



Journal of
**Software
Engineering**

ISSN 1819-4311



Academic
Journals Inc.

www.academicjournals.com

ERP Integration: A Reuse Based Approach, Evaluation and Prospect

¹Mohamed Ramzi Bouzid, ^{1,2}Naoufel Kraiem and ¹Henda Ben Ghezala

¹RIADI Laboratory, ENSI, University of Manouba, Tunisia

²Department of Computer Science, College of Science, Sultan Qaboos University, P.O. Box 36, Al-Khoudh 123, Muscat, Oman

Corresponding Author: Mohamed Ramzi Bouzid, RIADI Laboratory, ENSI, University of Manouba, Tunisia

ABSTRACT

One of the critical points in an Enterprise Resource Planning (ERP) integration project is matching the ERP functionalities with the organizations' requirements, so as to define how to adapt the system and/or the organization to reach an efficient functioning. A number of approaches have been elaborated to solve the issues posed by this matching activity in particular one of the conceptual mismatch between different kinds of used models. Moreover, one aspect is not often considered in these approaches. It is the opportunity to reuse experience from one ERP implementation project to another. This study presents a reuse-based requirements elicitation approach for ERP integration. The suggested approach has been subsequently tested through a series of experiments aimed to evaluate the efficiency of reusing generic domain knowledge in the matching activity. The case study showed that starting requirements elicitation and matching at the generic domain level is always more efficient than starting it directly at the organizational specific level.

Key words: ERP integration, requirements engineering, reuse process, evaluation

INTRODUCTION

In the domain of requirements specification, several studies (Van Lamsweerde, 2009; Berenbach *et al.*, 2009; Pohl, 2010; Ernst *et al.*, 2014) have shown that implementation failures of computer systems in an enterprise are mainly due to the poor identification and/or comprehension of the business needs that these systems attempt to answer.

In general, from these poorly expressed requirements, the developer realizes a solution that does not fully meets, in whole or in part, with the initial needs desired by stakeholders.

To meet their needs through the integration of a new system, an organization has usually:

- Either develop oneself in-source its own application, which is expected that will allow it, in general, to full fit come as close as possible to the needs expressed by the various stakeholders. This approach is known as the build approach
- Either directly installs an existing solution (e.g., application as service, component, or ERP). This approach is referred to as buy approach as its design and development are out-sourced

When integrating a software solution, one of the major problems facing the buy approach is that the matching between the business need of the organization and the features of out-sourced system. The success of the whole project depends directly on this matching (Esteves and Pastor, 2003; Kolovos *et al.*, 2009; Bellahsene *et al.*, 2011).

The difficulty of the matching between the organizational requirements and the system functionality resides in the difference between the levels of abstraction these two words express. Indeed, while the stakeholders express problems and requirements in their own languages (mostly in plain text), the buy approach expresses a solution in technical concepts. This results in a mutual misunderstanding incomprehension between, on the one hand, the business experts who speak in terms of objectives, goals and strategies to achieve (with a minimum of operational details) and the actors of the operational level who speak in terms of solution and technical details.

To address this matching issue, several studies and researches such as by Davenport (2000), Giunchiglia *et al.* (2007), Chen *et al.* (2011) and Shvaiko and Euzenat (2013) are mainly interested in developing metrics to match user needs with features offered by a system in to meet the conceptual discordance that exists between these two worlds (business and system).

Most of these approaches consider projects mapping as being completely independents, even if it remains in the same activity domain. Thus, these approaches leading to lose the acquired experience and to implement each new project from scratch. Yet, very few approaches are based on this generic aspect of requirements elicitation (Kensche *et al.*, 2009; Schneider *et al.*, 2011; Hull *et al.*, 2011; Mussbacher and Kienzle, 2013).

Moreover, in the case of the Small and Medium-sized Enterprises (SMEs), there is still a real problem because of solution based approach followed. In fact, this type of companies generally can't develop and express their own requirements freely without being influenced by the solution imposed by editors and integrators.

Among these different problems, we are interested, in this study, on the domain reuse because very few ERP implementation methods are able to propose a concrete model of knowledge reuse at the requirements level (Daneva, 2000, 2001; Ettlíe and Kubarek, 2008; Chernak, 2012; Orawski *et al.*, 2013).

For implementing ERP, this study presents a requirements reuse based approach using three levels of requirements modeling: Domain, Enterprise and ERP. This study mainly aims to:

- Take advantage of the generic knowledge to reduce the development costs and optimize the project achievement time by eliminating redundant requirements elicitation effort and specification rework
- Improve the quality of requirements by reusing requirements of a known high quality
- Show the relevance of the use of a unique modeling language to reduce abstraction level mismatch between the business world and the system world
- Show applicability of the approach through a case study that explains its steps
- Evaluate and validate the research hypotheses formulated through this study

STATE OF THE ART

Difficulties of the ERP integration: Different types of problems and difficulties faced by SMEs/SMIs have chosen to integrate ERP systems have been discussed here. It should be noted that it is difficult to define in a simple and exhaustive manner the ERP concept that refers to various realities (Addo-Tenkorang and Helo, 2011).

Davenport (1998) defined it as "A package that promises the transparent integration of all the information in the company: information on finance, human resources, supply chain and customer base".

Rowe (1999) notes that “Technological innovation ERP carries the old dream of a single repository of information system of the company. With such systems, the actors of the company operate daily with a common language. As well the tool responds to the leaders’ desire of real-time control”.

ERP can then be presented as a “Ready to implement” software package (developed by an editor) that can, in general, covers all organization processes allowing it to manage and optimize all of its resources through several adaptable modules.

During ERP integration, the main observed difficulties can be classified in three categories:

- Organizational difficulties, related to the company’s business aspects
- Technical difficulties, related to hardware company
- Human problems, related to human resources

Organizational difficulties: One of the most important steps in the ERP integration process is to ensure, throughout the project, that the system to implement meets the requirements expressed by different users.

Another difficulty, as important and relevant, is that many enterprises, making the choice to implement an ERP, are tempted or forced to align their needs with the ERP functionalities (and not the reverse) to avoid the cost of customization work, often expensive in terms of time and money. These enterprises then choose features that already exist in the ERP that does not exactly meet the need initially expressed and are therefore obliged, first, to remove one or more requirements and secondly, to increase their reliance on ERP, its publisher and its integrator (Malhotra and Temponi, 2010; Addo-Tenkorang and Helo, 2011).

Indeed, it’s usually difficult for an organization to disregard the existing solutions and only focus on its requirements. The organization is therefore forced to align its processes, business objectives and organizational model to those proposed by the solution of Aloini *et al.* (2007) and Hwang and Grant (2011).

In addition to these difficulties faced by most SMEs/SMIs, there are other obstacles that may face companies in an ERP implementation project:

- Lack of involvement and communication (Scott and Vessey, 2002; Romeyer, 2003)
- Complexity of enterprise (Malhotra and Temponi, 2010)
- Level of functional specialization (Brehm *et al.*, 2001)

Technical difficulties: ERP is particularly difficult to implement and a low system control can have a significant impact on the entire project. Indeed, the integrated nature of the ERP and the complexity of the system make the success of the integration work very critical and risky. It’s therefore important to control all ERP components to carry out their integration with other components of the organization information system.

In addition, for enterprises with very specific needs, it is often very difficult for an ERP to support such requirements (Davenport, 1998). The enterprise asks then the adaptation of functionality offered by ERP to take into account its specific processes. These adaptations are made through the customization of the concerned module or through new components integration. This choice is usually very expensive for SMEs/SMIs.

Another alternative that may be adopted by the project manager is that to “Co-exist” the old system that supports the specific requirements of the organization with the new ERP system. This choice reduces the integrated aspect of the ERP and is not technically simple to implement since it requires a delicate task to maintain the overall coherence of information system. This choice is often costly and ultimately ineffective.

In addition to these limitations, other obstacles and difficulties may face the system solution:

- Lack or absence of tools for decision support (Romeyer, 2003)
- Difficulties related to the technical complexity of ERP (Liu and Wu, 2008)
- Difficulties related to database settings and ERP testing (Liu and Wu, 2008)
- Technical incompatibility of software with enterprise specificities (Romeyer, 2003)

Human resources difficulties: Generally, modification or upset of information access and handling, raised after the ERP implementation, causes reluctance and resistance of many actors in the organization that deals with change.

The obstacle facing this kind of project can be also related to the lack of management commitment (Aloini *et al.*, 2007) that does not invest enough to support the project with staff. Laudon and Laudon (2001) also cited the resistance to change usually due to lack of support staff during an ERP implementation.

Indeed, an ERP implementation project transform, in various degrees, existing jobs within the organization (Malhotra and Temponi, 2010), old professions become obsolete and new occupations appear. Other authors have mentioned, through literature, various obstacles related to this human aspect. Among these problems, it was mentioned:

- Lack of responsibilities definition (Schmidt *et al.*, 2001)
- Weak commitment of the project team (Kliem, 2000)
- Difficulties related to power struggles and conflicts of interest (Romeyer, 2003)
- Lack of coordination and/or experience of the project team (Romeyer, 2003)

However, very few requirements engineering approaches are proposed to provide an effective response to these various challenges, especially organizational one. We believe, as some authors (Bush and Finkelstein, 2002; Zoukar *et al.*, 2004; Kolovos *et al.*, 2009; Chen *et al.*, 2011), when implementing an ERP system, remains to match ERP functionalities with various organization requirements.

In this regard, the following focuses on the analysis of the literature concerning the matching between business needs and system functionalities.

Matching process: A quick survey of the literature allows us that several definitions of the matching exist. This concept of matching has been defined by Klein (2001) as “A set of relationships between the elements of two representations (ontologies, database schemas, etc.), indicating a relative similarity according to a given measure”. Other authors, such as Xu and Embley (2003) and Giunchiglia *et al.* (2007) described the matching process as “A process of defining a functions set for specifying correspondences between terms”.

Another definition given by Shvaiko and Euzenat (2005) and Shvaiko and Euzenat (2013) presented the matching process as “A critical operation in various domains. It takes as input two schemas/ontologies and determines as output existing relationships between these entities”.

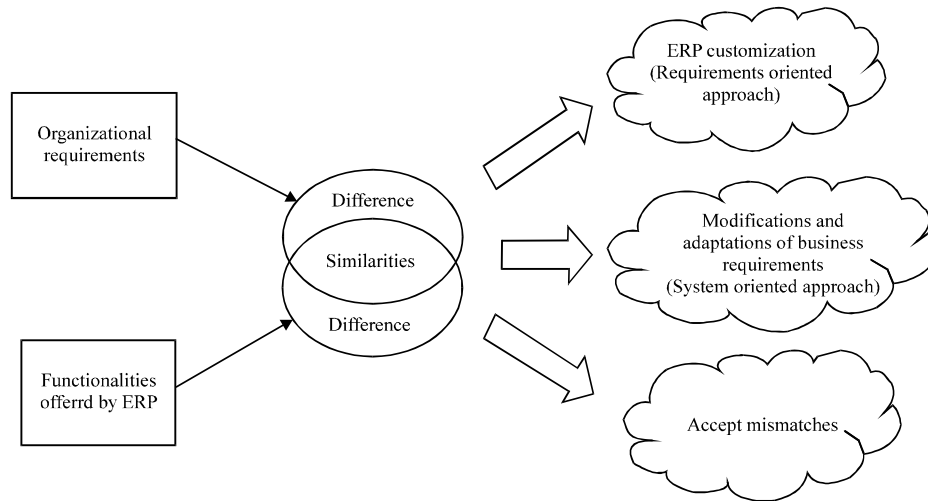


Fig. 1: Alternatives of matching when integrating ERP

The process of “Matching” is done to reduce differences that may exist between two models or two diagrams. In this regard, several solutions have been proposed to provide an answer to the problems of matching (Aumueller *et al.*, 2005; Fagin *et al.*, 2009; Chen *et al.*, 2011; Bellahsene *et al.*, 2011).

Through literature study, it appears that often ERP functionalities don't fully cover the business needs of an organization. These specific requirements that are not met by the system to implement, must then be treated through one of the three following alternatives in order to reduce mismatches (Martin *et al.*, 1999) (Fig. 1):

- Adapt ERP functionalities to organization needs through a specific customization process. This approach is known as “Directed by the business needs of the organization”
- Modify the specific business need to meet system features. This approach is called “Run by the system solution”
- Accept existing mismatches between business processes and system and live with it

MATERIALS AND METHODS

Description and evaluation of the approach: To carry out this study, a literature review relating to the existing approaches was conducted dealing with the problem of mismatch and the difference of abstraction level between business needs and system functionality.

Then, the interest focused on the weaknesses and limitations that can represent this kind of approaches. Among aspects that these approaches do not allow to take into account. The ability is mainly cited to capitalize on the expertise acquired from previous projects to help the requirements specification and the matching process with an ERP.

By following this literature review, various hypotheses were made that could overcome the identified limitations of existing approaches namely the relevance of introducing a domain level allowing the reuse of generic requirements.

Based on these hypotheses, a requirements engineering approach was suggested that aims to assist: (1) The elaboration of the specific needs mode of a well-defined organization and (2) The construction of the model representing the matching of these needs with the system functionalities during an ERP implementation process.

This stage is followed by an empirical study conducted as a series of controlled experiments in order to assess the impact of the suggested approach on different hypotheses that was made through this research study.

Few methods can guide the matching process with a real balance between organizational business needs and functionalities of the system. We think, as several authors and Gulla (2004) and Bernstein *et al.* (2011) that such matching must be made in the intentional level.

MAP formalism: It is necessary to use some sort of formalism that is able to represent the three different levels of the framework. It should be noted that this formalism must already have shown its effectiveness in several domains and more particularly in the ERP implantation project.

It was proposed to use the MAP formalism (Rolland and Prakash, 2000; Nurcan *et al.*, 2005) that include several positive aspects among which:

- The formalism can be understandable by every body (people of business and process)
- It allows refinements of MAP sections (that is not permitted by I* (Alencar *et al.*, 2000)
- It emphasizes variability (that is not taken into account by KAOS (Van Lamsweerde, 2001)
- It removes irrelevant details (by abstraction)

For all above mentioned reasons and in order to better assess the impact of our model, we have chosen to use the MAP in order to provide a unified view on the enterprise requirements as well as on the ERP functionalities.

MAP is a directed graph in which nodes are labeled with goals and edges labeled with strategies. The directed nature of the graph is a way to represent the flow of goals. A goal aims at some situation that an organization wants to reach through one or several business processes. Strategies define approaches, means or manners to achieve goals.

A section is the key element of a map. It is a triplet $\langle G_i, G_j, S_{ij} \rangle$ and represents a way to achieve the target goal G_j from the source goal G_i following the strategy S_{ij} . Each section of the map captures the condition to achieve a goal and the specific manner in which the process associated with the target goal can be performed.

Suggested approach: The framework (Fig. 2) is based on a review of the literature of requirements reuse (Baxter *et al.*, 2008; Wang *et al.*, 2010; Chernak, 2012; Orawski *et al.*, 2013) and requirements elicitation (Berenbach *et al.*, 2009; Pohl, 2010; Ernst *et al.*, 2014).

In this framework, the organization is modeled according to three levels:

- Domain level is related to the common business aspects of associations
- Enterprise level is allowing the extension of these generic business aspects to a specific solution for an association
- ERP level is representing the ERP features

This framework (Bouzid and Kraiem, 2012) is established in order to take in to account the knowledge reuse aspect. In this respect, the suggested approach would initially consist in the matching of the generic requirements (Generic Model GM) with the functionalities offered by the ERP (Solution Model SM).

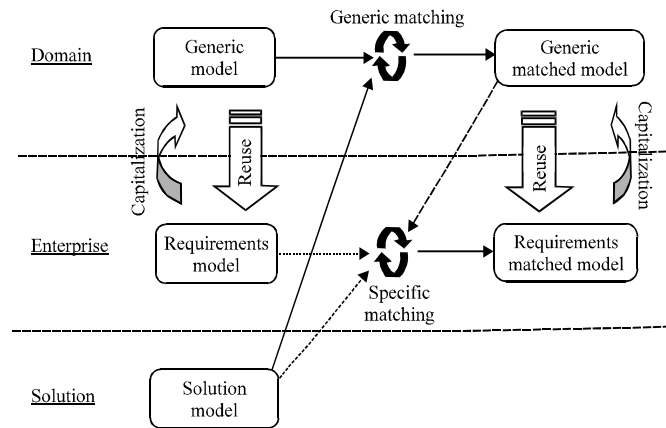


Fig. 2: Suggested approach

At the domain level, the process consists on an analysis of similarities between requirements domain and ERP features. A certain number of information of matching are thus collected (Generic Matched Model GMM) and form a reusable knowledge for the specific analysis.

The next step is to build the specific map of the enterprise (Requirements Model) from the domain map (Generic Model GM). To do so, the first step is to conduct an analysis of similarities and differences between Generic Model and organization’s specific needs. Based on these differences and similarities, the Requirements Model RM can be built by derivation.

Once the generic matching conducted, the specific matching will be considered that consists, in a first step, in identifying the differences and similarities between the domain requirement map and the specific needs of the enterprise. Using the reuse of acquired knowledge in the matching of the domain level, the Requirements Model will be then matched with the SM to obtain the Requirements Matched Model RMM.

It may be seen that the process of matching at the specific level, unlike the generic level is based on the analysis of the variations between domain requirements and specific requirements. It should be recalled, in this regard, that few differences exist between two levels that facilitate the analysis and make it much more reliable. More details for the description and the application of the suggested approach are available in Bouzid (2012).

Evaluation of the approach: In order to perform this approach, an application domain that is original (for ERP), random and that doesn’t require a long phase of learning was adopted.

As case study, a charity association was chosen whose main objectives are to improve life quality of hospitalized children and to reduce their isolation by providing laptops and software. It aims to facilitate communication, education and entertainment for children. It’s the “Docteur Souris” Association (DS) (Bouzid, 2012).

This experienc gave us an opportunity to assess:

- The influence of the use of a domain model (the Domain Model) when building a model of specific requirements (Specific Model)
- During the matching process in the specific level (between the ERP functionality and the specific needs of the enterprise)

Table 1: Experimental protocol

Steps	Groups			
	1	2	3	4
1st	Elaborate the Requirements Model (RM) (Dr. Souris)	Elaborate the Generic Model (GM)	Analyse the provided Generic Model	X
2nd	Elaborate the Generic Model (GM)	Elaborate the Requirements Model (Dr. Souris)	Elaborate the Requirements Model (Dr. Souris)	Elaborate the Requirements Model (Dr. Souris)
3rd	Elaborate the Generic Matched Model (GMM)	Elaborate the Generic Matched Model (GMM)	Analyse the provided Generic Matched Model	X
4th	Elaborate the Requirements Matched Model (RMM)	Elaborate the Requirements Matched Model (RMM)	Elaborate the Requirements Matched Model (RMM)	Elaborate the Requirements Matched Model (RMM)

In this context, these experiments were mainly conducted to evaluate the different hypotheses made by this study. These hypotheses are formulated as follows:

- C1: Reuse the Domain matching in the Enterprise Level to build the RMM
- H0: Reuse has no addition value to matching
- H1: Get RMM faster
- H2: Reduce the loss of knowledge and experience
- H3: RMM more consistent (richness, completeness, same abstraction level)
- C2: Use of domain knowledge to build the Requirements Model (RM)
- H0: Domain knowledge has no addition value to matching
- H4: Increase the rapidity of RM build
- H5: Build more consistent RM (completeness, same abstraction level)

In order to ensure a measurable and objective analysis of our approach, the experimental activities were assigned to work groups according to Table 1.

Measures: To objectively evaluate the effectiveness of different tools used in each of these approaches, we are based on Precision P, Recall R and Effectiveness F measures (Olson and Delen, 2008).

Precision and Recall are two widely used statistical classifications. Precision can be seen as a measure of exactness or fidelity and Recall as a measure of completeness.

A measure that combines Precision and Recall is the Effectiveness $F\beta$ (where β = Importance Recall/Importance Precision). Three commonly used F measures are the F1 measure (recall and precision are evenly weighted), the F2 measure (which weights recall twice as much as precision) and the F0.5 measure (which weights precision twice as much as recall). More details of the experimental protocol are available in Bouzid and Kraiem (2012).

RESULTS

Figure 3 shows, for each of the 4 groups, precis and recall and 3 values of effectiveness F according to different values of Beta ($\beta = 0.5, 1$ and 2).

A quick look on these first results can already make the following findings:

- Group 3 had the best Precision, Recall and F-measures values for goals and sections of the constructed Requirements Matched Model RMM (Group 3 is the group that used the Generic Model of reference to build its various models)

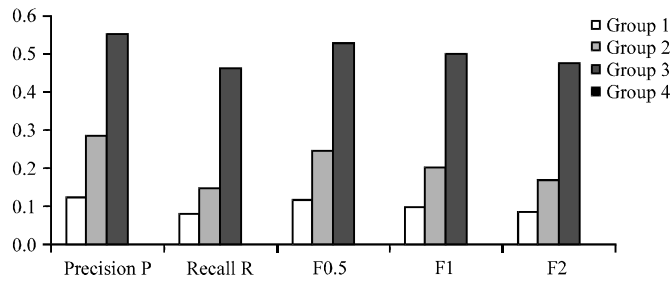


Fig. 3: Sections of proposed RMM/sections of reference RMM

- The results obtained by group 2 (that used its own Generic Model) are in second position in terms of P, R and F. These results are followed by those obtained by Group 1
- Group 4 (that has not any reasoning at the generic level) has been able to identify any strategy of MMR, which explains the null measurements for different values of this group

Alongside each of these tasks, it was said that each different group to draw a graph representing the evolution of his model (in terms of sections number) versus time. The 3rd group was also the fastest to have constructed its model followed respectively by groups 2, 1 and 4.

From these results, it was concluded that reliance on a provided generic model helps to build a more rich and coherent requirements Model than groups who have a completely specific approach (hypothesis H3). It also helps to build RMM faster mainly thanks to learning time that was significantly lower since this group relied on the acquired knowledge from the generic model (hypothesis H1 and H2).

DISCUSSION

After following the experimental protocol and results achieved, it was found that there are minimal differences between the Generic Model GM and the specific Requirements Model RM. This observation allows us to note the relevance of generic requirements to build an organization's specific Requirements Model easier than other approaches starting from scratch (Zoukar *et al.*, 2004; Nurcan *et al.*, 2005). Moreover, it should be noted that by using predefined generic model helps to significantly reduce the learning and the elicitation time. This will allow a faster work of specification, modeling and matching (Bouزيد and Kraiem, 2012).

Despite their relevance, the results could be discussed because of a possible bias introduced by the use of measures P, R and F related to reference RMM that was developed.

To address this potential bias, it was decided to recalculate the values of precision, recall and effectiveness for different groups compared to a new reference model. This "updated" RMM will include new Intentions and Sections Map from RMM models, proposed by different groups, which may logically be added to the initial RMM.

The results obtained from the second evaluation confirm and consolidate those obtained during the first experiment.

Indeed, the group 3 which had the best results in terms of Precision, Recall and efficiency is the group that identifies the most coherent new sections to be included in the updated RMM.

As part of approach, this capitalization process of acquired knowledge enriches the Domain models already developed.

From the obtained results, it was seen that to follow an approach based on the Domain knowledge reuse to identify the specific requirements mainly allows to:

- Build a specific requirements model and a matching model richer and more consistent than a model built “from scratch”: Hypothesis H2, H3, H5 and H6 validated
- Build these models faster than through an approach fully conducted on a specific level: Hypothesis H1 and H4 validated

So, the aspects of a top-down reasoning (generic to specific) and of the domain expertise reuse in addition to the capitalization process mainly reflect the power and relevance of the suggested approach. Moreover, the illustration of the approach by a case study has served to illustrate the applicability of different steps of the suggested approach and to demonstrate its feasibility in a real context.

The results, observations and measurements made have therefore demonstrated, on the one hand, the advantages of the approach for specifying business requirements of an organization and their matching with an ERP system and on the other hand, validated the formulated research hypotheses.

It was noted that our experience has shown its relevance in the evaluation of our approach. However, despite these results, a self-critic made about the followed experimental protocol can raise some limitations:

- The need to extend these controlled experiences to other activity domain in order to make global and definitive conclusions
- The sample size and the number of groups’ persons may be greater
- The experiments were conducted over a limited time

CONCLUSION

The work presented in this study opens the way to different perspectives and can be continued in several directions. These perspectives can be related to two levels: that of continuity of the work done and the enlargement of the research area:

- Introduce some additional aspects to the Map model, it should be noted that there are some aspects that are not supported by the Map formalism and it would be interesting to integrate. Among these aspects, we can mention the identification of the different actors involved and their respective roles in the realization of a goal or in the strategy selection
- Develop new case studies, it turns out to be appropriate to extend the application of our approach to others case studies from various domains of activity. This perspective would aim, firstly, to demonstrate the applicability and feasibility of this approach to new domain (other than associative one) and secondly, to enrich the base of illustrative examples of our approach. This base will form therefore a rich repository where will be stored the generic requirements modeled for different domains
- Enrich the empirical study is the empirical study lead as a series of controlled experiments allow validating the ability of the suggested approach to solve the existing problems. Nevertheless, we believe that further experiments should be conducted to confirm the validation of the

various research hypotheses formulated through this study. We propose, to this end, to extend these experiences to other samples of subjects and/or other sectors (retail, manufacturing...). This would allow us to deduce improvements

- In automate the approach, maps are one of central concepts of the suggested approach. A Maps modeling software could be very useful to support users to model and validate different Maps from the organization requirements

On the other hand, the large number of manipulated data (for big structures) makes very difficult the manual application of the matching process. So, the automation of the approach seems essential for using it in large projects. A software environment would be a logical continuation of this study.

REFERENCES

- Addo-Tenkorang, R. and P. Helo, 2011. Enterprise Resource Planning (ERP): A review literature report. Proceedings of the World Congress on Engineering and Computer Science, Volume 2, October 19-21, 2011, San Francisco, USA.
- Alencar, F., J. Castro, L. Cysneiros and J. Mylopoulos, 2000. From early requirements modeled by the i* technique to later requirements modeled in precise UML. Proceedings of 3rd Workshop on Requirements Engineering, July 13-14, 2000, Rio de Janeiro, Brasil, pp: 92-109.
- Aloini, D., R. Dulmin and V. Mininno, 2007. Risk management in ERP project introduction: Review of the literature. Inform. Manage., 44: 547-567.
- Aumueller, D., H.H. Do, S. Massmann and E. Rahm, 2005. Schema and ontology matching with COMA⁺⁺. Proceedings of the ACM SIGMOD International Conference on Management of Data, June 14-16, 2005, Baltimore, Maryland, USA., pp: 906-908.
- Baxter, D., J. Gao, K. Case, J. Harding, B. Young, S. Cochrane and S. Dani, 2008. A framework to integrate design knowledge reuse and requirements management in engineering design. Rob. Comput. Integr. Manuf., 24: 585-593.
- Bellahsene, Z., A. Bonifati and E. Rahm, 2011. Schema Matching and Mapping. Springer, New York, ISBN-13: 9783642165184.
- Berenbach, B., D.J. Paulish, J. Kazmeier and A. Rudorfer, 2009. Software and Systems Requirements Engineering in Practice. McGraw Hill, New York, USA., ISBN-13: 9780071605472, Pages: 356.
- Bernstein, P.A., J. Madhavan and E. Rahm, 2011. Generic schema matching, ten years later. Proc. VLDB Endowment, 4: 695-701.
- Bouzid, M.R., 2012. Reuse in the elicitation of customization requirements. Proceedings of the 14th International Conference on Enterprise Information Systems, Volume 2, June 28-July 1, 2012, Wroclaw, Poland, pp: 187-191.
- Bouzid, M.R. and N. Kraiem, 2012. An empirical evaluation of a reuse based approach for ERP customization. Int. J. Comput. Sci., 9: 17-28.
- Brehm, L., A. Heinzl and L.M. Markus, 2001. Tailoring ERP systems: A spectrum of choices and their implications. Proceedings of the 34th Annual Hawaii International Conference on System Sciences, January 3-6, 2001, Maui, Hawaii, USA.
- Bush, D. and A. Finkelstein, 2002. Requirements elicitation for complex safety related systems. Proceedings of the London Communications Symposium, September 15-18, 2002, London, UK.

- Chen, F., H. Zhou, H. Yang, M. Ward and W.C.C. Chu, 2011. Requirements recovery by matching domain ontology and program ontology. Proceedings of the IEEE 35th Annual Computer Software and Applications Conference, July 18-22, 2011, Munich, pp: 602-607.
- Chernak, Y., 2012. Requirements reuse: The state of the practice. Proceedings of the IEEE International Conference on Software Science, Technology and Engineering, June 12-13, 2012, Herzlia, pp: 46-53.
- Daneva, M., 2000. Practical reuse measurement in ERP requirements engineering. Proceedings of the 12th International Conference, Center for the Advancement of Informal Science Education, June 5-9, 2000, Stockholm, Sweden, pp: 309-324.
- Daneva, M., 2001. Evaluating the value-added benefits of using requirements reuse metrics in ERP projects. Proceedings of the Symposium on Software Reusability: Putting Software Reuse in Context, May 18-20, 2001, Toronto, Ontario, Canada, pp: 155-163.
- Davenport, T.H., 1998. Putting the enterprise into the enterprise system. *Harvard Bus. Rev.*, 76: 121-131.
- Davenport, T.H., 2000. *Mission Critical: Realizing the Promise of Enterprise Systems*. Harvard Business School Press, Boston, MA., USA., ISBN-13: 9780875849065, Pages 335.
- Ernst, N., A. Borgida, I.J. Jureta and J. Mylopoulos, 2014. An Overview of Requirements Evolution. In: *Evolving Software Systems*, Mens, T., A. Serebrenik and A. Cleve (Eds.). Springer, New York, ISBN-13: 9783642453984, pp: 3-32.
- Esteves, J. and J. Pastor, 2003. Strategic and tactical critical success factors behavior along the ERP implementation phases. Proceedings of the 10th European Conference on Information Technology Evaluation, September 25-26, 2003, Madrid, Spain.
- Ettlie, J.E. and M. Kubarek, 2008. Design reuse in manufacturing and services. *J. Prod. Innov. Manage.*, 25: 457-472.
- Fagin, R., L.M. Haas, M.A. Hernandez, R.J. Miller, L. Popa and Y. Velegrakis, 2009. Clio: Schema Mapping Creation and Data Exchange. In: *Conceptual Modeling: Foundations and Applications*, Borgida, A.T., V.K. Chaudhri, P. Giorgini and E.S. Yu (Eds.). Springer, New York, ISBN-13: 9783642024634, pp: 198-236.
- Giunchiglia, F., M. Yatskevich and P. Shvaiko, 2007. Semantic Matching: Algorithms and Implementation. In: *Journal on Data Semantics IX*, Spaccapietra, S., P. Atzeni, F. Fages, M.S. Hacid and M. Kifer *et al.* (Eds.). Springer, New York, ISBN-13: 9783540749875, pp: 1-38.
- Gulla, J.A., 2004. Understanding requirements in enterprise systems projects. Proceedings of the 12th IEEE International Requirements Engineering Conference, September 6-11, 2004, Kyoto, Japan, pp: 176-185.
- Hull, E., K. Jackson and J. Dick, 2011. A Generic Process for Requirements Engineering. In: *Requirements Engineering*, Hull, E., K. Jackson and J. Dick (Eds.). Springer, New York, ISBN-13: 9781849964043, pp: 25-45.
- Hwang, Y. and D. Grant, 2011. Understanding the influence of integration on ERP performance. *Inform. Technol. Manage.*, 12: 229-240.
- Kensche, D., C. Quix, X. Li, Y. Li and M. Jarke, 2009. Generic schema mappings for composition and query answering. *Data Knowledge Eng.*, 68: 599-621.
- Klein, M., 2001. Combining and relating ontologies: An analysis of problems and solutions. Proceedings of the IJCAI-01 Workshop on Ontologies and Information Sharing, August 4-5, 2001, International Joint Conference on Artificial Intelligence, Seattle, USA., pp: 53-62.

- Kliem, R.L., 2000. Risk management for business process reengineering projects. *Inform. Syst. Manage.*, 17: 66-68.
- Kolovos, D.S., D. Di Ruscio, A. Pierantonio and R.F. Paige, 2009. Different models for model matching: An analysis of approaches to support model differencing. *Proceedings of the ICSE Workshop on Comparison and Versioning of Software Models*, May 17, 2009, Vancouver, BC., pp: 1-6.
- Laudon, K.C. and J.P. Laudon, 2001. [The Management Information Systems: Organization and Strategic Networks]. Village Mondial, France, ISBN: 2842111133, Pages: 900.
- Liu, Y. and J. Wu, 2008. Integration of OA and ERP based on service-oriented architectures. *J. Comput. Applic.*, 3: 816-818.
- Malhotra, R. and C. Temponi, 2010. Critical decisions for ERP integration: Small business issues. *Int. J. Inform. Manage.*, 30: 28-37.
- Martin, E.W., C.V. Brown, J.A. Hoffer, W.C. Perkins and D.W. DeHayes, 1999. *Managing Information Technology: What Managers Need to Know*. 3rd Edn., Prentice Hall, Englewood Cliffs, NJ., USA., ISBN-13: 9780138609252, Pages: 716.
- Mussbacher, G. and J. Kienzle, 2013. A vision for generic concern-oriented requirements reuse@21. *Proceedings of the 21st IEEE International Requirements Engineering Conference*, July 15-19, 2013, Rio de Janeiro, pp: 238-249.
- Nurcan, S., A. Etien, R. Kaabi, I. Zoukar and C. Rolland, 2005. A strategy driven business process modelling approach. *Bus. Process Manage. J.*, 11: 628-649.
- Olson, D.L. and D. Delen, 2008. *Advanced Data Mining Techniques*. 1st Edn., Springer, New York, ISBN-13: 9783540769163, Pages: 180.
- Orawski, R., C. Hein, D. Polenov, M. Holle, S. Schenkl, M. Mortl and U. Lindemann, 2013. Reuse of requirements: An approach with a generic requirements pool. *Proceedings of the 19th International Conference on Engineering Design*, August 19-22, 2013, Seoul, Korea.
- Pohl, K., 2010. *Requirements Engineering: Fundamentals, Principles and Techniques*. Springer, New York, USA., ISBN-13: 9783642125775, Pages: 814.
- Rolland, C. and N. Prakash, 2000. From conceptual modelling to requirements engineering. *Ann. Software Eng.*, 10: 151-176.
- Romeyer, C., 2003. [Analysis of barriers to the implementation of a hospital information system mapping activities: Lessons for the SI and traceability?]. *Proceedings of the 8th Association Information and Management Conference*, May 21-23, 2003, Grenoble, France, pp: 1-13.
- Rowe, F., 1999. [Consistency, informational integration and change: Outline of a research program based on the integrated management software]. *Inform. Syst. Manage.*, 4: 3-20.
- Schmidt, R., K. Lyytinen, M. Keil and P. Cule, 2001. Identifying software project risks: An international Delphi study. *J. Manage. Inform. Syst.*, 17: 5-36.
- Schneider, J.P., J. Champeau, D. Kerjean, O.K. Zein, Y. Auffret and L. Dufrechou, 2011. Domain specific modelling applied to smart sensors. *Proceedings of the IEEE OCEANS Conference*, June 6-9, 2011, Santander, Spain, pp: 1-6.
- Scott, J.E. and L. Vessey, 2002. Implementing enterprise resource planning systems: The role of learning from failure. *Inform. Syst. Front.*, 2: 213-232.
- Shvaiko, P. and J. Euzenat, 2005. A Survey of Schema-Based Matching Approaches. In: *Journal on Data Semantics IV*, Spaccapietra, S. (Ed.). Springer, New York, USA., ISBN-13: 9783540314479, pp: 146-171.

- Shvaiko, P. and J. Euzenat, 2013. Ontology matching: State of the art and future challenges. *IEEE Trans. Knowl. Data Eng.*, 25: 158-176.
- Van Lamsweerde, A., 2001. Goal-oriented requirements engineering: A guided tour. *Proceedings of the 5th IEEE International Symposium on Requirements Engineering*, August 27-31, 2001, Toronto, Ont., Canada, pp: 249-262.
- Van Lamsweerde, A., 2009. *Requirements Engineering: From System Goals to UML Models to Software Specifications*. John Wiley and Sons, New York, USA., ISBN-13: 9780470012703, Pages: 682.
- Wang, G., R. Valerdi and J. Fortune, 2010. Reuse in systems engineering. *IEEE Syst. J.*, 4: 376-384.
- Xu, L. and D.W. Embley, 2003. Using schema mapping to facilitate data integration. <http://dagwood.cs.byu.edu/deg/papers/integration.ERO3.pdf>
- Zoukar, I., C. Salinesi and C. Rolland, 2004. [Evolution of an information system by ERP implementing an ERP]. *Proceedings of the 22nd Congress INFORSID*, May 25-28, 2004, Biarritz, pp: 1-17.