Using Quantitative Methods as Support for Audit of the Distributed Informatics Systems

Marius POPA[1], Sergiu CAPISIZU[2]
[1]Department of Computer Science in Economics,
Academy of Economic Studies, Bucharest, Romania
[2]Bucharest Bar Association
marius.popa@ase.ro, capisizu@ew.ro

This paper highlights some issues regarding how an indicators system must be developed and used in an audit process. Distributed systems are presented from de points of view of their main properties, architectures, applications, software quality characteristics and the scope of audit process in such systems. The audit process is defined in accordance to standard ISO 19011 and the main characteristics of this process are highlighted. Before using quantitative methods in audit processes, the framework in which the indicators are built must be defined. There are presented types of indicators used in audit process and classes of measurement scale. An audit process is carried out on different levels and support indicators must be in accordance to audit object. The paper presents some requirements of the indicators depending on the level of audit.

Keywords: *Quantitative Methods, Audit Process, Distributed Informatics System*

1 Audit Process of the Distributed Informatics Systems

An informatics system uses automatic methods and means to collect, transmit, store and process data for information capitalization in the management process of the organization [9].

The informatics system resides in all the informational flows and circuits and all the methods, techniques used to process the data necessary to the decision system. The informatics system is the middle layer between the decision system and informational system. The communication between these layers is made in all possible directions. Also, this layer records, processes and transmits the information from the operational system to the decision one [9].

There is no single definition of a distributed system. Thus, a distributed informatics system is a component of the informational system. This kind of informatics system collects, processes, transmits, stores and presents data by using computing systems. Also, it is responsible for automatic processing of the data by using various methods and techniques [11]. An informatics system is called distributed because its components are placed in different logical and physical locations.

In [6], the distributed informatics system is defined as a set of hardware and software components interconnected in networks, the organizational and administrative framework in which these components are working.

A distributed system consists of multiple autonomous devices linked by communication channels. In a distributed informatics system, a device is commonly given by a computer, and the communication channel is a computer network infrastructure. Through the communication channel, devices interact with each other to reach a common goal. Usually, the goal is a large computational problem supposing use of data and processing facilities in different physical and geographical locations. The interaction between computers is made through message passing.

Some applications of the distributed systems and software are illustrated below [15]:

- *Telecommunication networks*:
  - Telephone networks and cellular networks;
  - Computer networks and Internet;
  - Wireless sensor networks;
  - Routing algorithms;
- *Network applications*:
  - World wide web and peer-to-peer networks;
  - Multiplayer online games and virtual reality;
  - Distributed databases and distributed databases management systems;
  - Network file systems;
  - Distributed information processing systems: banking systems, airline reservation systems and so forth;

- *Real-time process control*:
  - Aircraft control systems;
  - Industrial control systems;
- *Parallel computation*:
  - Scientific computing: cluster computing, grid computing;
  - Distributed rendering in computer graphics.

There are various architectures used for distributed computing, as it follows [15]:

- *Client-Server* – the tasks are divided between server which provide services and client which request service to the server; often servers and clients operate over a computer network and they are on separate hardware; clients initiate communication sessions with servers; servers listen to requests from clients;
- *3-tier architecture* – user interface, functional process logic which implements the business rules, computer data storage and data access are independent modules most often on separate platforms; the three tiers are:
  - Presentation – displays information related to services;
  - Application – controls the functionality of the application by performing detailed processing;
  - Data – consists of Database Servers where data are stored and retrieved; scalability and performance of the data are improved by giving this separate tier. The 3-tier architecture is depicted in figure 1.

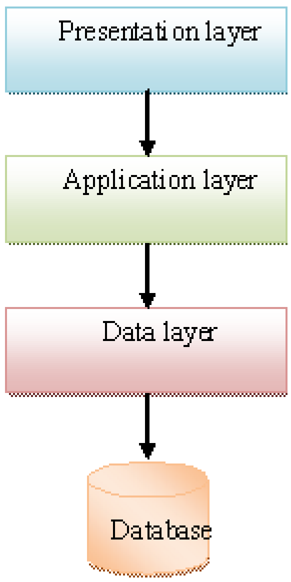


Fig. 1. 3-tier architecture

- *N-tier architecture* – provides a model to build flexible and reusable applications; developers can add or modify layers instead to rewrite the entire application;
- *Distributed objects* – represents software modules working together; they reside in multiple computers connected by computer networks or in different processes running on the same computer; the objects communicate through messages sent in a remote machine or process; some examples of frameworks for distributed objects are: Objective-C, Java RMI, CORBA, DCOM, DDObjects, JavaSpaces, Pyro, Distributed Ruby;
- *Loose coupling* – occurs when a dependent class a pointer only to an interface; provides extensibility to designs because a new concrete class can be added later without requiring modification and recompilation of the class;
- *Tight coupling* – relates to computer cluster and how tightly-coupled the individual nodes are; a computer cluster is a group of linked computers working together so that they form a single computer;
- *Peer-to-peer* – is any distributed architecture composed of computer with parts of their resources directly available to other computers; the resources are processing power, disk storage or network bandwidth;
- *Service-oriented architecture* – is based on web services; a web service makes a reusable component available and accessible across the web.

The characteristic set of a distributed informatics system includes the following:

- *Resource sharing* – sharing of hardware and software resources;
- *Openness* – equipments and software from different vendors are used in distributed system;
- *Concurrency* – concurrent processing is allowed to improve the performance;
- *Scalability* –the size or volume of a distributed system component or the number of users can be increased in such way not to interrupt the function of applications or components;
- *Fault tolerance* – ability of the system to continue its operation after a fault has occurred.

The software quality characteristic represents all the attributes of the software products used to describe and evaluate the product quality.

Regarding the software of a distributed

informatics system, in accordance to ISO 9126 the quality characteristics are:

- *Functionality* – capability to provide functions which meet the users' declared and implicit needs; the following sub-characteristics are included: compatibility, accuracy, interoperability, completeness, security;
- *Reliability* – capability to maintain a specific level of performance under specified conditions; sub-characteristics of the reliability: maturity, fault tolerance, recovering;
- *Usability* – capability to understand, learn and use a software product which has to be attractive for the users; the following sub-characteristics are include: understanding, learning, operability;
- *Efficiency* – capability to provide appropriate performance relative to the amount of used resources; sub-characteristics of efficiency: time saving, resource saving;
- *Maintainability* – capability of a software product to be modified: corrections, improvements, adaptation to the changes in environment, in requirements and functional specifications; the following sub-characteristics are included: correction ability, expandability, testing;
- *Portability* – capability of a software product to work in various environments; sub-characteristics of this software quality characteristic: hardware and software platform independence, instability, reusability.

In accordance with [12], the concept of *audit* means a systematic, independent and documented process to obtain audit evidences and their examination with impartiality to establish the degree in which the audit criteria are met. The examination is made by persons having specific qualifications who are independent in relation to audit process.

The audit process is developed for more purposes [2]:

- *Initial point* – audit process is the initial point to develop a management system;
- *Compliance/Noncompliance* – audit process establish the compliance of the management system to the requirements;
- *Efficacy* – audit process establish the efficacy of the management system in relation to objectives of organization;
- *Critical points* – the audited organization can identify its vulnerabilities and critical point and it can improve the system;
- *Measure applying* – audit process leads to measure applying to prevent and correct the system and to follow up the applying procedures.

Depending on audit purpose, it can identify the following audit classes [2]:

- Audits for establishing the situation in a time point;
- Audits for accreditation;
- Audits for certification.

The audit scope classifies the audits in the following classes [2]:

- Audit of the management system;
- Audit of the process;
- Audit of the product or service.

Documents used for audit records must be safely stored to be easily found and readable and they must meet the integrity characteristic to conclude correct and objective opinions.

The distributed informatics systems became very popular for the most part of the organizations and government departments due to evolution of the ICT technologies and business globalization.

The audit processes are lead by persons with high level of professional competencies and skills. They follow the standards, guidelines, procedures and legal requirements to assess distributed informatics systems. Standards impose a rigorous way to organize and carry out the audit processes on stages with precise delimitations between audit stages.

To cover the requirements of the audited organization, the auditing team examines the following ICT areas, as it is shown in [8]:

- *ICT strategy* – level of alignment between business and IT strategies;
- *ICT organizing* – compliance of the ICT organizing to supports all processes and systems deemed critical;
- *Application management* – management and maintaining of the application systems;
- *Service management* – internal management of the services, quality parameters assumed by ICT department to deliver services;
- *Data and database management* – management and maintaining the data and databases;
- *Computer network management* – management and maintaining the computer networks and communication systems;
- *Hardware and workstation management* – management and maintaining the servers,

mainframes and operating systems;
- *Computer operation management* – planning and logging the operational activities in data centers and other data processing facilities;
- *Security management* – management and maintaining the physical and logical access to the ICT resources;
- *Business continuity management* – process of planning, maintaining and improvement of the security procedures to continue the service delivery within organization;
- *Asset management* – inventorying, management, configuring and maintaining the ICT assets, including the systems, applications, data and infrastructure components;
- *Change management* – changes in ICT architecture to assure compatibility, feasibility, planning, correct and timely implementation of the proposed modifications;
- *Solution development and implementation* – process of analyzing, designing, development, configuring, testing, acceptance and release of the ICT solutions.

The distributed informatics systems are complex constructions. In audit process for such systems, it must be involved very good specialists as members of the audit team or experts whose opinions are essential to gathered the proofs for statement the conclusions in audit report.

The goal of the audit process carried out for distributed informatics systems is to establish the degree in which this one is in accordance to the requirements stated by the target group, standard specifications or to assess its performance. The audit process can be used as management tool to make decisions regarding the implementation or changing of the business strategies.

The audit process is a qualitative one carried out by specialists who must meet some requirements regarding their knowledge, skills, experience and behavior. Characteristics of the audit process and distributed informatics system can be evaluated through indicators to highlight a numerical way to state opinions regarding the quality of the process and informatics system.

2 Issues Regarding the Building of an Assessment Framework

Premises and characteristics for development of an assessment system are presented in [8] and [11].

An assessment system is based on indicators. Indicators allow to create criteria, requirements, standards and to compare characteristics and results across components of distributed informatics systems.

The assessment system reduces the degree of subjectivity because the indicators offer an objective way to quantify characteristics or processes. Some problems regarding the assessment system aim the correctness, reliability, testing and validation of the mathematical models of the indicators and the way in which the input data are gathered and prepared must be attenuated.

An indicator represents a value on a scale of measurement derived from data series.

The scales of measurement are classified in the following classes [7]:
- *Nominal scale* – classification is the lowest level of measurement; elements are sorted in categories in accordance to a certain attribute; also, the elements are jointly exhaustive and mutually exclusive; jointly exhaustive means that all categories cover the all possible categories of the attribute; mutually exclusive means that an element is classified into one and only one category; there is no assumption about relationships among categories;
- *Ordinal scale* – elements can be compared in order; more that nominal scale, the ordinal scale group elements in categories and order the categories; only relational operators are allowed to be applied and not the arithmetic ones;
- *Interval scale* – differences between measurement points are indicated; the arithmetic operators can be applied on this measurement scale; this kind of scale requires a well-defined unit of measurement agreed as common standard and that is repeatable;
- *Ratio scale* – it is the highest level of measurement; all operations are allowed to be applied on it; it represents an interval scale with an absolute or non-arbitrary zero point; measurement is expresses in both integer and non-integer data.

The measurement scales are hierarchical. A measurement scale has all the properties of the lower ones.

Depending on the number of audit processes, indicators are classified in [11]:
- *Primary indicators* – they aim the primary characteristics of the distributed informatics system; they are computed in a single audit process within department/ module/

component;
- *Aggregated indicators* – they result from more audit processes, applications or aggregation operations of the primary indicators.

Depending on their applying scope, the indicators of an audit process are grouped in [8]:
- Indicators of audit process quality – they are used evaluate the quality level of the audit process: audit program developed within organization, further performing of the audit program;
- Indicators of auditing object – they are used to get data about the object of the control: elements of the control that can be quantified; as effect, the gathered data are in accordance with reality and the audit team has objective data to state its conclusions.

Also, in [8], the following quality characteristics of an assessment system are presented:
- Indicators are correctly developed in accordance with requirements of a such process;
- Indicators do not use important quantities of resources: human, time and financial.

Depending on measurement scale, the indicators are classified in the following classes [8]:
- *Qualitative* – the measurement scale has discreet units as very good, good, satisfactory, poor, very poor;
- *Quantitative* – the assessment is numeric and very precise.

In COBIT, the indicators are defined on the following levels [13]:
- How to measure them the business expects from IT;
- How to measure the IT processes that support IT's objectives;
- How to measure the needs insight the process to achieve the required performance.

The two types of indicators defined in COBIT 4.1 are [13]:
- *Outcome measures* – indicate whether the goals have been met;
- *Performance indicators* – indicate whether goals are likely to be met.

In relation to the reaching of the organization goals, the following three classes of indicators are identified in [5]:
- *Success indicators* – they are used to establish if the goals are met;
- *Progress indicators* – they are used for tracking the execution of tasks;
- *Analysis indicators* – they assist the analyzing the outputs of the tasks.

Development and implementation of an assessment system must take into account the following requirements as it shows in [5]:
- Identifying the indicators based on a methodology;
- Specifying the goal of the assessment system;
- Indicator traceability back to the goals;
- Clear understanding of the type and purpose of each indicator;
- Small start point for assessment;
- Indicators for detecting the trends and hidden tradeoffs;
- Customizing the indicator template;
- Use of definition checklist;
- Dissemination of the unambiguous information;
- Privacy issues of the indicators;
- Respecting the needs of involved people;
- Identifying the adequate solutions available if there is no consensus;
- Using of pilot implementation;
- Planning some assessment on short term;
- Maximizing the relevant information and minimizing the collection effort;
- Testing of the assumptions;
- Taking into account the unintended consequences and the perspectives of different stakeholders.

A template for defining and documenting an indicator is provided in [5], having the following fields:
- Precise objective of the indicator – purpose of the indicator;
- Inputs – data used in indicator applying;
- Algorithms – combining data;
- Assumptions – business environment, business processes and so forth;
- Data collection information – description of how, when, how often and by whom data are to be collected;
- Data reporting information – responsibilities to report data;
- Analysis and interpretation of result – meaning of the different values for indicator.

A methodology to identify indicators is presented in [5]. The steps of the methodology are:

Step.1 Identifying the goals;
Step.2 Identifying what it wants to know;
Step.3 Identifying the sub-goals;
Step.4 Identifying the entities and attributes;
Step.5 Formalizing the measurement goals;
Step.6 Identifying the measurement questions and indicators;

Step.7 Identifying the data elements;
Step.8 Defining and documenting measures and indicators;
Step.9 Identifying the actions needed to implement measures;
Step.10 Preparing a plan.

The building process of an assessment system based on indicators must take into consideration the following characteristics of the indicators during their development process, as it is shown in [13]:

- A high insight-to-effort ratio;
- Comparable internally;
- Comparable externally;
- Good metrics vs. low-quality metrics;
- Easy to measure.

Indicators are used to extract meaningful information from gathered data of the analyzed process or system. There are some basic measures used in software quality assessment. There are also sophisticated statistical techniques and methodologies for data analysis.

The basic measures are [7]:

- *Ratio* – it results from dividing one quantity from another; numerator and denominator are mutually exclusive; the model of an ratio is:

$$R = \frac{N}{D} \times 100$$

A value of R less than 100 means that D is greater than N; otherwise D < N.

- *Proportion* – numerator is a part of the denominator; it can be used for multiple categories of one group; the model of this kind of measure is:

$$P = \frac{a}{a + b + ... + z}$$

where:

$$\frac{a}{a+b+...+z} + \frac{b}{a+b+...+z} + ... + \frac{z}{a+b+...+z} = 1$$

Proportion can be also referred as relative frequency if numerator and denominator are integers and they mean counts of certain phenomena.

- *Percentage* – it is a proportion with denominator normalized to 100; percent means per hundred; it is used when it is known the total number of cases; to use percentage, it must be large enough number of cases; when the number of cases is small then absolute numbers are used because percentage is not stable for a small total; the model of percentage is:

$$P_{\%} = \frac{a}{a + b + ... + z} \times 100$$

- *Rate* – it highlights the changes of the analyzed phenomena in their evolution; rate has the following analytical model:

$$R_K = \frac{X}{Y} \times K$$

The rate highlights measure of change in Y of another quantity X on which Y depends. Quantity K is a constant.

The two most important issues of measurement quality are reliability and validity [7].

An indicator has a high level of reliability if repeated measurements are highly consistent or even identical. If the variations of repeated measurements are large, then reliability is low.

The goal for building an assessment system is to achieve indicators having high reliability. The reliability is established on the following analytical model:

$$I_V = \frac{SD}{M}$$

where:

- I_V is index of variation;
- SD is standard deviation of the repeated measurements;
- M is mean.

A smaller I_V means a great reliability of the indicator.

Validity of an indicator highlights that the measurements are accurate to the analyzed process or system. There are the following kinds of validity:

- *Construct validity* – it refers to the validity representing the theoretical construct;
- *Criterion-related validity* – it is referred to as predictive validity;
- *Content validity* – it refers to covering the range of meanings by an indicator.

Indicators are affected by errors during applying of an assessment system. There are two types of errors [7]:

- *Systematic error* – it is associated to the validity;
- *Random error* – it is associated to the reliability;

In a general case:

$$V = T + s + r$$

where:

- V is the value of measured indicator;
- T is the true value of measured indicator;
- s is systematic error;
- r is random error.

An assessment system used in audit processes offers quantitative and objective information very useful for auditor's team to extract valuable opinions that are accurate to the audited product, process or system.

3 Using Quantitative Methods in Audit Process

Implemented controls in an audit process are based on measures. Measures offer to the auditor's team information to be compared with the standards or management requirements. After comparison, extracted information is used to write down the audit conclusions and recommendations in the final report.
Not all controls can be driven by indicator implementation. There are controls where the experience, competence and skills of the auditor are critical to extract accurate information regarding the object of the investigation.
The controls that aim the technical issues of the distributed informatics system can be supported by an indicator system to achieve accurate information in a fast and rigorous ways. Also, indicators are used to extract valuable information about the quality of the audit process and the activities carried out by audit team.

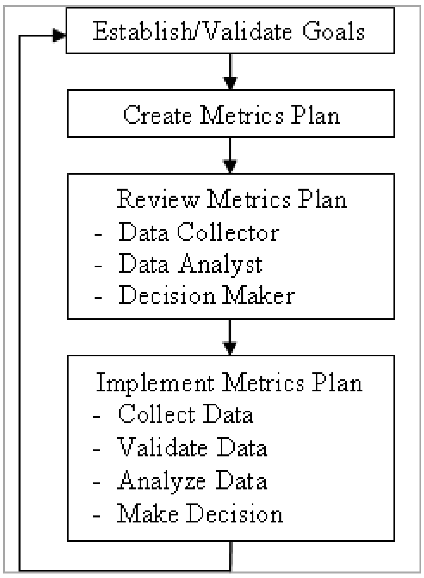


Fig. 2. Indicators process model

Implementation of an assessment system means applying the indicators in accordance to the indicators plan. The indicators plan is a component of an indicators process model. An example of indicators plan is offered in [1] and it is summarized in figure 2.
Implementing the indicators plan aims the following issues [1]:

- *Policy and process issues* – the indicator plan details the way in which data are collected, people can collect and analyze data; the initial effort to gather and quantify information is considerable, but it becomes a regular task after that;
- *Indicator utilization* – an indicator plan is very good if it is correctly implemented and it is used to make decisions on measurements resulted from indicators applying; the audit team must be aware about the importance of the indicators system to achieve accurate and valuable information about the audit object and to make decisions appropriate to the audited system state;
- *Reviews* – they assure the effectiveness of the indicators plan; there is an initial review to highlight the way in which indicators are used to make decisions; the following reviews aim the system and audit process.

Initial, applying a new or improved assessment system in audit process is made on pilot implementation. Pilot implementation tests the feasibility of the assessment system and the operational issues as [5]:

- Forms for collecting and recording data;
- Data storage and access tools;
- People who collect, store and access data;
- Tools used to collect and analyze data;
- Roll up procedures;
- Training.

Input data for indicators system can be collected and organized by automatic tools. Automatic tools reduce de time consumption regarding the data preparing for applying the indicators. Selection of automatic tools must minimize de collecting effort and maximize the yield of relevant information.
An example of indicators implementation process is depicted in [3] and it has the following stages:

1. Create or update indicators – the following activities are included:
   – Goals statement;
   – Data sources and elements are identified;
   – Analyzing relations between goals and data elements;
   – Building indicators;
2. Collect data – it aims data gathering from available date sources;
3. Store data – it aims documenting and storing data in appropriate repository;
4. Analyze and compile data – it includes the following activities:
   – Collected data are analyzed;

– Compiling and aggregation into indicators;
– Data interpretation;
– Causes of findings are identifying;

5. Report indicators – the following activities are included:

– Indicators are documented in appropriate report formats;
– Indicators are reported to stakeholders;

6. Use indicators – it aims the following activities:

– Decision making;
– Resource allocation;
– Improvements are prioritized;
– Communication to executives and external stakeholders.

A continuous improvement process is carried out for indicators system. This process includes the following activities [3]:

- Indicators updating to keep their relevance to the assessment process;
- Training the staff who implements the indicators system.

Indicators system can be used in an audit process of a distributed informatics system on the following levels:

- Controls of the distributed informatics system in operation;
- Controls of the development life cycle for distributed informatics system;
- Controls of the audit process quality.

Using indicators in controls of distributed informatics systems is made in accordance to the audit criteria. The audit criteria can be established by executives or they can be specified in a standard of ICT field.

Thus, indicators have particular models depending on audit object that it is evaluated on audit criteria specified in standards.

Some indicators to control the secure development of a software product are [14]:

- Defect density – average occurrence of programming faults per lines of code;
- Lines of code – count of executable lines of code; it quantifies the size of code;
- Function point – estimation of software size by measuring functionality;
- Risk density – using of risk in defect density;
- Cyclomatic complexity – it takes into consideration the structure of the code; complex code means higher risk of defects;
- Inspection rate – estimation of the time to perform code review; it is computed as lines of code per unit of time;
- Defect detection rate – it computes as number of defects found per unit of time;
- Code coverage – proportion of the reviewed code;
- Defect correction rate – it is computed as number of defects corrected per unit of time; the effort to correct the defects can be estimated;
- Re-inspection defect rate – after correction, some defects still exists or new defects appear; defects after correction are highlighted on this indicator.

If the assessment system aims the evaluation of audit process quality, the following perspectives of process measurement must be considered [4]:

- Performance – it is highlighted by measuring the results of the audit process and by measuring the attributes of the process itself; examples: ease of understanding and reliability of the audit opinions, expended effort in audit process, time necessary to perform a task, characteristics and quantities of resources, numbers, types and sources of audit proofs and so forth;
- Stability – lack of variation for the characteristics of the audit processes; it is determined by measuring the attributes of the process and tracking the results over time;
- Compliance – the following measurable aspects of compliance can affect the process performance:
  – Ability to execute process – establishing the ability of the organization to carry out the process;
  – Use of defined process – execution of the process as it was defined;
  – Oversight, benchmarking and assessment of the process – reviews of the process status, formal assessment and benchmarking;
- Capability – the process is stable and conforming to requirements;
- Improvement and investment – identifying the changes and benefits that justify the costs needed to improve a process; to improve a process it is needed training, additional equipment, new automatic tools and so forth.

Indicators used during the audit process are an effective tool to extract valuable information regarding the process, product or service that is audited. Based on indicators, the audit team formulates objective opinions regarding the state of audit object. A very important requirement is that audit team to use the indicators system and to know how this system must be used to get objective measures as base of conclusions formulated in the audit report.

4 Conclusions

Quantitative methods are very important means to support an audit process. They extract valuable information regarding evaluation of processes, products and services through audit processes. It gets objective measures to support the audit proofs used as base of evidences in the audit report.

Distributed informatics systems are very complex constructions as architectures, scope, technologies used in their development, functionalities, involved resources and so forth. The audit process of this kind of systems is very difficult as their complexity. An assessment system based on indicators can make easier the work of the audit team. Also, it can be used automatic tools to implement the indicators.

Indicators implementing must meet the requirements of the mathematical models associated to indicators. A good indicators system is useless if they are not use in accordance to the specifications elaborated during indicators building. The indicators building process must identify the requirements, limits, scope and other elements that can influence the measurement quality when the indicators are applied.

Acknowledgement

The paper was elaborated within the research project with code 1838/2008, contract no. 923/2009 and the title *Implementation of the Quantitative Methods in Distributed Informatics System Audit*, financed by The National University Research Council – Ministry of Education, Research and Innovation from Romania.

References

[1] T. Augustine and C. Schroeder, “An Effective Metrics Process Model”, *The Journal of Defense Software Engineering*, Vol. 12, No. 6, 1999, pp. 4 – 7.

[2] F. Baicu and A. M. Baicu, *Auditul şi securitatea sistemelor informatice*, Victor Printing House, Bucharest, 2003.

[3] N. Bartol and B. A. Hamilton, *Practical Measurement Framework for Software Assurance and Information Security*, National Institute of Standards and Technology Software Assurance (SwA) Measurement Working Group, 2008.

[4] W. Florac, R. Park and A. Carleton, *Practical Software Measurement: Measuring for Process Management and Improvement*, GUIDEBOOK CMU/SEI-97-HB-003, 1997.

[5] W. Goethert and W. Hayes, *Experiences in Implementing Measurement Programs*, Software Engineering Measurement and Analysis Initiative, Carnegie Mellon University, Technical Note, November 2001.

[6] I. Ivan, G. Noşca and S. Capisizu, *Auditul sistemelor informatice*, ASE Printing House, Bucharest, 2005.

[7] S. Kan, *Metrics and Models in Software Quality Engineering, Second Edition*, Addison Wesley, 2002.

[8] M. Popa, “Requirements for Development of an Assessment System for ICT Security Audit,” *2nd International Conference on Security for Information Technology and Communication (SECIT&C 2009), Journal of Information Technology & Communication Security*, Academy of Economic Studies, Bucharest, 19 – 20 November 2009, ASE Publishing House, Bucharest, Romania, pp. 221 – 230.

[9] M. Popa, “Detection of the Security Vulnerabilities in Web Applications,” *Informatica Economică*, No. 1(49), 2009, pp. 127 – 136.

[10] M. Popa, C. Toma and C. Amancei, “Characteristics of the Audit Processes for Distributed Informatics Systems,” *Informatica Economică*, Vol. 13, No. 3, 2009, pp. 165 – 178.

[11] M. Popa and A. Paraschiv, “Premises for Development of an Assessment System for Security Audit of Distributed Information System”, in *Proceedings of the Ninth International Conference on Informatics in Economy – Section 7: Informatics Security*, May 7 – 8, 2009, ASE Publishing House, Bucharest, Romania, pp. 827 – 832.

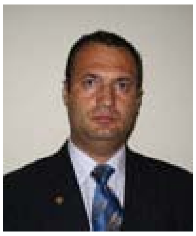
[12] ASRO, SR EN ISO 190011, *Ghid pentru auditarea sistemelor de management al calității şi/sau de mediu*, July 2003.

[13] IT Governance Institute, *COBIT 4.1*, 2007.

[14] http://www.owasp.org/index.php/OWASP_Code_Review_Guide_Table_of_Contents

[15] http://en.wikipedia.org/wiki/Distributed_computing

Marius POPA has graduated the Faculty of Cybernetics, Statistics and Economic Informatics in 2002. He holds a PhD diploma in Economic Cybernetics and Statistics. He joined the staff of Academy of Economic Studies in 2002. Currently, he is university lecturer in Economic Informatics field and branches within Department of Computer Science in Economics at Faculty of Cybernetics, Statistics and Economic Informatics from Academy of Economic Studies. He is the author and co-author of 6 books and over 100 articles in journals and proceedings of national and international conferences, symposiums, workshops in the fields of data quality, software quality, informatics security, collaborative information systems, IT project management, software engineering. From 2009, he is a member of the editorial team for the *Informatica Economica* journal.



Sergiu CAPISIZU has graduated the Faculty of Cybernetics, Statistics and Economic Informatics in 1997 and National University of Defense in 2005. He holds a PhD diploma in Economic Cybernetics and Statistics, having the title *Models and techniques to perform the economic information audit*. He is co-author of books and articles in information audit and ICT fields. Also, he has published articles in proceedings of national and international conferences, symposiums, workshops in the fields of data quality, software quality, information audit and juridical aspects in ICT field. He is evaluator of ANEVAR association.