

NEW ALGORITHMS IMPROVING THE WORK OF AN INFORMATION SYSTEM FOR THE BAILIFF'S OFFICE

Ciechański P., Bąkała M., Bąkała A.*

Abstract: The use of ICT enables enterprises to increase their overall efficiency and makes them more competitive. Bailiff office can be an example of a firm not from ICT branch, but widely using new information technologies. The idea of this paper is to show some modifications of existing algorithms of an information system for a bailiffs office to enhance its efficiency. This concerns selected processes essential for efficient work of the office. In the first part of the paper the needs and the opportunities of information technologies for the bailiffs are shown. The necessity of EDI use especially in a bailiffs office has been pointed. Some work areas of the application Komornik SQL where efficiency problems appear have also been presented. These caused lower work of the whole team. Then solutions tending to eliminate human error and manual data introducing as well as its verification have also been described. Four algorithms improving the mentioned system are presented.

Key words: bailiffs, bailiffs office, efficiency, information system for the bailiffs

Introduction

Until the mid-1990s, the electronic data interchange was rather a rarely heard phenomenon amongst bailiffs in many european countries. This applied not only to the acquisition of the information required for executing judgements or execution orders, but also for performing tasks such as attachments management. Traditionally, bailiffs simply applied for and received the information they required from the authorities to perform their tasks (such as verifying addresses with the municipality or tracing assets via a debt collection agency) in written form **Błąd! Nie można odnaleźć źródła odwołania..** Nowadays computerization reaches even the most hermetic and specific law areas such as bailiffs offices. Thanks to that the bailiffs can perform part of their duty electronically much faster and with lower costs.

Increasing number of formalities connected with the execution management, and the increasing number of debtors, involve much greater demand for IT solutions allowing to standardize and speed up the process of enforcement. Another demand is the elimination of errors associated with the amount of data and human mistakes. The aim of this article is to indicate some problems appearing while using selected IT tools dedicated for the bailiffs office. As the exemplary IT tool the application Komornik SQL has been chosen. The authors suggest possibilities of improvement in areas that have been indicated. The methodology bases on Rapid Application

* **Piotr Ciechański**, Lodz University of Technology Faculty of Electrical, Electronic, Computer and Control Engineering, **Marcin Bąkała, PhD**, **Anna Bąkała, PhD**, University of Lodz, Faculty of Economics and Sociology

✉ corresponding author: mbakala@uni.lodz.pl, abakala@uni.lodz.pl

Development (RAD) designed to ensure that developers build the systems that the users really need [6]. Thus before introducing any improvements, the authors gathered some information about the application from the bailiffs office employers using it in practice.

In this article first some IT tools for the bailiffs offices in Poland are described. Special attention was paid to the application Komornik SQL. Then some weaknesses of it are shown, in four areas: the method of sending notices of the execution initiation, the process of treatment of each case, problems with the connection to the main portal EPU and the process of scanning documents for the program. In the next section some suggestions of solving the above problems are presented.

Selected IT tools for the bailiffs offices in Poland

The role of courts in general is to produce enforceable decisions, in other words: to provide title **Błąd! Nie można odnaleźć źródła odwołania..** The title provision gives the other parties tools enabling to further action. Having a judicial decision in hand, one can go to the bailiff for contract enforcement. In all those actions there's a need for automatization. As mentioned in the introduction, it has been introduced and realized as computerization process.

Nevertheless most bailiffs' offices in Poland face with certain efficiency problems of computer systems and services cooperating with the bailiffs. In this section some computer tools for the bailiffs in Poland are described, especially the program Komornik SQL as some of its features have been improved as the result of further work.

The bailiffs offices are nowadays linked with the systems CEPIK 1 (Central Register Of Vehicles And Drivers), OGNIVO (system issued by the National Clearing House KIR2, allowing to find debtors' bank accounts), EPU3 (Electronic Reminder Proceeding), and many others. This association with specialized systems, in principle, eliminates the possibility of not having a computer system supporting the office.

One of available solutions dedicated to judicial officers in Poland is "KomornikSQL" Currenda Ltd. The system works in client-server architecture based on Microsoft SQL Server. The system allows for handling the cases since the beginning to the end of their "lives" in the bailiffs office. It also has a built-in modules for scanning documents (if you decide to have all documents in the database), e-mails (built-in SMTP client), and to connect the mentioned modules like CEPIK, OGNIVO and EPU.

The great advantage of the program is the database of documents. Thanks to this the new letters or statements can be created on the bases of templates, which

¹ Centralna Ewidencja Pojazdów I Kierowców

² Krajowa Izba Rozliczeniowa

³ Elektroniczne Postępowanie Upominawcze

eliminates human mistake. The system also has the accounting function. This eliminates the tedious process of counting the debt for each case separately.

A very useful functionality is the construction of the bailiff reports (MS-KOM 23), which the bailiff must consign twice a year. Unfortunately, the office work is based not only on supporting the debtors. Equally important is to meet the needs of the creditor who delivers the cases which are the source of the office's income. It is very common for the creditors to prepare the requirements list for reports and statements that must be provided every given period of time.

Despite so many features and facilities for the Bailiffs Office, the system has many faults which often make the work impeded. Obviously this doesn't concern the basic functionality but the more advanced or rarely used by the Bailiffs components.

Often, scalability is of great importance in the activity effectiveness. For law firms with an annual impact of about three thousand cases certain features of the program are performing well and are not the "bottleneck". In the case of an office with one hundred thousand annual cases it could be a great problem to make use of the functions, which finally exclude these.

In this article the second case (one hundred thousand annual actions) shall be analyzed. The aim of the analysis is to point how to fix the existing faults and create new facilities supporting or enhancing the effectiveness of the Bailiffs Office work.

Weaknesses of Komornik SQL as a selected IT tool for the Bailiffs Office

For the purposes of this study a system "Komornik SQL" has been selected as an example of a system dedicated to bailiffs institutions. Although it has lots of advantages, in some cases, the application is not enough efficient.

All the disadvantages described in this section have been pointed out by the workers of the bailiffs office. Thanks to this the real need of improvement has been indicated.

The first weakness that has been perceived, was the method of sending notices of the execution initiation according to the art. 8 of Code of Civil Procedure⁴.

Most of law firms in Poland have this problem as it affects all bailiffs who carry out executions outside of their district (which is determined by the properties of the District Court⁵ to which they belong). This implies a duty to inform electronically the bailiffs from other polish districts of initiating the execution in their circuit.

This is a matter of effectiveness primarily because for each case, the person sending such a notice first must find the appropriate District Court for the debtor's place of residence. The second step is to find all bailiffs who work in the area, to determine their e-mail addresses and send an email to all of them with the

⁴ Kodeks Postępowania Cywilnego

⁵ Sąd Rejonowy

mentioned notice. Although “KomornikSQL” has options for sending e-mails to multiple addresses, the search process is time consuming, and when the number of cases reaches tens of thousands, it becomes a huge problem.

Below a block diagram illustrating an algorithm of notifying the other bailiffs of the execution initiation has been presented (Fig. 1).

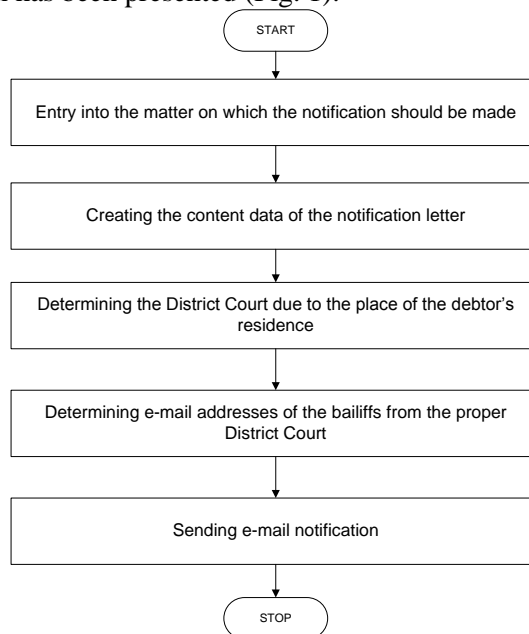


Figure 1. Block diagram of the algorithm of the bailiffs' initiate execution notice sending

Source: own elaboration

Another weakness area that has been noticed was the process of treatment of each case coming from the EPU. Cases of this type are automatically downloaded into the database via „KomornikSQL”. Unfortunately, in each of the cases some information appropriate for the debtor's place of residence must be introduced manually. These are: the District Court, Social Insurance Institution⁶, Inland Revenue⁷, and the approximate distance from the bailiff's office to the debtor's residence. The amount of data that must be completed and the difficulty of finding it meant that, one person was able to treat in this way maximum about one hundred cases. It became clear that with rising number of cases, this work would take on average 2 days a week of five people. Considering further development, thus enlarging the proportion given, this solution was unacceptable. Additionally the human mistake would cause about 5% of wrong treated cases.

⁶ ZUS

⁷ Urząd Skarbowy

The following block diagram (Fig 2) shows the algorithm of completing a new case with the required data.

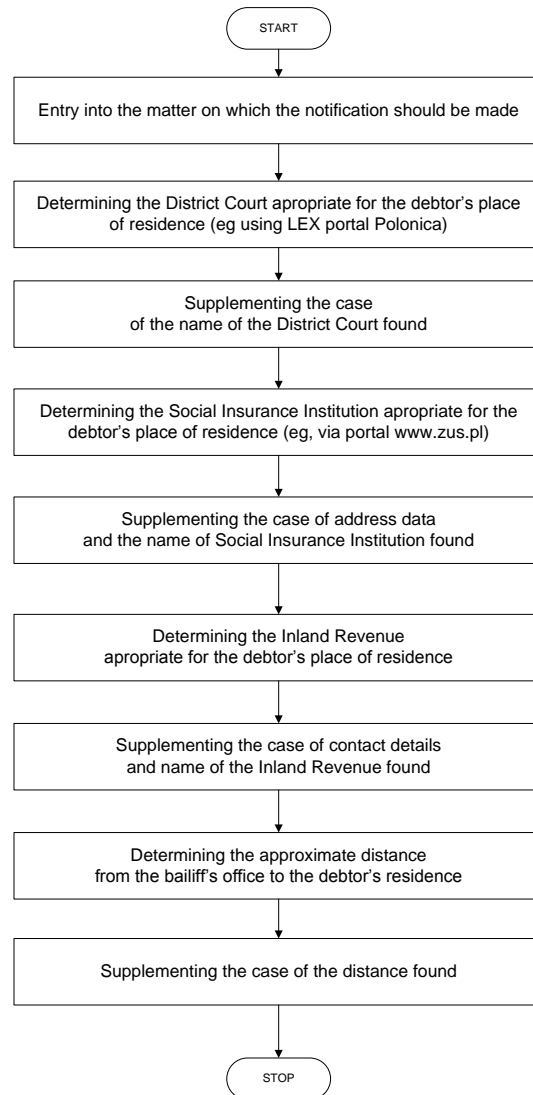


Figure 2. Block diagram showing the algorithm of supplementing the case with the required information

Source: own elaboration

The next problem is the portal EPU itself. All the cases are charged from this and it is overloaded. There are often problems with the connection which results in delays in the work of the office. However, its biggest fault are the errors occurring when the connection is broken. The cases happen to be duplicated, disappear, etc.

During the tests it has been noted that while working with EPU, not by “KomornikSQL” system but through the portal site, the stability and the speed of the cases download were improved (for automatic processing of cases, they are drawn manually in the XML file and not sent to the “KomornikSQL” system). Although the problem of stability decreased, the problems with the EPU still appeared. Nevertheless efficient work was possible to some extent. Unfortunately, at some point it became clear that the XML files downloaded from the portal had fatal errors preventing the work on these issues. Lack of support from the EPU IT department, and the increasing number of cases in the portal, resulted in inventing a solution for the problem of errors in XML files.

Below (Fig. 3) a block diagram illustrating the algorithm of the cases export from EPU portal to XML and their import into the “KomornikSQL” has been presented.

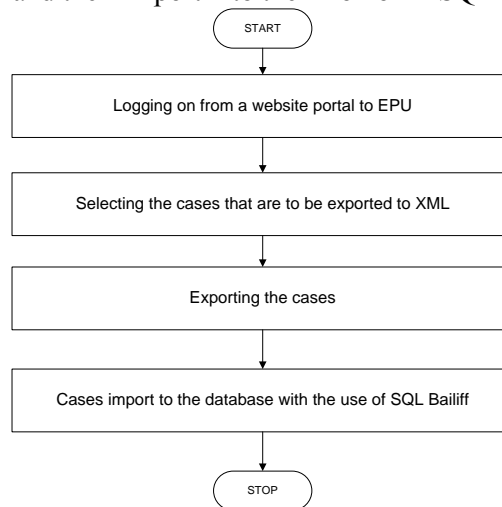


Figure 3. Block diagram showing the of the cases export from EPU portal to XML and their import into the “KomornikSQL”

Source: own elaboration

The last major problem was the process of scanning documents for the program. If the firm decides to have all documents in electronic version, additional work (scanning) is required.

The “KomornikSQL” system has its own unit for scanning documents, however, it is quite impractical in terms of use on a larger scale; while increasing number of scanned documents, it runs slower and slower. There is also a second method, which is not affected by the amount of scanned documents, but due to the impracticality of use with large quantities of documents shall also be disqualified.

Below (Fig. 4) a block diagram describing the algorithm of scanning module in the system “KomornikSQL” has been shown.

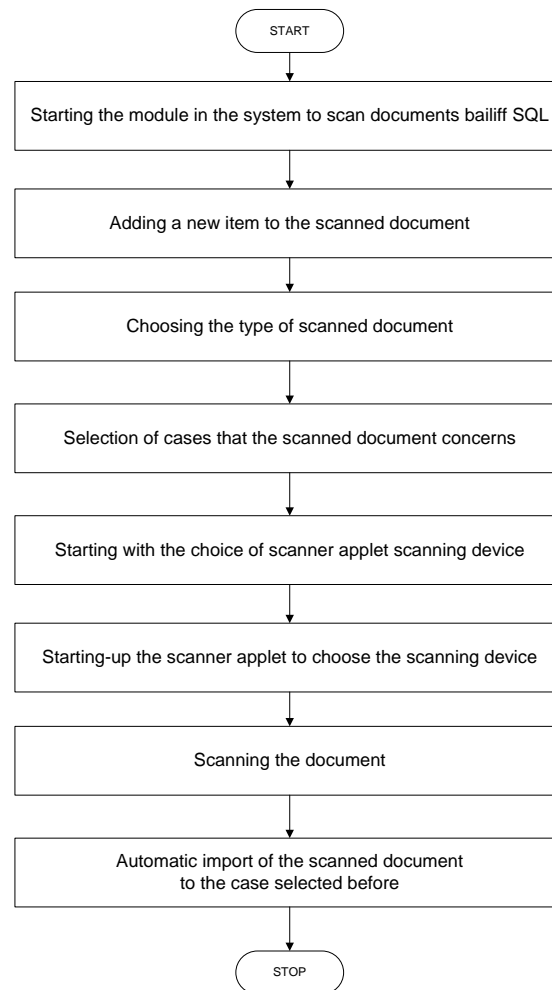


Figure 4. The algorithm of scanning documents in the system “KomornikSQL”
Source: own elaboration

Improvement solutions for the Komornik SQL application

In this section some solutions improving the work of a bailiff office due to automatization of selected processes have been described. Problems appearing while using the application, that have been described before, are as follows:

1. The method of sending notices of the execution initiation according to the art. 8 of Code of Civil Procedure.
2. The process of treatment of each case coming from the EPU.
3. The portal EPU itself (overloaded system, problems with the connection).
4. The process of scanning documents for the program.

All of the above mentioned disadvantages have been improved with the use of programming tools described in the following part of the article.

The proposed solutions were firstly based on a script language AutoHotkey only. AutoHotkey is a free, open-source utility for Windows. It allows for automating almost anything by sending keystrokes and mouse clicks, creating hotkeys for keyboard, joystick, and mouse, creating custom data-entry forms, user interfaces, and menu bars. It also enables to convert any script into an EXE file that can be run on computers that don't have AutoHotkey installed.

It has very advanced features of writing and reading text such as finding the character or string, downloading text files of specific variables, splitting lines of text, write, and read the text from memory. Besides its main task which is the work automation, it can run and close processes, make appropriate changes to the registry and create a simple graphical interface so that some scripts written in it can carry the name of an application.

The advantages mentioned, an open license, and the simplicity of programming in this environment are the reasons for selecting the solution. The stability and speed for applications written in this environment showed that the right choice has been made. Unfortunately, at some point, it appeared that AutoHotkey despite its many advantages is not enough to solve the problems mentioned in this article. A more powerful language with greater capabilities was necessary. The authors have chosen C#.NET, environment created by Anders Hejlsberg for Microsoft. This language has been chosen due to its intuitive character, very good development environment, big amount of documentation and a very good technical support. More over applications written in this environment are of very high efficiency.

More complex parts of the applications have been created in Microsoft Visual C# 2010 Express. This product is a free set of tools that Windows developers at any level can use to create custom applications using basic and expert settings **Błąd! Nie można odnaleźć źródła odwołania..**

Creating and sending notices of initiation of execution to other bailiffs

For the need of sending notices to other judicial officers to initiate the execution, a new algorithm has been proposed. It runs more efficiently, omitting the human element.

The first step of the algorithm is to export the cases in which there should be a notice of initiation of execution to an XML file (which is allowed in Komornik SQL system). This file contains all data contained in the case, therein the debtor's address data which is required to create the initiation of enforcement notices. When the data is exported to XML, a dedicated application runs.

It is worth mentioning that the debtor is not always a private individual, but e.g. a company. In such case the tags in an XML file change. However, this was included in the application's operation, so no data is discarded. The resulting text files are already prepared to provide the data needed to generate and send notifications to initiate execution.

The next and final step is to run a script written in AutoHotkey environment, which uses the above mentioned data from the previously prepared text file to complement the blank template in RTF (Rich Text Format). The script also sets the Internet addresses of bailiffs of the place of the debtor's residence.

For this solution a database was prepared. It contains of all e-mail addresses bailiffs acting on Polish territory, together with either the name assigned to the District Court, to which it belongs. The completed RTF template is automatically printed, saved to your computer and sent as an attachment in an email to addresses found in the database by the script. After the process of sending automatically starts the application Komornik SQL and each generated statement is imported into the history of the cases.

Completing the address data of the institutions

To solve the problem of the time-consuming manual completion of address data of institutions mentioned in chapter 3 (the District Court, Social Insurance Institution, Inland Revenue, and the approximate distance from the bailiff's office to the debtor's residence) the environment of Microsoft Visual .NET with C# has been used.

The greatest difficulty was to develop a new algorithm for filling the case with the required data. It was necessary at the design stage already to pay particular attention to errors that might arise during the operation. Even a small mistake in the operation of the program introducing false data to the case, would be to block the work of the whole bailiff at an undetermined time when processed required amount of times **Błąd! Nie można odnaleźć źródła odwołania.** The algorithm has been implemented in a new application „Filled XML Builder”. When it starts, first the instance dedicated for bailiff firm is taken from the EPU portal. Cases stored in an XML file have only the basic information about the debtor that is not sufficient to start of the execution. The next step is to load the downloaded XML file to the "Filled XML Builder". Next point is the extraction of postal codes in the XML file for each case to a separate file and extracting the ID number of writings called "Application Enforcement Order"⁸. Then the zip codes are used to determine the appropriate departments. After exporting the data, a special function will re-load the XML file. As a result a user gets an XML file complete with all the data needed to carry out executions, which we import into the Komornik SQL.

Customizing the portal EPU

Correct and fast work of the portal means stability and speed of the office work, thus the effective execution and satisfaction of creditors. The tests showed that apart from the lack of stability in the operation of the EPU, some errors appeared in the XML files that were downloaded. They concerned the ID application

⁸ Wniosek Egzekucyjny

execution. Admittedly it was possible to find the correct ID numbers, in the source of the EPU portal page, but this method was very inefficient and unstable due to lack of proper operation of the EPU website. To solve this serious limitation partly the technical documentation of portal EPU was used **Błąd! Nie można odnaleźć źródła odwołania.** and C # environment. Direct communication with the EPU enabled adding new modules that significantly improved the speed and stability of the portal. Another dedicated application has been proposed. The first step of the application is to test the connection to the EPU. This function is used to verify proper work of the portal. If the system works, as a result we get the ID of the bailiff's office trying to connect to the system. While there appears a problem of the communication, the result is the entire description of the connection error **Błąd! Nie można odnaleźć źródła odwołania..** After next steps we get the XML file with the cases and the list of IDs of requests for downloaded cases. This file is ready to be filled with the use of the program "Filled XML Builder". Thanks to the correct ID numbers of applications, also the letters required to initiate the execution are downloaded. Once the correct file is imported into the Komornik SQL, the file with the ID numbers of execution requests should be selected as downloaded. Thanks to this the danger of downloading them again by mistake will be eliminated.

The process of scanning documents for the Komornik SQL program

Scanning through an additional module of Komornik SQL is very inconvenient, because after each scan of a letter, the whole list of previously scanned documents is shown. The display process is long due to the number of records. It consumes lots of the server's hardware resources, which affects lower speed of the other employees work in the system. It is possible to scan the documents outside of the additional module, but it is very laborious for the user. This involves the opening of each case individually and creating a new document to which only the file you import scanned by an external program. The advantage of this method is the speed, because it doesn't open a whole list of records of documents scanned before. And even though the second way is completely inefficient of the user, it turns out that an appropriate application is capable of adding the documents scanned before efficiently and without overloading the server

But there were still some questions of the scanning itself. In this case, the problem turned out to be more difficult to solve. The letters coming by traditional mail come from many different offices and their scanning is associated with assigning them to specific cases. It also requires describing the content and the origin of the letter in the title of document being scanned. This process was long and cumbersome for the user, thus another algorithm, greatly accelerating this action has been proposed. This algorithms also facilitates the work and eliminates human errors. The application process was written in AutoHotkey environment.

The work of this algorithm is as follows. The first step is to segregate letters by the type of recipient and the type of response. This is required because such letters can

be sorted quickly and efficiently scanned. Then the letters are placed in the scanner. At this point the the user chooses the scan date of the document, the sender's name and the type of response. Then the proper application start.

It should also be noted that the scanning device of the bailiff's office has a mechanical button starting the process of scanning the documents. When pressed, it enables to automatically place the document being scanned to the specified directory in the settings of the scanner.

Then the user must choose the type of case and its number. Next, the file of the scanned document is transferred from the default recording folder to a dedicated one. The file name is also changed due to better organization and later finding the files.

At the same time a line entry in the "wykaz.txt" in the same directory as the scanned documents is also added. Scanned letter correctly goes to the proper case and is appended to it the appropriate name.

Summary

The example of an IT tool for the baliffs office - Komornik SQL – allowed to point out some areas that seemd to slow down the work of the whole office, especially when the number of cases was bigger than one hundred thousand.

With the use of described programming tools the application has been equipped in new functionality.

In this paper selected applications and scripts to improve the bailiff's office work have been described. Some inconveniences of operating a program for the bailiffs – Komornik SQL (from the user's point of view) are pointed and solutions improving the work of the system are given. Better efficiency of selected processes means better work of the whole bailiff's office, thus it is a matter of big importance from the economical point of view as well.

In the first part of the article, some block diagrams showing the start state of using the program Komornik SQL have been shown. New algorithms, improving the work of the system are described. All block diagrams showing their work will be presented in a separate piece of work. It is essential to compare the results obtained with the use of new algoritms with the standard method using the program Komornik SQL. The efficiency tests are also to be carried out in further works.

References

- [1]. Struiksma D., Jongbloed A., *Bailiffs on the e-highway*, Information & Communications Technology Law, Volume 15, Number 2, June 2006 , pp. 201-206
- [2]. Reiling D., *Understanding IT for Dispute Resolution*, International Journal For Court Administration, April 2011, ISSN 2156-7964
- [3]. <http://www.microsoft.com/visualstudio/en-us/products/2010-editions/visual-csharp-express>
- [4]. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, *Algorytmy i struktury danych* 2003

- [5]. Usługa sieciowa e-Sąd, Usługa sieciowa dla komunikacji powoda masowego/komornika z EPU ver. 1.14.1, <https://www.e-sad.gov.pl/>
- [6]. *What is Rapid Application Development?*
http://www.casemaker.com/download/products/totem/rad_wp.pdf.

NOWE ALGORYTMY USPRAWNIAJĄCE PRACĘ SYSTEMU INFORMATYCZNEGO DLA KANCELARII KOMORNICZNYCH

Streszczenie: Wykorzystanie technologii informatycznych w przedsiębiorstwie przyczynia się do zwiększenia jego efektywności, a co za tym idzie – wzmocnienia konkurencyjności. Nowoczesne technologie ułatwiają działalność w niemal każdej gałęzi gospodarki. Przykładem firmy funkcjonującej poza obszarem ICT, ale mogącej szeroko wykorzystywać technologie informatyczne, jest kancelaria komornicza. W artykule przedstawiono propozycje poprawienia wydajności systemu informatycznego dla kancelarii komorniczych Komornik SQL poprzez zastosowanie nowych algorytmów wybranych procesów. W pierwszej części artykułu omówiono potrzeby i możliwości technologii informatycznych usprawniających pracę kancelarii. Wskazano na potrzebę wykorzystywania elektronicznej wymiany danych w szczególności w pracy kancelarii komorniczych. Następnie zaprezentowano obszary działania systemu Komornik SQL, w których pojawiają się problemy z wydajnością, przekładające się na spowolnienie pracy całego zespołu ludzi. W dalszej kolejności zaproponowano zastosowanie rozwiązań dążących do wyeliminowania błędów ludzkich oraz ręcznego wprowadzania, tudzież weryfikacji danych. Omówione zostały cztery algorytmy usprawniające wspomniany system informatyczny.

Słowa kluczowe: kancelaria komornicza, efektywność, wydajność systemu informatycznego

新算法改進信息系統的工作，為執達主任辦事處

摘要：利用信息通信技術使企業能夠提高他們的整體效率，使它們更具競爭力。執達主任辦事處可以是一個公司的例子不是從信息和通信技術的分支，可廣泛利用新的信息技術。本文的目的是展示一個信息系統的一個法警辦公室現有的算法進行一些修改，以提高其工作效率。這涉及選擇辦公室高效工作必不可少的過程。在本文的第一部分信息技術為執需要和機會被示出。EDI 的使用，特別是在一個法警辦公室的必要性已經指出。應用 Komornik SQL 地方出現效率問題的一些工作領域也已經提出。這些造成了整個團隊的較低的工作。然後溶液趨向於消除人為錯誤和手動數據引入以及其驗證也已被描述。四種算法提高了系統所提呈現。

關鍵詞：對法警法警，法警辦公室，效率，信息系統。