

Simulator for testing hardware and software of the office system with RFID tags

Tadeusz Nowicki^{1*}, Krzysztof Chlebicki¹, Dariusz Pierzchała¹, Kazimierz Worwa¹, and Robert Waszkowski¹

¹Military University of Technology, Cybernetics Faculty, 00-908 Warsaw, gen. S. Kaliskiego str., Poland

Abstract. This paper presents the method for examining the properties of the RFID-tagged document management system. The system is composed of computers, where the software for supporting processes of the RFID-tagged documents was installed. Furthermore, the system cooperates with many other elements of the office (cabinets, sluices, copiers, try rider, end so one). The examination of the properties of the RFID-tagged document management system is, in this case, complex due to the number of a possible examination scenarios. The simulator method for examining the system properties was design and implemented. It allows to conduct the examination of the properties in a short period of time for numerous testing scenarios.

1 Introduction

The paper presents a software simulator based on the method of testing the reliability and performance of a document management system with tags (tags) RFID. The system under study is composed of computers that have software for supporting document management processes with RFID tags installed. The system also cooperates with many elements of the office (closets, locks, copiers, trays, tunnels). Document ownership management system

RFID tags are complex in this case due to the large number of possible test scenarios. The built-in simulator allows you to use a simulated method to test system reliability and performance. It also allows you to carry out these property studies in a short time for a number of test scenarios.

The use of RFID tags (radio-frequency identification) for various types of applications is presented in [10], [30]-[34]. The application of this technology is found in numerous branches of the economy, including in the state administration [17] as well as in other organizations [6], including logistics [27] and [26], manufacturing [2], medical [11].

This technology allows permanently tagged RFID tags, tags, so that the radiated waves generated from the respective antennas send signals from those tags that record their existence at a certain distance from the antennas. So documents with RFID tags are passive objects. If there is no radiation from the antennas, they do not send any signals into the ether. In the case of antennas, the reading system enables the identification of multiple tags simultaneously in the reading field. RFID tags go through the last great transformation in the sense of their size. The size of the RFID tag is constantly decreasing.

Given the fact that RFID antennas send signals at a certain frequency, you can organize a modern document

management system with RFID tags, for example, in the law firm [12] and [13]. This type of system allows you to record a series of events in the room where your RFID document management system is located. The main events include:

- the emergence of new documents with RFID tags,
- the sale of a single RFID tag or package thereof outside the antenna location area,
- moving of the document with the RFID tag within the antenna
- registration of new RFID tags.

Implementing a document management system with RFID tags requires a specialized, multi-element environment to create a computer-aided process for identifying document flow. One of the most important studies of the RFID tagging system is the development of numerous experiments to test the reliability and performance of the system. Scenarios such as this can be many. This causes the number of variants of the system test scenarios to be enormous. It is worth taking into consideration that certain characteristics of using RFID tags are not uniform in time. They change quite often, which is because some RFID-tagged documents are out of system and new ones are emerging.

The concept of creating an event simulator [1], [8], [20] and [23]) generated signals of various types of activities of RFID document management system users. The simulator will operate in a network environment [24]. This allows for the preparation of a number of remotely operated user scenarios. Programmatic action scenarios can be generated automatically. Their number is not limited as they will be tested by the RFID tags management system, which will take place in virtually negligible time. The work is devoted to presenting a simulation of RFID tagging software, which will allow you to quickly perform complex tests of its essential system features related to its reliability and performance.

* Corresponding author: author@e-mail.org

2 Office with RFID tagged documents

The components of the office, apart from its typical components, using RFID technology include:

- input sluice with RFID antennas as access point to the office,
- cabinets with RFID antennas,
- document read tunnel with RFID tags,
- RFID tray reader,
- specialized copier with RFID module,
- printer server,
- RFID tagging software server,
- document flow management software system with RFID tags.

The RFID input lock is intended for registration filed or removed from a law office. The person entering or leaving the office also has an RFID tag in the form of a special personal chip. RFID cabinets are a natural document storage facility at the office. The document read tunnel serves the user of the law firm to submit the documents brought and left at the office with the simultaneous removal of them from the catalog already held. Similarly, a tray reader is designed to record individual documents with RFID tags. A specialized RFID printer records the fact that a document is being copied, but has a built-in verification that the document can be copied, by whom, and in how many copies. All these devices connect to the RFID document management system via a computer network controlled by a network router.

The RFID tag management system consists of the following components: Aurea BPM server business processes that describe the possible behavior of system users, a Cosmos server primarily serving the printer (copier) in the system, a Nofilis server for system configuration (setting up co-operating devices in the office) Receive signals from devices with RFID tags. Servers are a computing environment that processes document flow with RFID tags in the system.

3 Simulation method for office with RFID tagged documents investigation

For this purpose, a replacement of the original work environment of the office was shown, as shown in Figure 1. In this case, a series of event scenarios would be required for messages sent between RFID software agents from external devices to the RFID server of the system. In addition, you will have to track the response of the document flow management server to the occurrences.

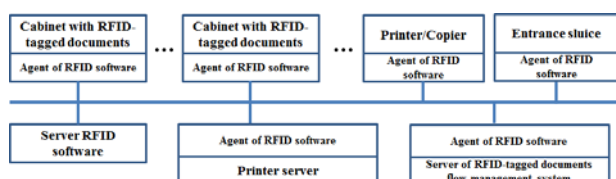


Fig. 1. System supporting the RFID-tagged document flow

It was decided to use an innovative method to test the reliability and efficiency of the system. A simulator

imitating document flow events was created, in this case generating signals that change the location of documents with RFID tags in the office system. Event simulators simulating the functioning of technical systems have been described, inter alia, in the model sense in monographs [8] and [23], and in the sense of implementation in monographs [1] and [24]. In normal mode, the RFID software server is only used to configure a scanner, which defines the devices that are part of the company's office.

In turn, the existence of documents with RFID tags is recorded because RFID antennas send messages to RFID software agents and these are then transmitted by the RFID software agent to the Document Management Server. Messages are XML strings. Thus, the idea of building a simulator of XML messages imitating the existence or non-existence and movement of documents in the system. The environment of the RFID tag assisted document flow system in Figure 1 was transformed into the artificial environment illustrated in Figure 2.

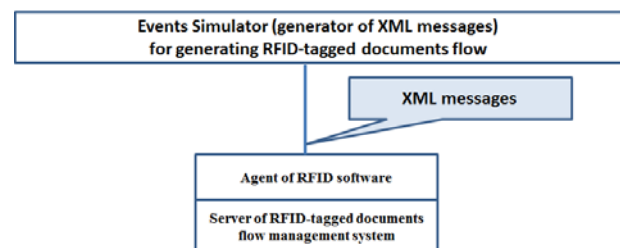


Fig. 2. Process related to the storage of the RFID-tagged documents

This makes it possible to prioritize RFID status papers in the office with their locations. This way, in the finite, soon to be able to study the reliability characteristics And the efficiency of the office system both in terms of the application software of the Document Management Server as well as the correct operation of records In the database. Without this modification, the study of characteristics would be extremely long.

4 Events in the system

The simulator should enable testing and testing of the performance and reliability of the RFID tag flow management system in a dedicated, complex system. The simulator is supposed to generate events by imitating activities carried out in a secret office equipped with specialized software. Currently, the following events are supported:

- Opening the front door - when opening the lock door, the device sends information about this event along with information about who did it and list of all RFID tags currently on the device,
- Closing the entrance door - When closing the lock outlet door, the device sends information about this event along with who it was and the list of all RFID tags currently on the device,
- Document Location - At the document location, the device sends information about this event along with a list of all RFID tags currently on the device,

- Document detection - When a document is detected, the device sends information about this event along with a list of all RFID tags currently on the device.

The simulator allows you to build the listed events into a form acceptable to the server module. The server module (monitor) in the real system communicates with devices in a secret office via http protocol. Transmitted events are XML files with a defined and structured structure. The monitoring module accepts XML files as call arguments on the services it issues.

4 Simulator architecture

An event-based scenario simulator constructs XML messages accepted by the server module and then sends them via http. The simulator has a graphical interface made using the JavaFX library and requires Java 8 virtual machine software (1.8). The simulator window is shown in Figure 3.

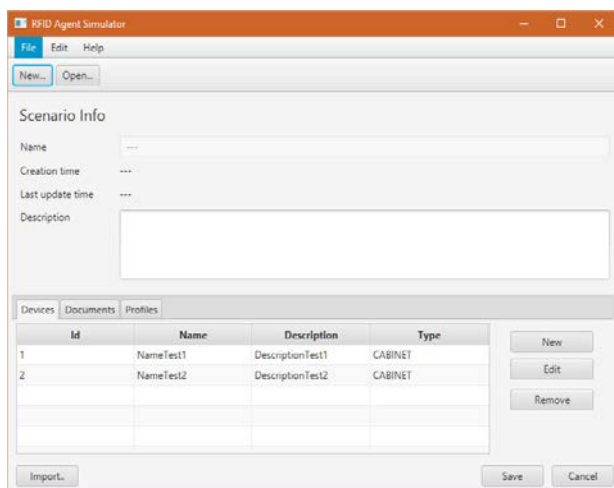


Fig. 3. Main simulator window

5.1. Description of the XML messages used in the simulator

In the real system, XML documents are used to describe events occurring in it. There are two types of these events:

- com: nofilis: crosstalk: event: tag-observation (tag-observation),
- com: nofilis: crosstalk: event: read-report (read-report)

The first is generated in response to a request to read a single RFID tag, while the second one is generated in response to a request for a report. About all RFID tags read in a single session. The descriptions of the XML documents used for exchanging data between the server and agents in cabinets, mash, trays, and tunnels have been established. Communication between the server and agents is strictly defined, and involves sending messages as XML strings between them: Person ID, Document ID, a moment of identification and a number of other important measured quantities. The sample XML document specifying the way to handle the tag-observation event is as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<events>
<event type="com:nofilis:crosstalk:event:tag-
observation" timestamp="1417780363293"
uuid="CCDEDBCA-6D07-2076-5CAE-
C1DBC42DEE5D" locationId="CABINET_3.R"
objectId="C01B2916">
<id="first-read" value="1417780363291" />
<id="last-read" value="1417780363291" />
<id="observationUUID" value="73B00D24-C3B6-
25FB-CFD1-F59157FC3856" />
<id="reads" value="1" />
</event>
</events>
```

5.2 Description of simulator application packages

On Figure 4 shows the main packages of the simulator.

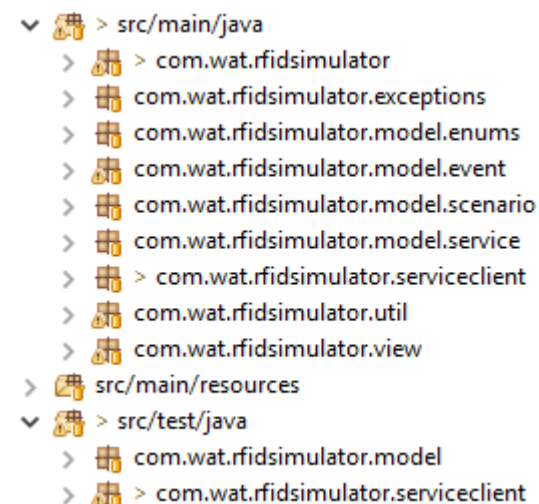


Fig. 4. Simulator application packages

Packages collect classes that perform specific and close actions. Here is an explanation of the purpose of the packages:

- com.wat.rfidsimulator - application runtime package,
- Com.wat.rfidsimulator.exceptions - package of classes responsible for operation and wrapping exceptions,
- com.wat.rfidsimulator.model.enums - enumerator class package,
- com.wat.rfidsimulator.model.event - a package of event classes generated by the simulation engine imitating real system events,
- Com.wat.rfidsimulator.model.scenario - JAXB class package representing the scenario being built. The script stored on the hard drive is an XML file,
- com.wat.rfidsimulator.model.service - the data model classes used by the monitoring module service client,
- com.wat.rfidsimulator.serviceclient - the client module package of the monitoring module service,
- com.wat.rfidsimulator.util - utility class package,
- com.wat.rfidsimulator.view - JavaFX GUI Class and Controller Package,

- Com.wat.rfidsimulator.model (src / test / java section) - a test suite for testing the data model used in the application,
- Com.wat.rfidsimulator.serviceclient (src / test / java section) - a package of classes that test the functionality of the monitoring client service.

5.3 The form of the simulation experiment scenario

The simulation scenario is built into a specialized application window (Figure 5). Its main features include: device definitions (Devices), user profile definitions (Profiles), event definitions, definitions of generators (nodes) and name and description.

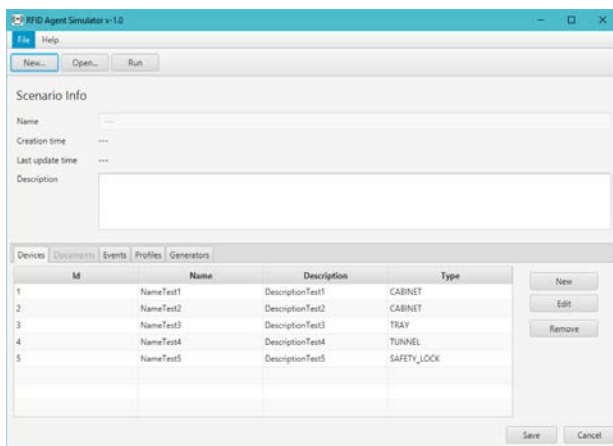


Fig. 5. Defining a new scenario

Correctly defining scenarios including used documents, devices and user profiles allow the application engine to function properly in the event design.

5.4 Running the simulator

Based on the scenario you created earlier, select the "RUN" button in the main application window. Then specify which script file to load. After this, the window will appear as shown in Figure 6. In it we can start the simulation, stop it or stop it.

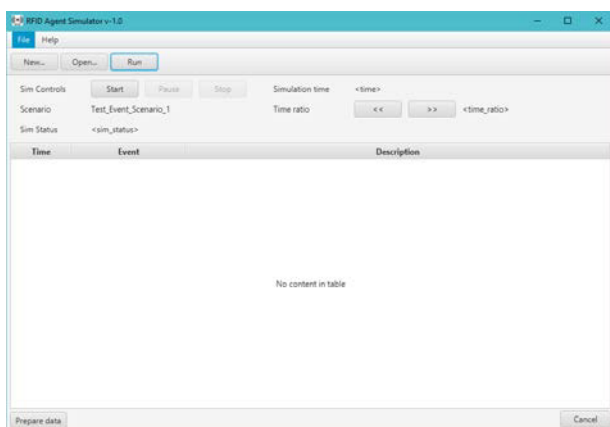


Fig. 6 Running a new scenario

In addition, you can control the rate of elapse of simulation time with special controls. Before you run the simulation, you may need to update your RFID agent data. This is done by a special button that initiates the required data based on a defined scenario. During the normal operation of the simulator, the events generated by the nodes (generators) are stored in the event log (Figure 7). In addition, they are obviously sent to the RFID agent service.

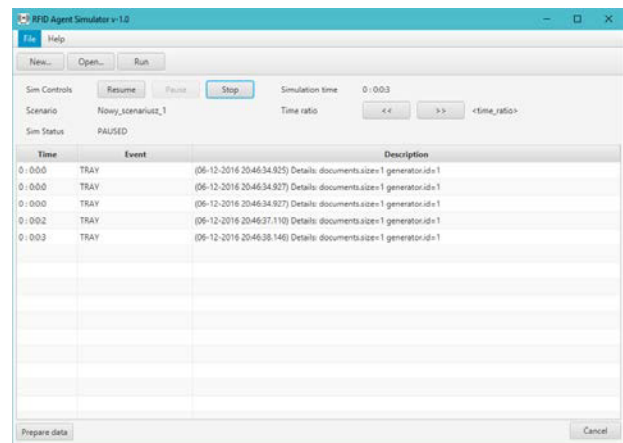


Fig. 7 Simulator activity - event log

After the simulation is complete, you can preview the report of the experiment (Figure 8).

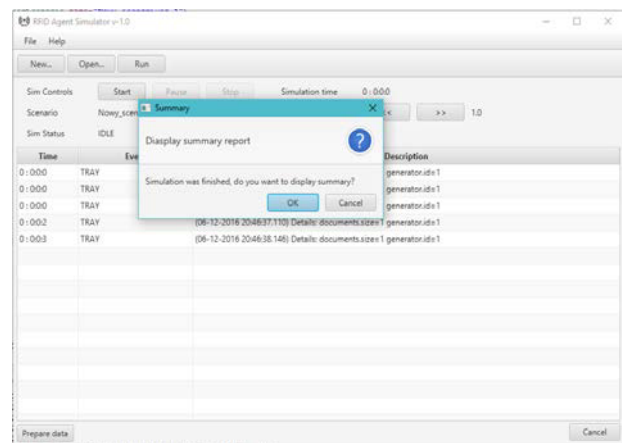


Fig. 8 End of experiment

6 Design of simulation mechanisms

Practical implementation of Java in the first of the two presented approaches in discrete simulation, the package is implemented using recognized libraries responsible for sending messages. The main classes of the package are presented in a simplified class diagram (Figure 9). Each simulation object must inherit from the abstract BasicSimEntity class. In its set of attributes, the values of state variables are stored. Events $e = \langle t, feS \rangle$ associated with state changes are created in objects inherited from the generic BasicSimStateChange class. The main methods of this class are: corresponding to feS

state change function, which removes the non-execution event, giving the current simulation time. Attributes are control parameters: the scheduled time and priority of

the status change (runSimTime, priority), and the optional time step (repetitionPeriod) - for a deterministic model with a fixed step.

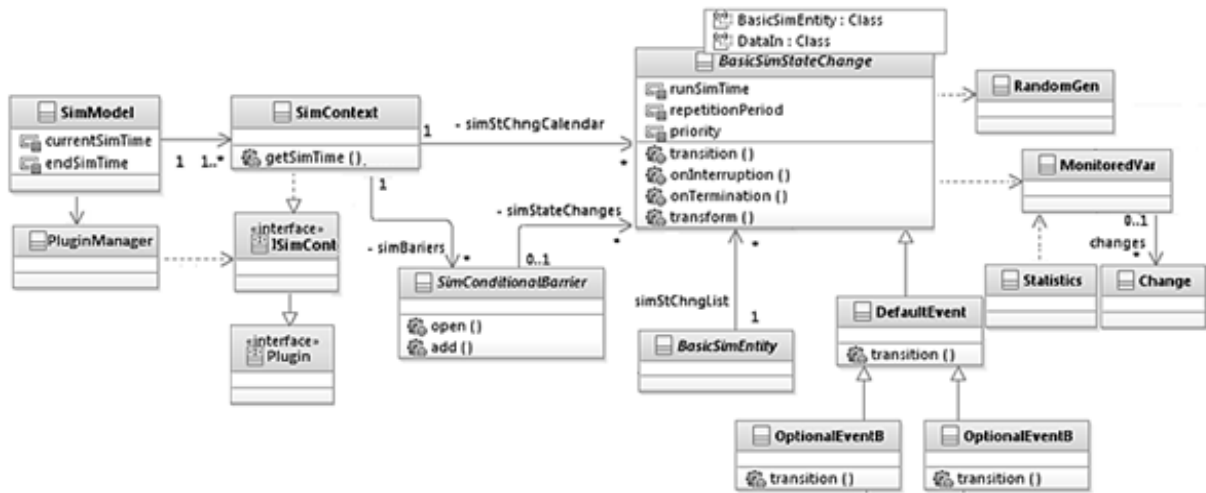


Fig. 9. Simulation classes

Depending on the constructor used, an object of one event is invoked from types: for changing the state that is scheduled only for the calendar (for runSimTime for a moment), for changing the planned state in the calendar. The accepted event broadcasting model enables you to inform all objects that are interested in a specific change in the state of the event (eg to create dependent events). The structure of the event calendar ordered by value runSimTime and priority is composed in the context object (SimContext). The package allows you to integrate many independently developed contexts modeling your various parts of the actual system into one simulation program. Each context must implement plugin interfaces (Plugin, ISimCont). Communication between objects derived from different contexts is based on a common message broker located there in the management object (SimModel). The management object, called overhead in context of the singleton pattern, is fully responsible for: start control, stop and pause in the experiment, time elapsing mode (as fast as possible, astronomical, astronomical scaling), network communication and persistent storage and recovery of states for given moments Simulation. Also, it keeps information about all contexts (so indirectly about objects and event calendars), indicates the specific event to be performed at each simulation time, and determines on this basis the current value of the simulation time that applies to all contexts and objects. Supplementing the basic functions of the package are the classes responsible for: Generating random numbers and pseudo-random processes according to given schedules (RandomGen), monitoring changes and gathering the values accepted by the indicated state variables in different simulations (MonitoredVar) and calculating different statistics on the time series monitors (Statistics). The main simulator software packages are shown in Figure 10.

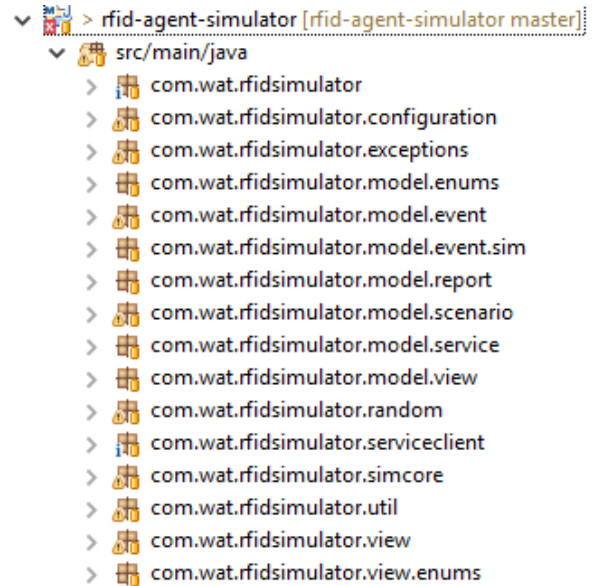


Fig. 10. Simulator application packages

Packages collect classes that perform specific and close actions. Programmable random number generators are based on computational formulas. The subsequent values generated by the generator are derived from the previous ones. The first value is called the generator seed. It can be initialized by the user (grain number) or by the system (according to the current date and astronomical time). Initialization by a grain number ensures the repeatability of the designated numbers in the unchanged initial data. It is said then about generators of pseudorandom numbers. On the other hand, initialization by date and time ensures the uniqueness of generated strings of numbers. Particularly important are generators of uniform distribution numbers, since the numbers they define are the basis for methods of generating

pseudorandom numbers with any distribution. The most commonly used are linear generators based on linear algorithms. Values of random variables with other distributions are generated by the methods of inversion of the distribution, elimination and superposition of distributions.

7 Conclusions

The basic property of the proposed solution is a simulated methodology of RFID document management, document access control and copy control [28, 29, 30, 31], as well as the management of authorized people access.

An additional advantage of the designed solution, besides ensuring the process of verifying the correctness of the office system software with documents with RFID tags, will ensure the possibility of rapid testing. Implemented and undoubtedly innovative methodology for simulating the reliability and performance of computer systems with complex software can also be used in other cases of this type.

References

1. E.M.Abu-Taieh, A.A.R. El Sheikh, *Handbook of research on discrete event simulation environments: Technologies and applications*, IGI Global, Hershey, New York (2010)
2. F.W.C. Araújo Filho, X.L. Travassos, P.S. Figueiredo, *Use of the RFID technology to overcome inefficiencies in the production process: an analysis of a microcomputer company in Ilhéus – Bahia*, Journal of Information Systems and Technology Management, Vol. 11, No. 1, pp. 65–84 (2014)
3. R.E. Barlow, F.Proshan, *Mathematical Theory of Reliability*, John Wiley & Sons, New York, London, Sydney (1965)
4. R.E. Barlow, F.Proshan, *Statistical Theory of Reliability and Life Testing Probability Models*, Holt, Rinehart and Winston Inc. (1975)
5. D. Bobrowski, *Modele i metody matematyczne teorii niezawodności*, WNT, Warszawa (1985)
6. I. Bose, E.W.T. Ngai, T.S.H. Teo, S. Spiekermann, *Managing RFID projects in organizations*, European Journal of Information Systems, 18, pp. 534–540 (2009)
7. B. Dodson, B. Nolan, *Reliability Engineering Handbook*, Marcel Dekker Inc. (1999)
8. G.S. Fishman, *Discrete event simulation. Modeling, programming and analysis*, Springer, New York (2001)
9. F. Grabowski, J.J. ażwiński, *Metody bayesowskie w niezawodności i diagnostyce*, WKŁ, Warszawa (2001)
10. C. Heinrich, *RFID and Beyond: Growing Your Business Through Real World Awareness* Wiley Publishing, Indianapolis (2005)
11. J. Kannry, S. Emro, M. Blount, M. Ebling, *Small-scale Testing of RFID in a Hospital Setting: RFID as Bed Trigger*, AMIA 2007 Symposium Proceedings Archive, pp. 384–388 (2007)
12. M. Kiedrowicz (ed.), *Zarządzanie informacjami wrażliwymi. Bezpieczeństwo dokumentów, wykorzystanie technologii RFID*, WAT, Warszawa (2016)
13. M. Kiedrowicz (ed.), *Zarządzanie informacjami wrażliwymi. Wybrane aspekty organizacyjne, prawne i techniczne ochrony informacji niejawnych RFID*, WAT, Warszawa (2015)
14. B. Kopociński, *Zarys teorii odnowy i niezawodności*, PWN, Warszawa (1973)
15. M.R. Lyu (ed.), *Handbook of Software Reliability Engineering*, McGraw-Hill (1996)
16. Y.K. Malaiya, M.N. Li, J.M. Bieman, R. Karcich, *Software reliability growth with test coverage*, IEEE Transactions on Reliability, Vol.51, No.4, 420–426 (2002)
17. T. Maniva, H. Sugano, M. Kato, *Mass Data Read/Write Technology for UHF-Band RFID Tags*, Fujitsu Sci. Tech. J., 43(4), 464–468 (2007)
18. J. Musa, *Software Reliability Engineering: More Reliable Software, Faster and Cheaper*, AuthorHouse, Bloomington (2004)
19. Pfefferman J.D., Cernuschi-Frias B., *A non-parametric non-stationary procedure for failure prediction*, IEEE Transactions on Reliability, Vol.51, No.4, 434–442 (2002)
20. H. Perros, *Computer simulation techniques – the definitive introduction*, North Carolina State University, Raleigh (2009)
21. H. Pham, *System Software Reliability*, Springer (2006)
22. M. Rausand, A. Hoyland, *System Reliability Theory: Models, Statistical Methods, and Applications*, Wiley Series in Probability and Statistics, Wiley-Interscience (2003)
23. S. M. Ross, *Simulation*, Elsevier Inc. (2006)
24. B. Sinclair, *Simulation of Computer Systems and Computer Networks: A Process-Oriented Approach*, University Press, Cambridge UK (2004)
25. Sołowjew A.D., *Analityczne metody w teorii niezawodności*, WNT, Warszawa 1983
26. C.T. Stambaugh, F.W. Carpenter, *Wireless innovation in inventory monitoring and accounting*, Strategic Finance, Vol. 91(6), 35–40 (2009)
27. S. Whang, *Timing of RFID Adoption in a Supply Chain*, Management Science, Vol. 56, No. 2, 343–355 (2010)
28. M. Kiedrowicz, T. Nowicki, R. Waszkowski, Z. Wesołowski, K. Worwa, *Optimization of the Document Placement in the RFID Cabinet*, EDP Science, MATEC Web of Conferences 76 02001, (2016), DOI:10.1051/mateconf/20167602001

29. M. Kiedrowicz, T. Nowicki, R. Waszkowski, Z. Wesołowski, K. Worwa, *Method for assessing software reliability of the document management system using the RFID technology*, MATEC Web Conf., 76 (2016) 04009, DOI: <https://doi.org/10.1051/mateconf/20167604009>
30. M. Kiedrowicz, T. Nowicki, R. Waszkowski, Z. Wesołowski, K. Worwa, *Software simulator for property investigation of document management system with RFID tags*, MATEC Web Conf., 76 (2016) 04012, DOI: <https://doi.org/10.1051/mateconf/20167604012>
31. R. Waszkowski, A. Chodowska, M. Kiedrowicz, T. Nowicki, Z. Wesołowski, K. Worwa, *Data flow between RFID devices in a modern restricted access administrative office*, MATEC Web Conf., 76 (2016) 04004, DOI: <https://doi.org/10.1051/mateconf/20167604004>
32. M. Kiedrowicz, T. Nowicki, R. Waszkowski, Z. Wesołowski, and K. Worwa, *Business processes in the RFID-equipped restricted access administrative office*, MATEC Web Conf., 76 (2016) 04003, DOI: <https://doi.org/10.1051/mateconf/20167604003>.
33. M. Kiedrowicz, *Location with the use of the RFID and GPS technologies - opportunities and threats*, GIS ODYSSEY 2016, pp. 122-128, (2016).
34. M. Kiedrowicz, T. Nowicki, R. Waszkowski, *Business process data flow between automated and human tasks*, 3rd International Conference on Social Science (ICSS 2016) December 9–11 2016, pp. 471-477, (2016).