An Empirical Evaluation of a Reuse Based Approach for ERP Customization

Mohamed Ramzi Bouzid[1], Naoufel Kraiem[2]

**[1] Centre de Recherche en Informatique, University of Paris 1,
Paris, 75013 90 rue Tolbiac, France**

**[2] RIADI, ENSI,
Campus Universitaire Manouba, 2010, Tunisie**

Abstract

One concern in ERP customization is the matching of ERP features with the organizations' requirements. An approach to achieve this more efficiently than with the traditional similarity measures, consists in reusing generic domain knowledge. However, except for a case study, this claim has never been substantiated before. This paper reports the first of a series of controlled experiments aimed at evaluating the efficiency of reusing generic domain knowledge in the matching activity.

The experiment showed that starting at the generic domain level is always more efficient than starting directly at the specific level, and this whether the domain knowledge is provided or constructed on the fly. Two questions remain however: (a) which is the most efficient formalism to specify generic domain knowledge? (b) Is the lesson learned valid for any domain, or is it only specific to some domains such as charity associations considered in this experiment?

Keywords: *ERP, information systems, requirements engineering, reuse.*

1. Introduction

As emphasized by various researches, one of the major requirements of an ERP implementation project is to deal with the issue of the risk of the matching between ERP functionalities and the enterprise initial needs. Most studies and researches [1] [2] [3] [4] on the matching issue are mainly interested in developing metrics to match user needs with features offered by a system in order to meet the conceptual discordance that exists between these two worlds (business and system). In addition to this major problem, we may encounter other problems, which make the similarities analysis task even more complex. Among these problems, we can cite:

- Experience and knowledge are lost.
- No support for domain analysis.
- Minor exploitation of domain-level models.
- Difficulty of matching between ERP features and the enterprise requirements by starting work from zero.

Another difficulty is that most of these approaches consider that each ERP implementation project is totally independent of the other [5] [6] [7] [8] [9] [10]. All these problems would significantly increase the risk of failure of the matching project. Moreover for the small and medium-sized companies, there is a real problem due to the fact that the ERP integration is totally based on a solution guided approach in particular for companies that cannot develop and express their requirements freely without being influenced by the solution imposed by the ERP editor.

Among the listed problems, we're interested, in this paper, on the domain reuse because very few ERP implementation methods are able to propose a concrete model of knowledge reuse at the requirements level [11] [12].

We therefore propose two main research questions:

– How reusable knowledge can contribute to achieve matching between enterprise requirements and ERP functionalities?
– What impact could have the domain knowledge reuse on the construction of enterprise specific models?

Unfortunately, there is very little empirical researches assessing the relevance and the impact of a reuse based approach on the success of a matching project and more especially in ERP implementation projects. When it is considered, they often neglect the empirical validation of the approach.

Through this article, we present a matching approach based on the domain reuse and an empirical study to evaluate the impact of this approach on the matching efficiency. This empirical study, presented as a sequence of tasks to be performed by different groups using different tools, aims to measure the influence of the generic knowledge reuse and to validate the various proposed hypotheses.

IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 3, No 1, May 2012
ISSN (Online): 1694-0814
www.IJCSI.org

The rest of the paper is organized as follow: the second section presents the methodological process followed to develop the suggested approach. Section 3 describes the experimental protocol. The analysis and evaluation of the obtained results are detailed in Section 4. The related works and the conclusion are discussed respectively in Section 5 and 6.

2. Overview of the approach

2.1 The current approach

The matching process is applied to the models derived from the Enterprise and the ERP functionalities. For that, the techniques of similarities analysis [10] [13] [14] [15] are used between these two models. Depending on the various detected similarities, a common model is built. This model represents the enterprise requirements that can be satisfied by the ERP.

Therefore, it can be seen that similarities analysis is a very important concept in this approach because it introduces a higher degree of automatization in the calculation of similarities between two different sets of models by metrics. These similarities are used to select the common parts of the two sets of models and to provide the necessary support for the model of matching in order to build the To-Be models.

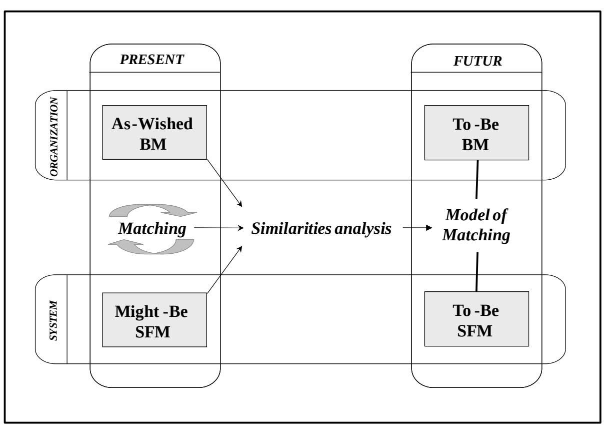

Fig. 1 The current approach

As shown, four models families have to be managed. The organization requirements are expressed in the As-Wished BM (Business Model). The Might-Be SFM (System Functionality Model) reflects the functional capability of the product. The To-Be BM and its counterpart, the To-Be SFM result from a matching process between the organizational requirements and the ERP features [10].

2.2 The approach to be evaluated

To meet the various limitations previously mentioned, we propose an approach (Fig. 2) that is based on reuse of the generic requirements of the Domain. This approach aims to improve and facilitate the matching process to be achieved between organization requirements and ERP features.

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

- Domain level: related to the common business requirements of organizations in the same activity domain.
- Enterprise level: allowing the extension of these generic business requirements to the organization specific needs.
- Solution level: representing the solution functionalities.

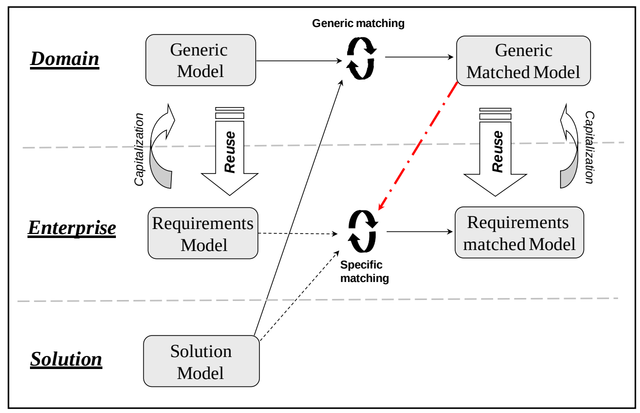

Fig. 2 The suggested approach

The suggested approach combines 2 types of matching. Firstly, the "Generic Matching" is achieved between the Generic Model and the ERP Solution Model. Secondly, the "Specific Matching" -that reuses the Domain knowledge- is done to elaborate the target Model (Requirements Matched Model).

This framework [16] is established in order to take into account the knowledge reuse aspect. In this respect, the suggested approach would initially consist in the matching of the Domain requirements (Generic Model) and the functionalities offered by the ERP (Solution Model). On this Domain level, the process is to analyse the similarities between generic requirements and ERP. A number of matching information are collected [16] and form a reusable knowledge for the analysis in the Specific level.

Once this step is accomplished, the challenge is to elaborate the Requirement Model RM (representing the own needs of a well determined enterprise). Then, the next step aims to put in correspondence these specific requirements with the ERP by taking into account the knowledge acquired in the generic matching.

The first step to build the Requirements Model RM from the Generic Model GM is to identify the differences and similarities between these two models. Based on these differences and similarities, the Requirements Model RM can be built by derivation.
Then, based on the reuse of the Generic Matching (done in the domain level), this RM will be matched with the Solution Model (ERP functionalities) to obtain the Requirements Matched Model RMM.

This specific matching is implemented according to the following stages:

- For sections of the Requirements Model (RM) that have been built directly from the Generic Model, the corresponding sections in the predefined Generic Matched Model are directly included into the Requirements Matched Model (RMM).

- For the RM sections that have slight differences (such terminologies discordance) with the Generic Model sections, the corresponding sections in the Generic Matched Model are considered with reflecting the necessary modifications (based on the changes that have been made to build these specific RM sections from the Generic Model GM).Then, these sections are included in the RMM.

- For RM sections with weak or non-existent similarities with the GM sections:
  - If this section has been created using an approach guided by the solution (ERP), so there is no need to make a matching and this section will be therefore included, as it is, in the RMM by parameterization of the corresponding ERP functionality.
  - Otherwise, a matching (similar to the matching in the Generic level) is done between this specific need and sections of Solution Model (ERP) that would allow meeting this need. This matched section is then included in the RMM. An attempt is made to enrich the Generic Model and Generic Matched Model by these added sections respectively in the RM and the RMM.

As stated above, the matching process at the specific level, unlike at the generic level, is based on the analysis of the variations between "Domain Requirements" (Generic Model) and "Specific Requirements" (Requirements Model). It should be recalled, in this regard, that few differences exist between these two levels (Domain and Specific levels) that contribute to facilitate the analysis and make it much easier.
More details for the description and the application of the suggested approach are available in [16].

2.3 The MAP formalism

To apply this Framework, it is necessary to use some sort of formalism that is able to represent the three different levels of the framework. The formalism has to:

- Be understandable by everybody (experts of business and ERP)
- Allow refinements (not permitted by I*) [31]
- Emphasize variability (not taken into account by KAOS [32]
- Remove irrelevant details (by abstraction).

For all the above mentioned reasons, we have chosen to use the MAP formalism in order to provide a unified view on the enterprise requirements as well as on the ERP functionalities. Furthermore, MAP has already shown its effectiveness in several domains [17], [18] and was very useful in the modelling requirements for different kinds of projects such as IS initial development, IS evolution [14], IS/BP alignment [20], and co-evolution [21], ERP integration [22].

Map [19] 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 and using one or several components of the organization's Information System. Strategies define approaches, means or manners to achieve goals. The strategy Sij characterizes the flow from the source goal Gi to the target goal Gj and the way Gj can be achieved once Gi has been achieved.

A section is the key element of a map. It is a triplet <Gi, Gj, Sij> and represents a way to achieve the target goal Gj from the source goal Gi following the strategy Sij. 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.

As an illustrative example of Map formalism, we present the Map (Fig.3) corresponding to the Generic Model of the Domain level representing charities [16].

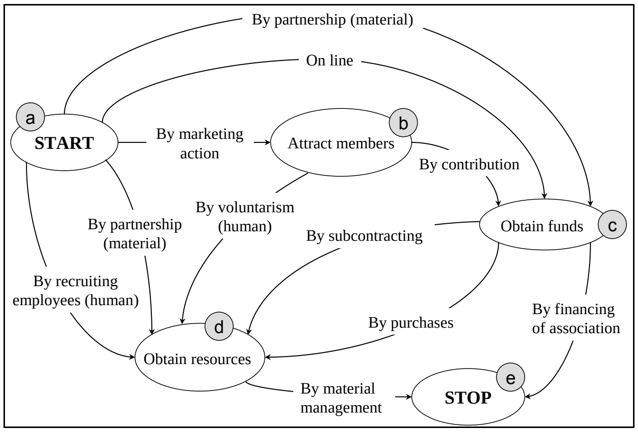

Fig. 3 Generic Model Map of Charities

3. Evaluation of the approach

In order to perform our approach, we choose to adopt an application domain that is original (for ERP), random and that doesn't require a long phase of learning. It is the domain of charities. We believe that this domain will assess the relevance and reliability of the approach for a random application domain.

As case study, we chose a charity association 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).

More details concerning the construction of the Map Models of DS specific requirements and their matching with ERP functionalities are available in [16].

3.1 Principle and purpose of the evaluation

We're currently conducting series of experiences in order to assess the impact of Domain Models in the context of the COTS engineering and more specifically its impact on the different phases of our approach of ERP integration. Our goal is to evaluate the strength of the suggested approach in terms of consistency and completeness as well as any possible improvement of this approach. The experience presented in this article consists to ask small groups of people to build different types of models used in our approach.

To ensure a great level of efficiency in conducting these experiences and to draw the appropriate conclusions, we've given different working documents to different groups (the list of provided tools will be detailed in the next paragraphs). The use of different materials, offering various forms of support to participants, helps us to assess their respective efficiency.

This experience gave us an opportunity in particular to assess: (i) the impact of the Domain Model use when building a Requirements Model (specific requirements). (ii) The impact of reuse on the matching process (between the ERP functionalities and the enterprise specific needs).

3.2 Evaluation criteria

Through the application of our approach, we expect a response to the mentioned limitations and to optimize the matching efficiency in terms of built models richness and the achievement speed.
In this context, these experiments were mainly conducted to evaluate the different hypotheses made by this research. 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 are more consistent (completeness, richness, 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...)

3.3 Subjects

In order to ensure a measurable and objective analysis of our approach according to different criterion (e.g. completeness, consistency, richness, satisfaction of needs...), we worked with 24 people who were divided into 4 groups. These people had an experience of at least 3 years in enterprises (part time). These people were not particularly expert in the charity associations' domain, so they had to learn this domain. However, these people were experts in conceptual modelling (I*, KAOS, MAP) and ERP integration domains (training on the different methodologies used in an ERP integration project as ASAP Rapid ER ...).

3.4 Experimental protocol

The activities that were assigned to these people were:

- A1 : To build the Generic Model (GM) (groups 1,2)
- A2 : To build the RM of Dr Souris Association. This activity was based on a textual description of the association needs (groups 1,2,3 and 4) and:
  - on the provided reference GM (group 3)
  - on the constructed GM (group 2)
- A3 : To build the Generic Matched Model GMM (groups 1, 2 and 3)
- A4 : To build Requirements Matched Model RMM:
  - Based on Solution Model SM (ERP functionalities), the RM and the constructed GMM (groups 1, 2 and 3)
  - Based only on the SM and the constructed RM (group 4).

Table 1: Experimental protocol

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

To better analyze the obtained results, we compared goals and sections of proposed models respectively to goals and sections of Reference models (The RM and RMM that we previously elaborated).

- If the proposed element in the constructed models (by different groups) corresponds to the reference model element (even with slight difference name but that leaves unchanged the meaning of the goal or of the section), we consider this element as “True”.
- Otherwise, if the element is absent in the reference model or is too general or is part of the reference element then we consider this element as “False”.

For each of these exercises, a graph, representing the percentage of the final model sections (already discovered) relative to time, was drawn. The time graphs of each group will be interpreted in the following section.
During this process, we’ve observed the participants activities and they were interviewed about their reasoning to build different models and about any faced problems.

4. Evaluation results and analysis

4.1 General observation

During experience, we noted that the main difficulties for participants were related to:

- Identification of requirements relating to the charity domain. The lack of knowledge of the charity domain required a major effort of research and analysis which interfered with the effort required for the modelling activity itself.
- Identification of different intentions and strategies that should exist in a Map.
- The uncertainty regarding the consistency of the final Map: the Map completeness and the verification of the abstraction level.
- The matching between different Maps: What are the intentions and strategies to be incorporated into the Matched Maps? What formulation to use?
- The appropriate use of the provided tools: doubt regarding the usefulness of the provided documents.
- Some blocking time, at the beginning of each experience related to learning time, that covers approximately 30% of the global time of activities realisation.

4.2 Detailed observations by activity

- Domain analysis: For different groups, 50% of the total time of the GM realization was mainly dedicated to the learning phase (data collection, identification of the associations’ requirements), while the real construction of the GM covered only 50% of time. In addition, the quality of the constructed models was variable (i.e. lack of consistency in terms of abstraction level between Map sections, insertion of strategy/intention with no added value for the Final Model, absence of an essential section required for the Domain).
  The experience shown that it’s difficult to control both expertise forms to ensure consistent GM (expertise in the well defined domain and in Model construction).
  We noted the repetition of few Generic goals in much models, specially: Collect donations, Satisfy beneficiaries and Communicate to attract members. This shows a certain form of determinism in the activity of intentional modelling. Moreover, this enabled us to confirm the relevance of the GM in terms of time saving in the identification of the

Generic requirements and in terms of consistency and completeness of the constructed model.

- Construction of the Requirements Model of Doctor Souris: the 3rd group was mainly based on the provided GM for determining the sections as well as for verifying the abstraction level of the different sections. The other groups were however rarely based on their own GM (1 and 2) for elaborating their RM (10 to 15% of the Model construction), some even worked entirely from scratch.
  Thus, we can note that, for group 3, the learning phase was significantly shorter. Overall, the time of Model achievement was also reduced. In addition, the constructed Models were much more consistent (in term of abstraction level) and richer (in terms of true intentions and strategies) than the maps of other groups (1, 2 and especially 4 who don't use a Generic Model).
  Furthermore, we noted that, in most cases, there are minimal differences between GM and RM. This allows us to conclude that derive a complete and coherent RM from the GM is much easier than to build it from scratch.

- Construction of the Requirement Matched Model: the first observation we have made was related to the importance of the achievement time respectively for groups 4, 1 and 2 which resulted to blocking time for the requested models. In fact, these groups didn't know how to perform the matching between the Solution Model SM and the Requirements Model, i.e. what were the sections (from these two maps) that had to be integrated into the final RMM; and how the matching should be done between these two models. This task was much easier for the group 3 that significantly reduce the achievement time of the Models.

4.3 Results and quantitative analysis

4.3.1 Definition of measurement units

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 [23].

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.

$$Precison = \frac{\left|\{relevant\ elements\}\cap\{retrieved\ elements\}\right|}{\left|\{retrieved\ elements\}\right|} \quad (1)$$

$$Recall = \frac{\left|\{relevant\ elements\}\cap\{retrieved\ elements\}\right|}{\left|\{relevant\ elements\}\right|} \quad (2)$$

A measure that combines Precision and Recall is the Effectiveness F_β (where β = Importance Recall / Importance Precision)

$$F_\beta = (1+\beta^2) \times \frac{Precision \times Recall}{\beta^2 \times Precision + Recall} \quad (3)$$

Three commonly used F measures are the F_1 measure (recall and precision are evenly weighted), the F_2 measure (which weights recall twice as much as precision), and the $F_{0.5}$ measure (which weights precision twice as much as recall).

4.3.2 Results compared to reference Requirements Model RM

The following figures show, for each of the 4 groups, Precision and Recall and 3 values of effectiveness F according to different values of Beta (β = 0.5, 1 and 2).

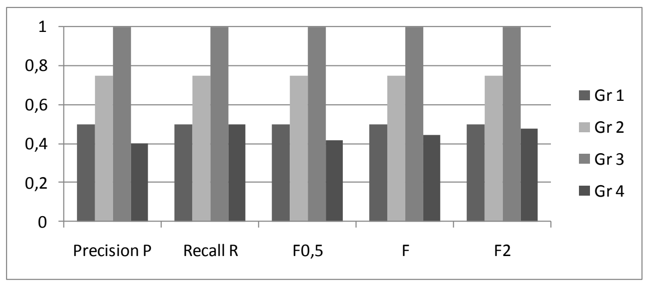

Fig. 4 Goals of Proposed RM/ Goals of Reference RM

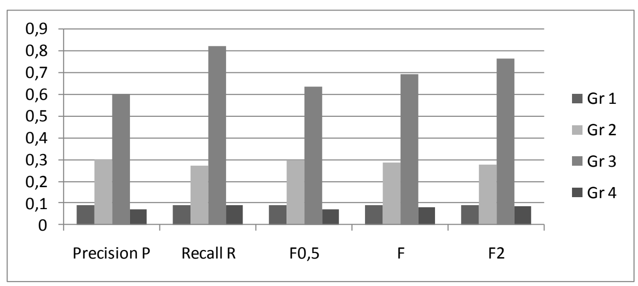

Fig. 5 Sections of Proposed RM/ Sections of Reference RM

Analysis. We note that it's the 3rd group -who is based on the provided Domain Model- had the best results in terms of Recall, Precision and therefore effectiveness. Indeed, this group has identified all the true goals without adding false one. Followed by group 2 who was based on his own built Domain Model (group 2 also followed an approach driven from generic to specific level).

The last 2 groups, who have directly begun to develop their specific model just from textual description of the association needs, had the worst results in terms of effectiveness for the construction of Docteur Souris RM.

The 3rd group was also the fastest to have constructed its model followed respectively by groups 2, 1 and 4.
We can therefore see that: to follow domain knowledge based approach helps to build faster a more accurate, complete and consistent RM regardless the β value.

4.3.3 Results compared to Reference Requirements Matched Model RM

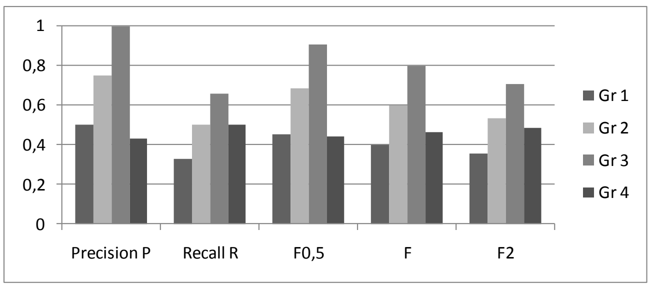

Fig. 6 Goals of Proposed RMM/ Goals of Reference RMM

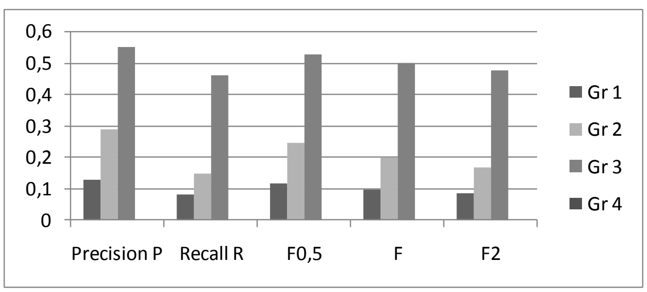

Fig. 7 Sections of Proposed RMM/ Sections of Reference RMM

Analysis. 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).
- Followed by group 2 (that used its own Generic Model).
- Group 4 had, for sections table, smaller values of P, R and F (Group 4 is the only group that has not any reasoning at the generic level to build his RMM): lack of strategies and intentions are inconsistent (not at the same abstraction level).

From these results, we conclude 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).

4.4 Self critics and threats to validity

As we have noted, our experience has shown its relevance in the evaluation of our approach, however, some limitations can be identified:

- Since it remains an approach based on modelling, several solutions are possible. Thus, there’s a need for a specific analysis for each solution.
- The time measurement isn’t very precise since it’s based on approximate measures provided by different users.
- The necessity to extend this experience to other domains (industrial, services...) and to other ERP in order to draw final conclusions for this evaluation.
- Size of the people sample could be larger.
- A limited time work.
- Choice of MAP formalism.

Among these limits to the validation, we chose to answer to the one that we considered the most important namely the comparison of results against a predefined reference model since these primary results could be discussed because of possible bias introduced by the incompleteness of the reference model.
To best meet this threat, we have re-examined the results obtained by different groups and we have recalculated the values of Precision, Recall and F-measure for all groups.

These calculations will be done based on an improved RMM (that will incorporate the intentions and the sections that can be logically added to the original RMM from the suggested RMM by different groups) in order to draw final conclusions.

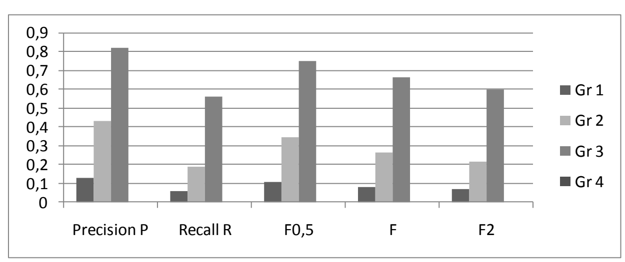

Fig. 8 Sections of Proposed RMM/ Sections of updated RMM

Analysis: The obtained results in this 2nd evaluation strengthen the first one. Indeed, we can still see that the 3rd group, in addition to have the best results in terms of Precision, Recall and effectiveness, it is the group that successfully identifies the most consistent new sections that could be included later in the updated RMM (Knowledge capitalization process).

However, the 4th group -who didn't at all based on reuse in its approach- has always the last place in terms of effectiveness of the final RMM, and none of its sections have been integrated into the updated RMM due to lack of coherent strategies in this model (This explains the zero values of this group in the graph above). We also note that whatever the value of β (if we increase the importance of completeness relative to the relevance or conversely increasing the importance of the relevance relative to completeness) it is the 3rd group who always keeps the best values of effectiveness. For RMM sections, the 4th group still has the lowest effectiveness values whatever the value of β.

This confirms the importance of an approach based on a domain analysis for the rapid construction (no blocking time relative to domain needs learning) of a more consist (more complete and rich) model that can enrich thereafter the updated model.

From this experience, we can point out the similarities between the GMM and RMM. Indeed, to obtain the RMM, we had just to impact into GMM, the modifications incorporated to the RM as compared to GM. Therefore, we can save considerable time in the construction of more consistent Models that better satisfy the needs and requirements of different associations.

4.5 Time measurements

Alongside each of these tasks, we asked each different group to draw a graph representing the evolution of his model (in terms of sections number) versus time. To better visualize and analyze time measurements results, we represent them in Fig. 9 and Fig. 10.

A quick look on these first results allows us to make these initial observations:

- The only curve that emerges clearly from the other is that of group 3 (other groups have more or less the same curves aspect).
- Group 3 was the first to complete the construction of its models (RM and RMM models).

Therefore, we can see that the 3rd group (the one that was based on reference models provided) is the fastest to achieve its own models.

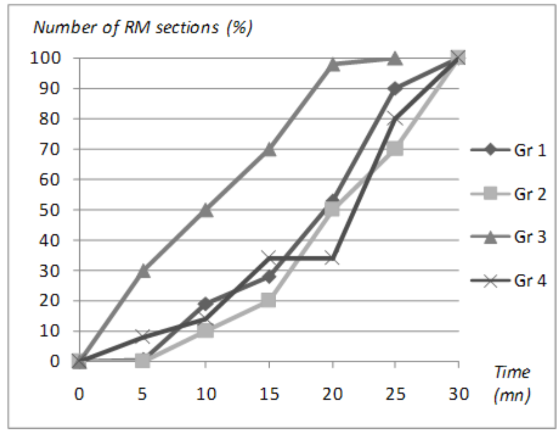

Fig. 9: Time measurements of RM sections build

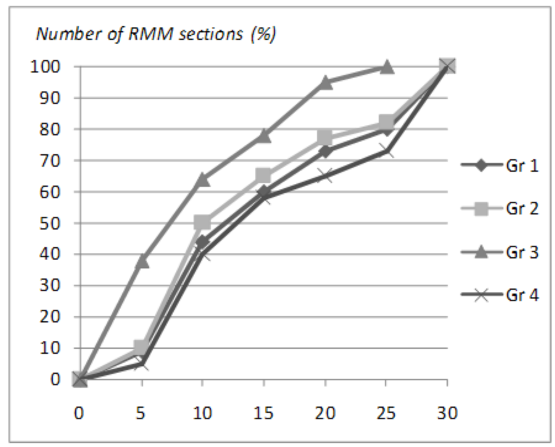

Fig. 10: Time measurements of RMM sections build

4.6 Discussion

As can be seen, the suggested approach can be used to satisfy a number of problems that have been exposed through this article, mainly:

- The construction of the Requirements Models and the Requirements Matched Model is more coherent, rich and so much easier.
- Considerable time-saving: the time spent to build the GM is totally compensated by the rapidity of matching between the specific needs and ERP functionalities.
- Reduction of the inconsistencies in RM and RMM.
- No loss of the acquired knowledge by capitalization.

These various observations reflect the pertinence of the above presented approach.

By putting the accent on this environment, original and random, the suggested approach will allow, by starting reasoning from the Domain level, modelling any association in a better, faster and easier way. This aspect of

Top-Down reasoning (from generic to specific), added to the aspect of knowledge reuse, could also be seen as a helpful tool to accelerate and facilitate the ERP implementation.

5. Related works

Software reuse is the process of reusing existing software artefacts in building a new system rather than starting the process totally from scratch. It should be recalled that the development of software systems based on reusable components has been present from the first stage of the computing. However, what has indeed changed over time is the size and complexity of the artefacts being reused. The arguments for reuse remain the same: we can do more with higher quality, for lower cost. Thus, this process would ensure a better response to the actual challenge facing this model in particular with products that are getting more complex with shorter times to market; there is definitely a pressure to reuse larger artefacts.

Most of the research in the reuse area neglect requirements engineering. Although it is argued [24] that requirements reuse can introduce more reusability at later stages in the product life cycle, it was not until recently that requirements reuse have received a greater attention by the researchers. In this regard, most of requirements reuse is done in an informal manner by researchers who can develop faster new requirements specification taking advantage of their previous experience in this domain. Their experience is used in this respect to helps them informally reuse requirements.
In the ERP implementation projects, it has been observed that most of the reuse metrics refer to design/code level (with issues still difficult to be understandable in the requirements analysis phase [25], [26], [27] and properly quantified) and few systematic investigations exist at the requirements level[11], [28], [29] and [30].

One might argue that the interaction analysis is only necessary at the time when the reusable requirements are created. Since these reusable requirements will not change, no further interaction analysis is necessary when new system is built based on the existing reusable library, one, however, has to consider that there are parameterized requirements that need specific values, that is, they are really "new" requirements.

6. Conclusion

A lot of researches and studies that have analyzed this issue have focused on the issue of matching between the business world with the ERP world, as well as on the techniques of similarities analysis to support this matching without taking into account the reuse aspect that exists between the SME in the same activity domain.
Our approach has therefore tried to bring a response to this problem, by ensuring an important time saving to build a better quality models.
Indeed, our approach demonstrates that it's not only the fact of reusing domain knowledge but also and especially the fact of following reasoning from a generic to a specific level that improves the quality of the specific requirements models and the matched models in terms of consistency, richness and completion time.

The originality of the suggested approach resides in the fact that it is guided by the domain and not by the solution allowing it to remain independent of the ERP.
In addition, this approach facilitates the matching between the ERP functionalities and the association needs by reusing the experience and the domain knowledge to avoid resuming work each time from scratch. This approach can therefore play the role of an accelerator in the matching by providing a support for the domain analysis. Moreover, it can constitute an efficient mean of training for the new users who can take advantages of the capitalized experience sharing.

Our experience with "Docteur Souris" Association allows us to provide an answer to several problems previously faced in particular those related to the loss of acquired knowledge and ensure a considerable profit in terms of quality and times.

Finally and in order to improve and evaluate the completeness of this approach, we must answer to some challenges in particular the applicability of the lessons learned to domains other than the charity associations, and the identification of the best formalism to capitalize the generic knowledge. Tasks those are underway.

References

[1] Standish, The Standish Group. Chaos Standish Group Internal Report. www.standishgroup.com/sample_research/chaos_1994_1.php, 1995.
[2] T.H Davenport, Mission Critical: Realizing the Promise of Enterprise Systems. Harvard Business School Press, MA, 2000.
[3] J.G. Bernard, and S.Rivard, and B.Aubert, Evaluation du risque d'implantation de progiciel. CIRANO Project Reports, Centre Inter universitaire de Recherche en Analyse des Organisations. 2002.
[4] J. Esteves and J. Pastor, Strategic and Tactical Critical Success Factors Behaviour Along the ERP Implementation Phases. The European Conference on Information Technology Evaluation (ECITE), Madrid (Spain), September 2003.

[5] M.Krumbholz and N. Maiden. How Culture might impact on the Implementation of Enterprise Resource Planning Packages. Procs of CAISE'00, Springer Verlag, Ref. HCID00/04, 2000.

[6] C. Ncube and N. Maiden, COTS Software Selection: "The Need to Make Tradeoffs Between System Requirements, Architectures and COTS/Components". Procs of the COTS2000 Workshop for ACSE2000, Limerick, Ireland, refHCID00/02, 2000.

[7] A. Finkelstein and D. Bush, Requirements Elicitation for Complex Safety Related Systems, In Procs London Communication Symposium. London, UK, Sept. 2002

[8] J. Esteves and J. Pastor and J. Casanovas, Monitoring Business Process Redesign in ERP Implementation Projects. Americas Conference on Information Systems, Dallas-USA 2002.

[9] D. Robey and J.W Ross and M-C Boudreau, Learning to Implement Enterprise Systems: An Exploratory Study of the Dialectics of Change. Journal of Management Information Systems, Summer 2002.

[10] I. Zoukar and C. Salinesi, Reducing the Language Disparity Issue in ERP Projects Using Goal/Strategy Modelling. The 6th International Workshop on Agent-Oriented Information System (AOIS'2004). Riga, Latvia, June 8, 2004.

[11] M. Daneva, Practical Reuse Masurement in ERP Requirements Engineering, LNCS, Proc. of Int. Conf. On CAiSE, Stockholm, Springer Verlag, 2000.

[12] M. Daneva, Evaluating the Value-added Benefits of Using Requirements Reuse Metrics in ERP Projects, Proc. of 6th Symp. on Software Reuse, Toronto, ACM Press, 2001.

[13] C. Rolland and C. Salinesi and A. Etien, Eliciting Gaps in Requirements Change. Requirements Engineering Journal, Vol. 9, pp1-15, 2004.

[14] C. Salinesi and A. Etien and I. Zoukar, A Systematic Approach to Express IS Evolution Requirements Using Gap Modelling and Similarity Modelling Techniques, International Conference on Advanced information Systems Engineering, Springer Verlag, Latvia, 2004.

[15] I. Zoukar and C. Salinesi, Matching ERP Functionalities with the Logistic Requirements of French Railways: A Similarity Approach. The 6th International Conference on Enterprise Information Systems, Porto, Portugal, 2004.

[16] C. Salinesi and M.R Bouzid, "An Experience of Reuse Based Requirements Engineering in ERP Implementation Projects."International Enterprise Distributed Object Computing Conference (EDOC), Annapolis, Maryland, USA, October 2007.

[17] C. Rolland and N. Prakash, Bridging the Gap Between Organizational Needs and ERP Functionality. In Requirements Engineering Journal, 5(2000).

[18] C. Rolland and N. Prakash, Matching ERP System Functionality to Customer Requirements. In Procs of the 5th IEEE International Symposium on Requirements Engineering, Toronto, Canada. August 27-31, 2001.

[19] S. Nurcan and A. Etien and R. Kaabi and I. Zoukar and C. Rolland, A Strategy Driven Business Process Modelling Approach. Special issue of the Business Process Management Journal on Goal-oriented Business Process Modeling, Emerald 2005.

[20] A. Etien and C. Rolland and C. Salinesi, Measuring the Business/System Alignment, Requirements Engineering for Business Need and IT Alignment (REBNITA), Paris, 2005.

[21] A. Etien and C. Salinesi, Managing Requirements in a Co-evolution Context, Requirement Engineering (RE), IEEE Computer Society Press, Paris, France, September 2005.

[22] I. Zoukar and C. Salinesi, Matching PeopleSoft Functionalities with SNCF Logistic Requirements: A Similarity Approach. 6th International Conference on Enterprise Information Systems, Porto (Portugal), 2004.

[23] J. Makhoul and F. Kubala and R. Schwartz and R. Weischedel, Performance measures for information extraction. In: Proceedings of DARPA Broadcast News Workshop, Herndon, VA, February 1999.

[24] M. Shehata and A. Eberlein, Issues in Requirements Reuse and. Feature interaction Management, ICSSEA 2002

[25] C. McClure, Reuse Engineering: Adding Reuse to the Software Development Process", Prentice-Hall, NJ, 1997.

[26] J. Poulin, "Measuring Software Reuse: Principles, Practices, and Economic Models", Addison-Wesley, Reading, MA, 1997.

[27] W. Lim, Managing Software Reuse, A Comprehensive Guide to Strategically Reengineering the Organization for Reusable Components, Upper Saddle River, NJ, 1998.

[28] M. Daneva, Measuring Reuse of SAP Requirements: a Model-based Approach, Proceedings of the 1999 symposium on Software reusability, Los Angeles, California, USA.

[29] M. Daneva, Identifying and Quantifying Reuse in the SAP Requirements Engineering Process, Proceedings of REFSQ'99.

[30] M. Daneva, Establishing Reuse Measurement Practices in SAP Requirements Engineering, Proceedings of the 4th International Conference on Requirements Engineering (ICRE'00).

[31] F. Alencar and J. Castro and L. Cysneiros and J. Mylopoulos, From Early Requirements Modeled by the i* Technique to Later Requirements Modeled in Precise UML. In Anais do III Workshop em Engenharia de Requisitos. Rio de Janeiro, Brazil, 2000.

[32] A.V Lamsweerde, Goal-Oriented Requirements Engineering: A Guided Tour. Invited Paper for RE' 2001 - 5th IEEE International Symposium on Requirements Engineering, Toronto, pp. 249-263, August, 2001.