The Modeling of the ERP Systems within Parallel Calculus

Loredana MOCEAN, PhD
Babes – Bolyai University of Cluj – Napoca
Business Information Systems Department
loredana.mocean@econ.ubbcluj.ro

As we know from a few years, the basic characteristics of ERP systems are: modular-design, central common database, integration of the modules, data transfer between modules done automatically, complex systems and flexible configuration. Because this, is obviously a parallel approach to design and implement them within parallel algorithms, parallel calculus and distributed databases. This paper aims to support these assertions and provide a model, in summary, what could be an ERP system based on parallel computing and algorithms.
Keywords: *ERP Systems, Modeling, Parallel Calculus, Incremental Model*

1 Introduction

Modeling is a method used in science and technique and consists in schematically reproduction of an object or system as a similar system or analogue with the scope of studying the properties and transformation of the original system.
The incremental model which is possible in ERP design is shown in Figure 1 [1].

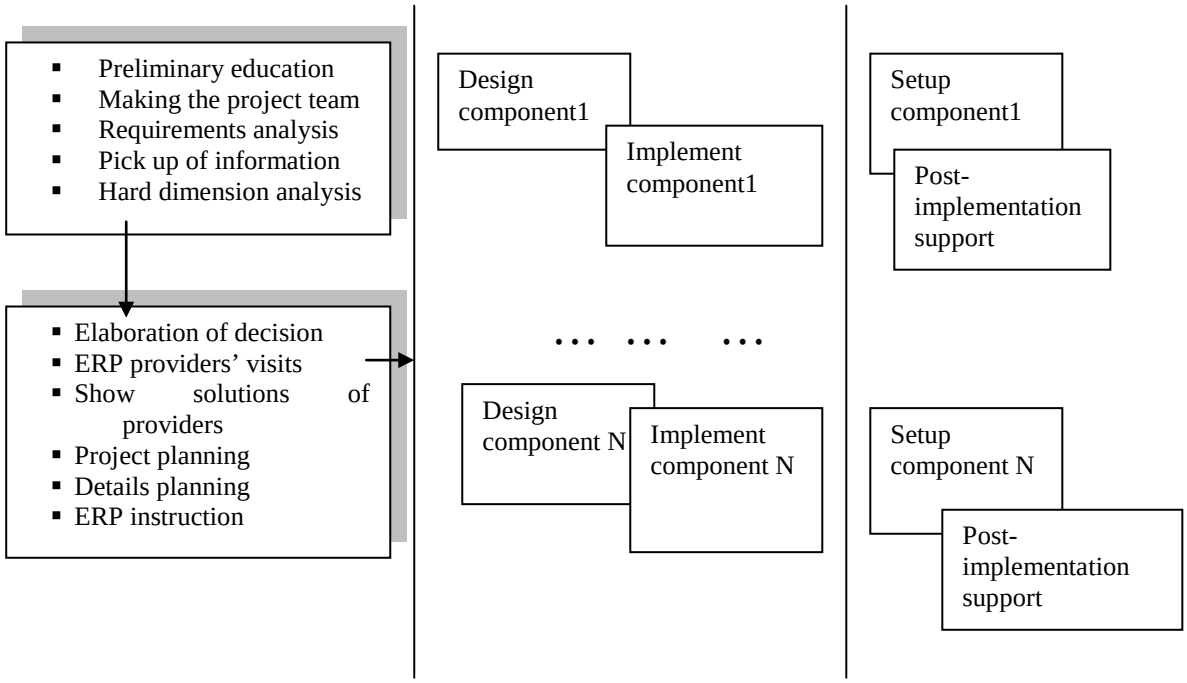

Fig. 1. The incremental model implemented on ERP systems

One of the first methods of producing parallel programs consisted of parallelizing sequential programs with the biggest amount of scientific calculations.
All loops express an intrinsic parallelism which can be exploited by dividing it into several simultaneous activities. As we see in Figure 1, it is very efficient to divide the programming work in parallel design of the components and parallel programming of the modules.
A second possibility is to design the best parallel version from scratch. It will be considered whether it is a parallelism of data or of control [2].

2 Purpose of the study

This study will be a result of important researches to help determine the answers to important questions about ERP systems, modeling, parallel calculus and distributed databases because currently, there is no effective approach to modeling a prototype of ERP system based on parallel calculus or parallel programming.

The purpose of this study is helpful for all the companies (commerce, programming, manufactures etc.) that desire to implement or upgrade the ERP basis system, to change the algorithms of implementation from linear algorithms to parallel algorithms, from simple databases to distributed databases, from linear computing to parallel computing.

3 Related work

"Enterprise Resource Planning (ERP) systems involve the purchase of pre-written software modules from third party suppliers, rather than bespoke (i.e. specially tailored) production of software requirements, and are often described as a buy rather than build approach to information systems development", tell us the authors of the paper [3]. It was a very important paper for us to think to the possibility of designing ERP systems in parallel mode.

D. Robey et al. in paper [4] present a comparative case study of 13 industrial firms that had implemented an enterprise resource planning (ERP) systems. The algorithms which had been used were linear.

In paper [5], the authors tell us" *Making changes to ERP software code (called modification) is usually strongly discouraged by vendors and implementation consultants"*. That's why we try to see the implementation of a ERP system by mean of parallel programming and parallel calculus.

Chen in paper [6] presents us briefly which are the secrets of a successfully implementation of a ERP system, the top management of each company is interested in these new information technology and philosophy and identify new opportunities in programming of such systems.

The business benefits of Enterprise Resource Planning (ERP) systems are presented also in paper [7]. We think that a parallel implementation of an algorithm and using a distributed database, will be a real benefit for all the companies.

"Over the past few years, firms around the world have implemented enterprise resource planning (ERP) systems to have a standardized information system (IS) in their respective organizations and to reengineer their business processes" [8]. But most of them implemented a classical ERP, with classic database and a unique administration of algorithms.

If we try to implement an ERP within a parallel algorithm and parallel calculus, we must work in a complicate way but the satisfaction of the clients and the usability of the system will be for us a starting point to continue our research.

4 The structure of an ERP system using parallel algorithms and parallel calculus

"An enterprise resource planning (ERP) system is an attempt to create an integrated product that manages the majority of operations in a company. What is different about ERP systems is that they integrate across functions to create a single, unified system rather than a group of separate, insular applications." [10]

The characteristics of ERP systems:

- modular-design;
- central common database;
- modules are integrated, data transfer between modules is done automatically;
- they can be configured and adapted to the needs of business customers;
- are complex systems, flexible and offers best practices;
- most ERP systems can be accessed via the Internet.

Integrating databases with the computer network has created a more powerful, explosive technology, whose development was done in steps of evolution. In applying parallel algorithms, we think that any ERP system could be structured as in Figure 2.

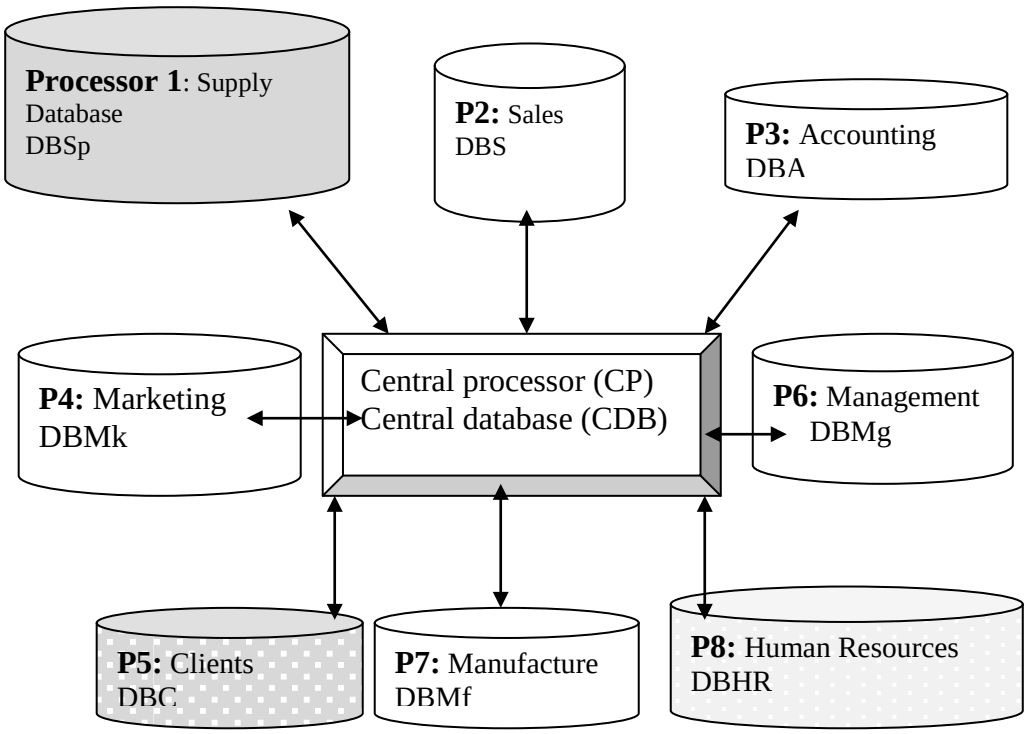

Fig. 2. A structure of an ERP system within parallel calculus with the help of P1 (processor 1), P2, P3 etc.

When we think to design a parallel ERP system, we must take in mind several important issues related to parallel processing:

- Construction of efficient parallel computers;
- Design of efficient parallel algorithms;
- The existence of metrics for performance evaluation of parallel algorithms;
- Programming languages for parallel processing;
- Parallel programming tools;
- The development of portable programs [11].

In the process of modernization, often firms require to modernize their applications less effective or even replace them with more efficient, which ensures that real-time and efficient decisions and therefore the management process.

Currently, we consider that the implementation of integrated ERP systems with parallel algorithms and parallel programming is able to help improve the efficiency of firms.

By making such systems company offering the possibility to integrate the new system to existing applications with new applications specific to the operating company.

Next we focus on the Client module. In this way we will work with multiple tables, between them exits relationship.

Tables used in the database are:

a. Table Clients, with the following structure (see table 1):

Table 1. The structure of Clients table

Filed	Type	Properties
Code	AutoNumber	Long Integer, primary key
Type	L	0 for individual, 1 for companies
Name	Text	50
IdentificationCode	Text	15 (CNP for individuals, CUI for companies)
IdentificationKey	Text	10 (No of CI)
Adress	Text	50
ZipCode	Number	10
IBAN	Text	24

Bank	Text	30

b. Table Documents, with the following structure (see table 2):

Table 2. The structure of Documents table

Field	Type	Properties
IdDocument	AutoNumber	Long Integer
Date	Date/Time	Medium Date
TypeDocument	Number	Integer
CodeClient	Number	Long Integer
CodeEmployee	Number	Long Integer
IdProduct	Number	Long Integer
Quantity	Number	Long Integer
Price	Number	Long Integer

c. Table Products, with the following structure (see table 3):

Table 3. The structure of Documents table

Field	Type	Properties
IdProduct	AutoNumber	Long Integer
Name	Text	50
Characteristics	Text	50
InitialStoc	Number	Integer
Price	Number	Long Integer

At the level of serial processing we propose three queries that we rely on.

1. The query *Sales Products*

```
SELECT Products.IdProducts,
Products.Name,
Products.Characteristics,
Documents.TypeDocument
FROM Products INNER JOIN Documents
ON Products.Idproduct =
Documents.Idproduct
GROUP BY Products.IdProducts,
Products.Name,
Products.Characteristics,
Documents.TypeDocument
HAVING
(((Documents.TypeDocument)="0"));
```

2. The query *Output Quantity*

```
SELECT Products.IdProducts,
Products.Name, Documents.Type,
Documents.Date,
Sum(Documents!Quantity) AS Canti
INTO Output
FROM Produse INNER JOIN Documents
ON Products.IdProducts =
Documents.IdProduct
GROUP BY Products.IdProduct,
Products.Name, Documents.Type,
Documents.Date
HAVING (((Documents.Type)="0") AND
((Documents.Date)<=Date()));
```

3. The query *Input Quantity*

```
SELECT Products.IdProducts,
Products.Name, Documents.Type,
Documents.Date,
Sum(Documents!Quantity) AS Canti
INTO Input
FROM Produse INNER JOIN Documents
ON Products.IdProducts =
Documents.IdProduct
GROUP BY Products.IdProduct,
Products.Name, Documents.Type,
Documents.Date
HAVING (((Documents.Type)="1") AND
((Documents.Date)<=Date()));
```

A database query can be run in parallel to increase the performance. The nature of the task, the database configuration, and the hardware equipment,, all determine the design for performing a task in parallel.

These considerations are interrelated, and should be considered together when we work on the physical and logical design of a database.

There are two types of query parallelism: inter-query parallelism and intra-query parallelism [12].

Inter-query parallelism refers to the ability of the database to accept queries from multiple applications at the same time. Each query runs independently of the others.

Intra-query parallelism refers to the simultaneous processing of parts of a single query, using either intra-partition parallelism, inter-partition parallelism, or both.

At Inter-query parallelism level, we propose the development of three applications:

1. *At the level of processor P5 and P8*

The proposed application will deal mainly with employees; will include Human resources department and Accounting. The main query is determining sales per employee.

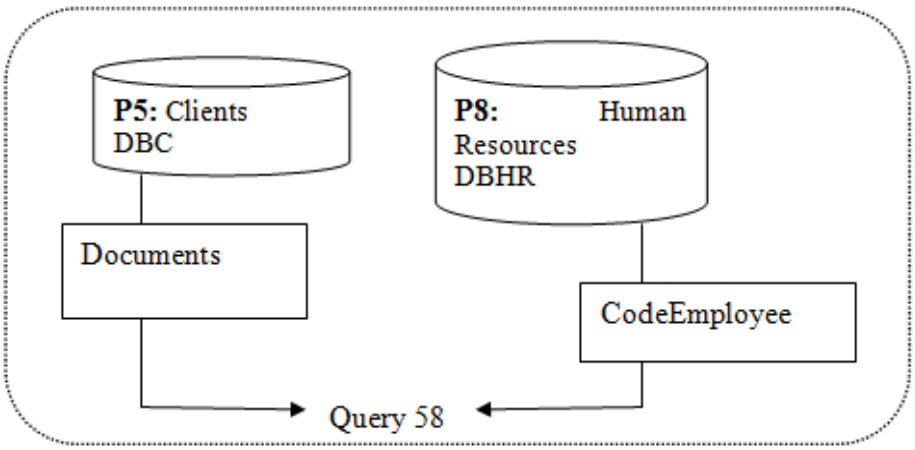

Fig. 3. A first query between DBC and DBHR

2. *At the level of processor P4 and P8*

Proposed application will handle the marketing services provided by this department, the number of clients made by each employee of the department of marketing and sales for each of them.

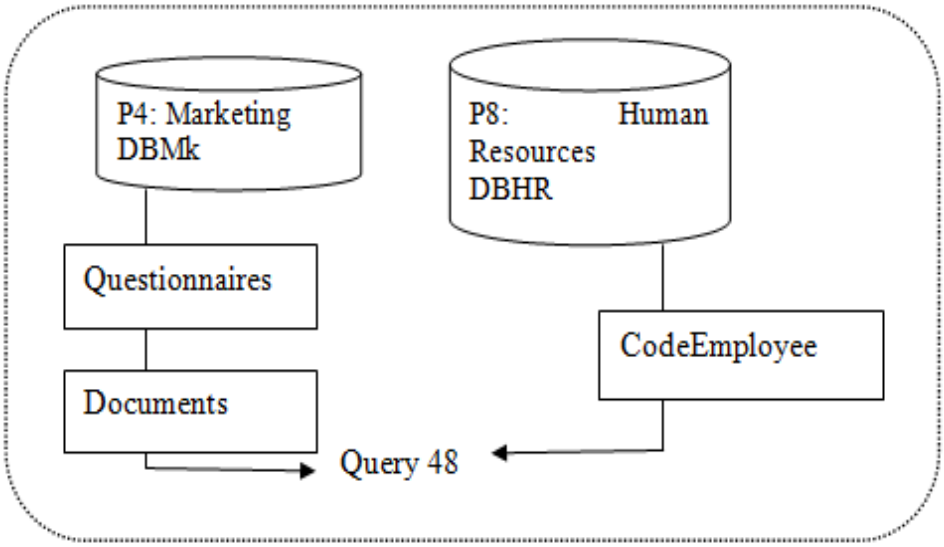

Fig. 4. A second query between DBMk and DBHR

3. *At the level of processor P2, P7 and P8*

The application will try to determine total sales of production achieved.

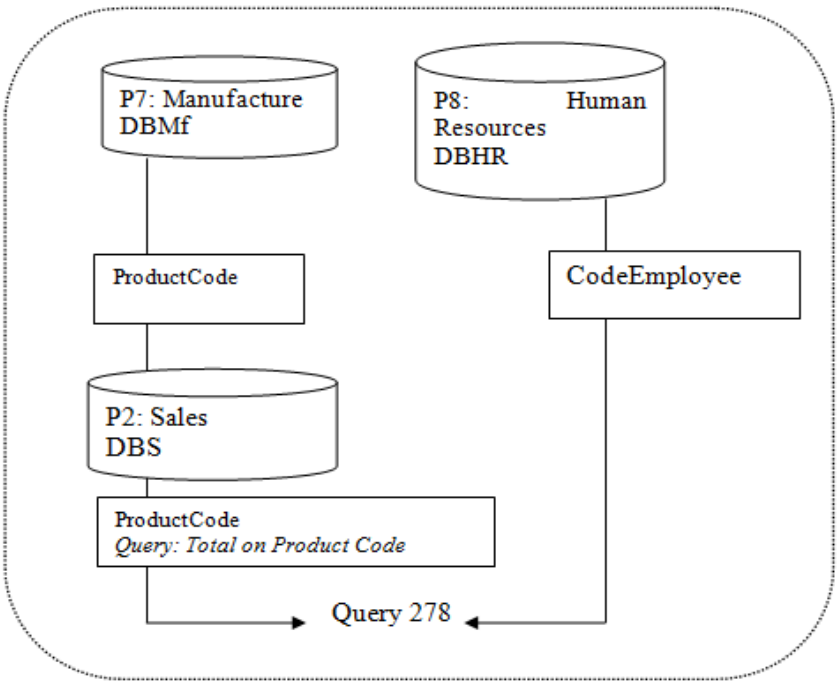

Fig. 5. A third complex query between DBMf, DBS and DBHR

At Intra-query parallelism level we aim to achieve the following queries:

1. The first query determines the products sold, input quantity, output quantity for each product. At serial processing level, the three queries have been previously described. At the level of parallel processing we will propose a parallel implementation of the three queries and finally the aggregation of the results in a new selection query.

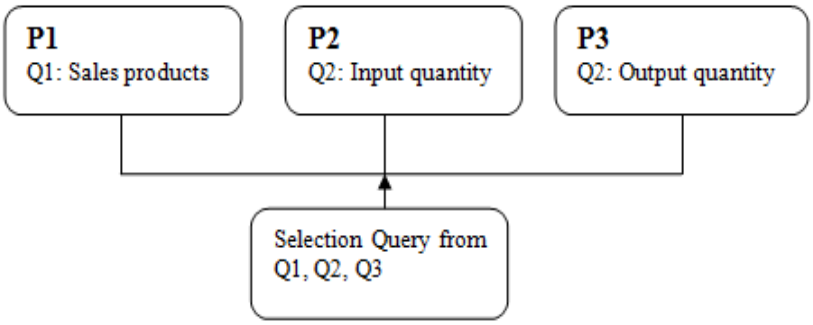

Fig. 6. An Intra-Query

2. The second query determines the customer and the employees issued number of documents issued for each document.

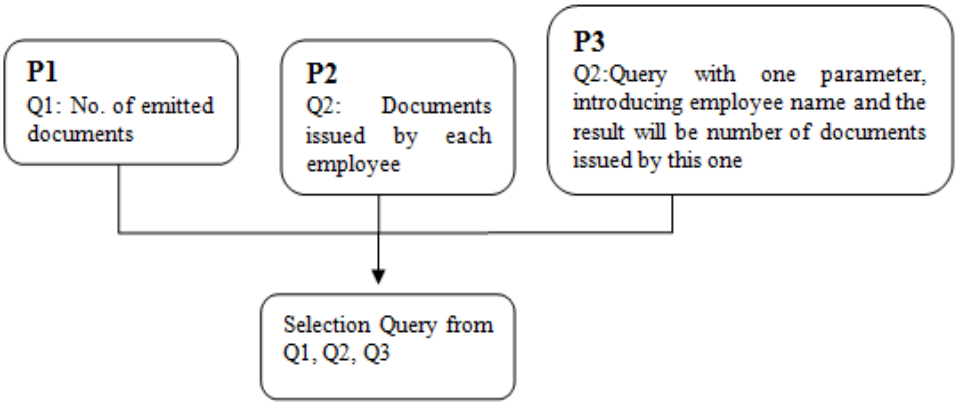

Fig. 7. An intra-query within three processors

3. The third query with two parameters run in this mode: the user enters the date as the first parameter, then the second parameter is client code, finally resulting all invoices to a customer at a certain date.

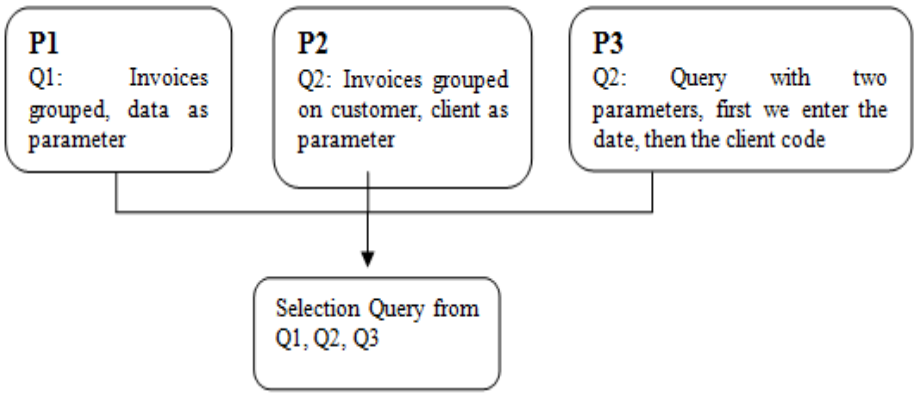

Fig. 8. An intra-query within three processors

5 Conclusions

This paper aims to provide some solutions to the implementation of parallel computing and parallel algorithms in studying and implementing ERP systems, as well as theoretical researchers seeking solutions in these areas.

The paper starts with some previous modeling results ERP systems wishing to research breast enlargement using parallelism.

In the paper are given some examples of implementation of inter-query queries and intra-query, simple, selection query and parameters query. The implementing of detailed system will be the subject of further publication.

References

[1] L. Mocean, M. Ciaca, "About Modeling The Erp Systems", *Studia OEconomica Journal*, No. 54, Issue 1, 2009, pp.30-38.

[2] L. Mocean, R. Buchmann, "Research On A Parallel Algorithm Implementation Strategy In Group Of Firms", in *the proceedings of The Tenth International Conference on Informatics in Economy IE 2011*, Bucuresti, 2011.

[3] W. Skok, M. Legge, "Evaluating Enterprise Resource Planning (ERP) Systems using an Interpretive Approach", in *SIGCPR '01 Proceedings of the 2001 ACM SIGCPR conference on Computer personnel research*, Available: http://portal.acm.org.

[4] D. Robey, J.W. Ross, M.C. Boudreau, "Learning to Implement Enterprise Systems: An Exploratory Study of the

Dialectics of Change", in *MIT Center for Information Systems Research*, June 13, 2000.
[5] L. Brehm, A. Heinzl, M. Lynne Markus, "Tailoring ERP Systems: A Spectrum of Choices and their Implications", Available: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=927130.
[6] I.C. Chen, "Planning for ERP systems, analysis and future trend", Available: http://cis.csuohio.edu/~ichen/ERP.pdf.
[7] S. Shang, P.B. Seddon, "A Comprehensive Framework for Classifying the Benefits of ERP Systems" , in *AMCIS 2000 Proceedings*, Available: http://aisel.aisnet.org/amcis2000/39/.
[8] F. Calisir, F. Calisir, "The relation of interface usability characteristics, perceived usefulness, and perceived ease of use to end-user satisfaction with enterprise resource planning (ERP) systems", Available: http://www.sciencedirect.com/science/article/pii/S0747563203000748.
[9] A.N. Parr et al., "Identification of Necessary Factors for Successful Implementation of ERP Systems", in *Proceedings of the IFIP TC8 WG8.2 International Working Conference on New Information Technologies in Organizational Processes: Field Studies and Theoretical Reflections on the Future of Work*, Available: http://portal.acm.org/citation.cfm?id=722079.
[10] T. Herzog, "A Comparison of Open Source ERP Systems", Vienna, 2006, Available: http://www.big.tuwien.ac.at/system/theses/20/papers.pdf?1298476232.
[11] G. Dodescu et al., *Procesare paralelă*, Ed. Economică, 2002.
[12] http://www.ibm.com/support/docview.wss?rs=71&uid=swg21370360.
[13] D.A. Sitar-Tăut, *Baze de date distribuite*, Ed. Risoprint, 2005 .
[14] I.Dziţac, *Sisteme informatice integrate*, Universitatea Agora, 2009.
[15] J. Chang, H. Ma, *Modeling the Architecture for Component-Based E-commerce System*, Springer-Verlag GmbH, ISSN 0302-9743, vol. 2495 / 2002.
[16] T.H. Davenport, "Putting the enterprise into the enterprise system", *Harvard Business Review*, nr. 1, 2001.
[17] D. Fotache, L. Hurbean, *Solutii informatice integrate pentru gestiunea afacerilor*, Ed. Economica, 2004.
[18] A. Cockburn, *Agile Software Development*, Addison-Wesley, 2001.
[19] R. Conradi, B. Westfechtel, "Version models for software configuration management", *ACM Computing Survey*, vol.30, no2, June 1998.
[20] A.I. Salah, "Engineering an Academic Program in Software Engineering Mills".
[21] D.Harlan, J.R. Newman, C.B. Engle, *An Undergraduate Curriculum in Software Engineering*, 2002.
[22] D. Budgen, P. Brereton, B.Kitchenham, *Stephen Linkman Realizing Evidence-based Software Engineering*, 2004.
[23] M. Pecht, *Product Reliability, Maintainability and Supportability*, Handbook, 1995.

Loredana MOCEAN has graduated Babes-Bolyai University of Cluj-Napoca, the Faculty of Computer Science in 1993, she holds a PhD diploma in Economics from 2003 and she had gone through didactic position of assistant and lecturer, since 2000 when she joined the staff of the Babes-Bolyai University of Cluj-Napoca, Faculty of Economics and Business Administration. She is the author of more than 10 books and over 35 journal articles in the field of Databases, Data mining, Web Ser-vices, Web Ontology, ERP Systems and much more.