT04	01.12.2017 0-19-25	01.12.2017 0:19	DAT File	16 KB

Fig. 1. The sequence of writing files in the Q_A_Dat folder

ac297e_atex_1	13.04.2006 16:23	Adobe Acrobat D...	359 KB
algor01.doc	01.03.2007 15:55	Документ Micros...	298 KB
AS-062	13.04.2006 17:23	Adobe Acrobat D...	200 KB
ASA 062	14.04.2006 15:09	Adobe Acrobat D...	178 KB
Bp188212	22.10.2002 10:18	Adobe Acrobat D...	424 KB
Cpu188-5	29.11.2002 19:01	Adobe Acrobat D...	541 KB
liter01.doc	06.03.2007 11:59	Документ Micros...	64 KB
mc35it_hd_v0200	17.12.2006 11:16	Adobe Acrobat D...	3 590 KB
Operator	27.10.2012 13:18	Документ Micros...	134 KB

Fig. 3. Doc folder of Intelligent Monitoring System for Compressor Devices

Commands for reading and editing text files

To ensure that the files stored in the system's Doc folder or other folders are accessed by software, reading these files or performing corrections in text files the following types of commands can be used to:

— The command to read text files and make corrections if necessary:

```
begin
ShellExecute(handle,nil,'C:\mk301_2_b\doc\
Operator.doc',nil,nil,sw_shownormal);
end;
```

This command provides open and read the Operator.doc file that is written in the "C: \ mk301_2_b \ doc" folder and guides the system operator's operation. If there is a need to make any additions to this document during the exploitation process, the document can be routinely corrected as it is opened as a word document. Attachments can be stored in the document.

— The command to read the system's algorithms and correct them if necessary:

```
begin
ShellExecute(handle,nil,'C:\mk301_2_b\doc\al
gor01.doc',nil,nil,sw_shownormal);
end;
```

This command will open and read the algor01.doc file, which is written in the "C: \ mk301_2_b \ doc" folder and contains the system's algorithms.

— The command that provides reading of passport of the vibrator converter of Intelligent Monitoring System of hidden periods of accidents for Compressor Devices:

```
begin
ShellExecute(handle,nil,'C:\mk301_2_b\doc\
AS-062.pdf',nil,nil,sw_shownormal);
end;
```

This command will read the technical and operational characteristics of the AS-062 vibrator converter which are written in the "C: \ mk301_2_b \ doc" folder and produced by Bruel and Kjaer. This information is provided in the AS-062.pdf file.

The Intellectual Monitoring System of hidden periods of accidents for Compressor Devices has been designed to monitor the changes in the technical condition of compressor and other strategic facilities, to assess the seismic stability of these facilities. The signals from the AS-062 vibrator converter are stored as 12-bit codes using the Analog Digital Converter (ARC) of MICRO PC type controls. The signal's 4-second codes are written as dat files in the same BASA_S folder as the software. If the signal read frequency is 2 Khs, then 8000 bytes will be recorded per file.

Algorithms for working with Dat files

The following two issues should be resolved in relation to this base. The first issue is to convert the data in a single file into a type of real number, and the second issue is to read the names and dates of the files in the directory.

The solution of the first issue.

— Open the selected file type and write it to AR_BT. This array must have at least 16,000 bytes. The first byte of each pair is the large byte, the second is the small byte. If the code is a positive number, the big byte will be less than 8, meaning the 12th bit will be zero. Otherwise, the large byte will be greater than 8, meaning that the 12th bit will be 1;

— We can store real numbers that can be retrieved after transformation in big_r real numbers. Write a large part of each pair of AR_BT in byte_text. Therefore, if big_k < 8 is big_r number

$$\text{big_r}[i] = \text{AR_BT}[i*2-1]*256 + \text{AR_BT}[i*2]$$

Calculated like that. Otherwise

$$\text{big_r}[i] = -1*((255-(\text{AR_BT}[i*2-1]))*256 + \text{not}(\text{AR_BT}[i*2]))$$

can be calculated as.

As a result, the big_r real numbers massive can be used for different purposes, including analysis and graphic design.

The solution of the second issue.

To save the names and creation dates of the files in the folderFp:

TSearchRec;

FAge: Integer;

FileParam: TDateTime;

A structured record should be created. At this time FindFirst('.\BASA_S*.dat', faAnyFile, Fp) with the command the necessary information about the first file is written to the Fp record. FindNext(Fp) and the command writes the files in the folder to the Fp record. Both two command Fp.Name file's name in the parameter, FileAge('.\BASA_S\' + filename); the date of the file being written in the parameter, by FileParam: = FileDateToDateTime(filedate), DateToStr(FileParam), TimeToStr(Fileparam) parameters give accordingly the date and time of writing. By using these commands and parameters properly, the names and record dates of all files in the BASA_S folder can be written to the arrays. As a result, it will be possible to carry out the intended operations:

```
procedure TForm1.Button2Click(Sender: TObject);
var
Fp:TSearchRec;
FAge: Integer;
FileParam: TDateTime;
label m_end_7;
begin
```

```

    f_c := 0;
    memol.Clear;
    if FindFirst('..\BASA_S\*.dat', faAnyFile, Fp) = 0
    then
    begin
        f_c := f_c + 1;

        mas_string[f_c] := Fp.Name;
        fileDate := FileAge('..\BASA_S\' + mas_
string[f_c]);
        FileParam := FileDateToDateTime(filedate);
        date_01 := DateToStr(FileParam);
        time_01 := TimeToStr(Fileparam);
        X_X := STRTODATE(DATE_01) +
        STRTOTIME(time_01);
        if fileDate > -1 then
        x_m[f_c] := X_X;
        end;

    while FindNext(Fp) = 0 do
    begin
        f_c := f_c + 1;
        mas_string[f_c] := Fp.Name;
        fileDate := FileAge('..\BASA_S\' + mas_
string[f_c]);
        FileParam := FileDateToDateTime(filedate);
        date_01 := DateToStr(FileParam);
        time_01 := TimeToStr(Fileparam);
        X_X := STRTODATE(DATE_01) +
        STRTOTIME(time_01);
        if fileDate > -1 then
        x_m[f_c] := X_X;
        end;
    FindClose(Fp);

```

Result

In the article saving methods information about large files, algorithms of working with such files while creating Data Bases of various systems. It has been shown that information about such files, including files in which folders are stored, is written in the

Database. The advantages of keeping files in equal folders with software are provided. It was noted that, information about software tools used in the creation of system, exploitation documents of system can be kept in such folders and this can increase exploitation quality of the system. The article recommends creating software screens to work with such files. The software contains algorithms for working with folders and folders for files stored in such folders.

Methods, algorithms and software fragments can be used to create software for different systems.

Reference

1. **Pashayeva S. E.** Creating a Directive Document Search System database, Nakhchivan State University, *Scientific Works*, 2016, no. 3 (77), pp. 59–67.
2. **Pashaeva E.** Local search documents in corporate life, *VII All-Ukrainian Scientific-Practical Conference, Computer Science and Systems Sciences*, Poltava, 10–12 March 2016, pp. 223–225.
3. **Aliguliyev R., Hacirahimova M. Sh.** Optimization model for auto-synthesis of texts, *Problems of Information Technologies*, 2015, no. 2, pp. 96–102.
4. **Alguliev R. M., Aliguliyev R. M., Hajirahimova M. S.** Quadratic Boolean programming model and binary differential evolution algorithm for text summarization, *Problems of Information Technologies*, 2012, no. 2 (6), pp. 20–29.
5. **Pashayeva S. E.** Methods for working with text files, *News of the Azerbaijan National Academy of Sciences, Information and Management Problems*, 2016, vol. XXXVI, no. 6, pp. 104–111.
6. **Guluev G. A.** Robust system of identification of the vibratory condition and forecasting of emergency conditions of compressor units, *SOCAR Proceedings*, Jun 2012, no. 3, pp. 65–70.
7. **Aliev T. A., Musaeva N. F., Guluyev G. A., Sattarova U. E.** Noise Indication of Change of Dynamical Condition of Production Objects, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2011, no. 8, pp. 2–5.
8. **Aliev T. A., Guluyev G. A., Pashayev F. H., Sadygov A. B.** Noise monitoring technology for objects in transition to the emergency state, *Mechanical Systems and Signal Processing*, February 2012, vol. 27, pp. 755–762.
9. **Aliev T. A., Guluyev G. A., Pashayev F. H., Sadygov A. B.** Algorithms for determining the coefficient of correlation and cross-correlation function between a useful signal and noise of noisy technological parameters, *Cybernetics and System Analysis*, May 2011, vol. 47, no. 3, pp. 481–489.
10. **Guluyev G. A.** Application of the neural network for identification of the vibration condition of the compressor unit, *Azerbaijan Oil Industry*, 2013, no. 1, pp. 40–45.