

REQUIREMENTS MANAGEMENT

USING POSITIONING REQUIREMENTS IN ENTERPRISE SYSTEM PROJECTS

Clotilde Rohleder, University of Paris 1 Pantheon Sorbonne, Centre de Recherche en Informatique,
clotilde.rohleder@ugs.com

ABSTRACT

Managing requirements in Enterprise System Projects should be easier using a more precise visualization tool as the current ones. Within Enterprise System projects, it is only possible for the decision makers to select the adequate requirements when they don't only consider the balance but also the importance of the implied organizational changes as well as the depth of the Enterprise System customization in a specific project situation. Our approach is based on industrial projects and suggests a tool called the Requirements Positioning. It considers the importance of the implied organizational changes and the Enterprise Systems customization and helps to visualize requirements for a specific project situation.

Keywords: Information Technology (IT), Positioning Requirements, Requirements Management

INTRODUCTION

Today, software companies have to face to the challenge of becoming and staying competitive. They have to steadily improve time-to-market, product quality, reliability and profitability. To reach this goal, they invest in new technology and attempt to improve their processes. Also, many are taking a comprehensive approach to requirements analysis that includes a determination of whether customer reorganization is appropriate. To develop recommendations, generally software companies employ experts on organizational structure in the requirements phase. Proposed structural changes could require the new enterprise system to be customized to support new organization structure. Because these customization requirements are neither trivial nor obvious, generally a tool is needed to help visualize, document and manage the changes in organizational structure and the software customization. Our proposed Requirements Positioning tool addresses this need. It considers the importance of the implied organizational changes and the necessary system customization and helps to visualize how they can be managed in a specific project situation. After we explain what “positioning

requirements” means, we will illustrate with some examples. Then we will show how it aids visualization of the project and how to analyse and interpret the results of Requirements Positioning. We conclude with a summary of this new approach and mention research opportunities

POSITIONING REQUIREMENTS

Positioning requirements is a way to achieve the balance recommended by Jon Atle Gulla [5]. Indeed, implementation of systems such as ERP or PLM requires in-depth collaboration between experts in management and software professionals. For example, implementations of enterprise systems at business giants from the automotive, defense, electronics or consumers industry has involved extensive consulting with experts from well known firms including A.T.Kearney from Electronic Data Systems, Mc. Kinsey Inc. and Andersen Consulting. The decision to implement a new enterprise system often provides an opportunity to restructure and optimize the organization. Often the Return on Investment (ROI) proposed by vendors of enterprise systems is based largely upon redesign of organization structure or workflow rather than the system itself. Failing to recognize these associated organization changes could cause problems such as user resistance [2, 8]. Reorganization could result in migration of human resources to different departments as well as creation or reduction of positions. In conjunction with implementing a new system, top management could set new goals for the company. For example they could set a goal to reduce data search time by 50% up to 5 seconds per document, or to reduce change request processing of product manufacturing by seventy percent, or to eliminate paper-based documentation, or to reduce errors to below 5%. But many of these requirements may not be satisfied by standard enterprise systems, so customization is necessary. This customization may involve revising data models, adding software functions, building interfaces to other enterprise systems, etc.

Developing complete requirements for software and for organizational restructuring is necessary but not sufficient for a successful enterprise system project.

The project must achieve an appropriate balance between structural change and system customization. The leader must be able to visualize the relative importance as well as the interplay and the interaction between these two aspects of the project. Not all departments of a company are affected by both particularities of projects implementation, where the organization needs to be completely restructured to reach its goals. This visualization must consider the various departments of the organization because each may have a different balance between structural change and system customization. The decision makers should find this information as quantifiable and qualifiable data.

We propose a visualization tool called Requirements Positioning (RP). RP provides the following:

- Aid in understanding the importance of the required organizational change as well as the importance of enterprise system customization.
- Expression of the importance of requirements either in absolute or relative values.
- Comparison of the requirements among departments.

Our Requirements Positioning model gives statement results. These statements are associated to places and marks and events to transitions.

VISUALIZING THE REQUIREMENTS POSITIONING

The Requirements Positioning tool provides a view that gives management information concerning the requirements of software customization, the organizational change, and the project situation:

- the first dimension for organisational change,
- the second dimension for the enterprise system customization

This visualization extends Jon Atle Gulla's [5] approach by handling the balance between organizational and customization. The Requirements Positioning can be applied to requirements for any enterprise system. Depending on the departments in which the company wants the system to be implemented, the positioning of requirements may differ from the organizational or the customizational point of view or both at the same time. We consider an Enterprise System project involving the Marketing and Quality Management Departments.

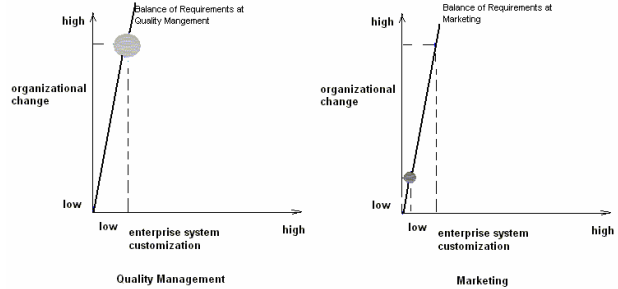


Figure 1. Requirements Positioning of Enterprise System at Marketing and Quality Management Departments

In Figure 1, notice that we could have the same balance for the system requirements at various departments but different depths of implementation. On the other hand, the representation and visualization portfolio of different enterprise systems requirements at the departments of Quality Management and Marketing could have a completely different balance. So the Requirements Positioning for the department Quality management *RP1QualityManagement* and the one for the Marketing Department *RP2Marketing* may look as follows:

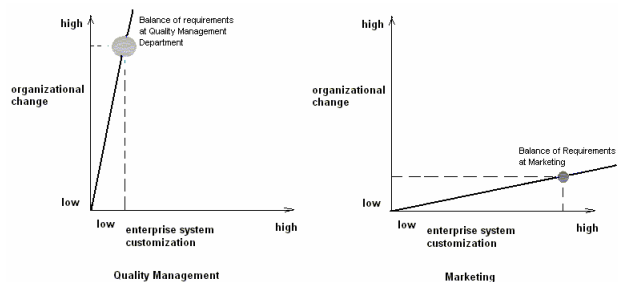


Figure 2. Requirements Positioning of Enterprise System at Marketing and Quality Management Departments Showing a Different Balance

As you can notice in the above figures, comparisons can be made very easily concerning the Requirements Positioning between all departments involved by the enterprise system project. In the Marketing Department, the requirements seem to give high priority to customization. For example, the system does not provide sufficient out-of-the-box functionalities needed by the department. On the other hand, the department's organization does not need to be restructured while implementing the software. The opposite situation of positioning requirements occurs at the Department of Quality Management as shown in the Figure 2. This lets us understand that the department of Quality Management will have another implementation plan as the customization seems to be low, whereas the

need of structural change is high in that case. The synthesized form for the Requirements Positioning of the Department of Quality Management RP1*QualityManagement* and Marketing Department RP2*Marketing* looks as follows (Figure 3).

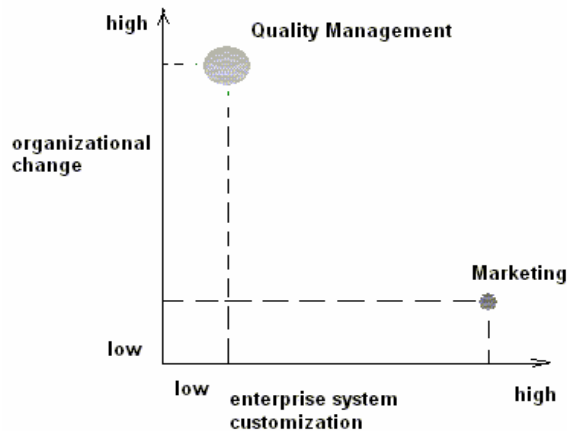


Figure3. Synthesized Form of Requirements Positioning for Departments of Marketing and Quality Management

This synthesized form can contain all Requirements Positioning of each department involved in the enterprise system project. The circle for positioning requirements at Quality Management is 4 times larger than this one of Marketing. This could mean Quality Management has 4 times more users affected by the enterprise system project.

As soon as the tool has synthesized the requirements by giving the management its visualization of Requirements Positioning, the result will be considered by the management and technical experts to design and plan the best way of implementation beginning by setting it in its globality and going forward in its precision for each department. The concept can respect the specificity of each department and can summarize all the steps by referencing to the department oriented Requirements Positioning. This valuable information concerning the requirements can be analyzed to help the decision makers. The profitability and correct run of cost aspects within the implementing organization is the first guarantee for an Enterprise System project to be successful. The Requirements Positioning allows a correct cost controlling. The correct analysis helps to precalculate the ROI per department and take the corresponding decision if the implementation concept suits the organizational and financial aspects like turnover, win and contribution plan.

REQUIREMENTS POSITIONING AND PRIORITIZATION

Our approach of positioning requirements completes Regnell's, Karlsson's and Höst's approach [9] of requirements prioritization by giving the management global requirements overview. The overview can be global or oriented to the controlling need of the Management. Unlike Regnell's, Karlsson's and Höst's approach, our tool places the requirements in the specific project situation. Moreover, requirements may collide within one enterprise system project as the priority differs from department to department. The coefficient of importance of some requirements can be visualized through the use of circles having different sizes that are easy to interpret. Using the synthesized result of Requirements Positioning, the management can view at first sight the global requirements situation and better estimate the consequences. As shown in the synthesized form of Requirements Positioning of enterprise system at the Marketing and Quality Management, it is then easier for the management to make decisions and set priorities for meeting the requirements, especially when requirements are different in the two departments. Our approach of positioning requirements requires the use of a requirements database, e.g., through the software Teamcenter Requirements®, and implements the process of estimation of cost and benefits as it is the case in Regnell's, Karlsson's and Olsson's approach [7, 11].

APPLIED EXAMPLE: TEAMCENTER REQUIREMENTS®

Our applied example of positioning requirements requires the use of the requirements database Teamcenter Requirements®, and implements the process of estimation of cost and benefits. The calculation has been preceded by a benchmarking method [6] which requires UML based benchmark-scenarios. These UML based scenarios are included in the database of Teamcenter Requirements® [1].

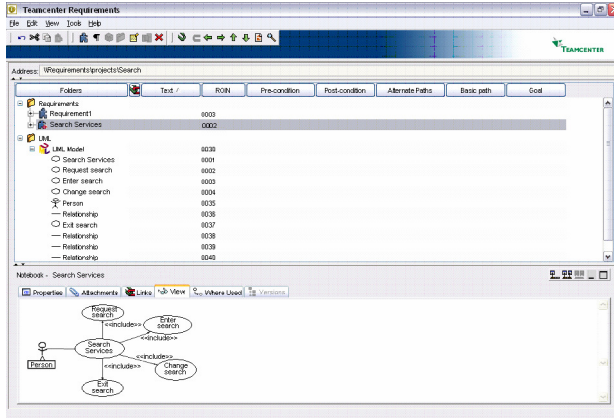


Figure 4. Integrated Requirements and UML Environment

The planning of additional features and releases has to be preceded by a useful method to capture the requirements and set the prioritization for each of them. This can be done manually [4] or with the use of requirements software like Teamcenter Requirements®. This method is based on the cost approach. The cost-value approach [7] can be used to calculate the software customization and organizational change part for Requirements Positioning. Teamcenter Requirements® enables capturing, organizing and prioritizing the requirements (software customization, organizational change), while allowing entitled users to easily access these requirements.

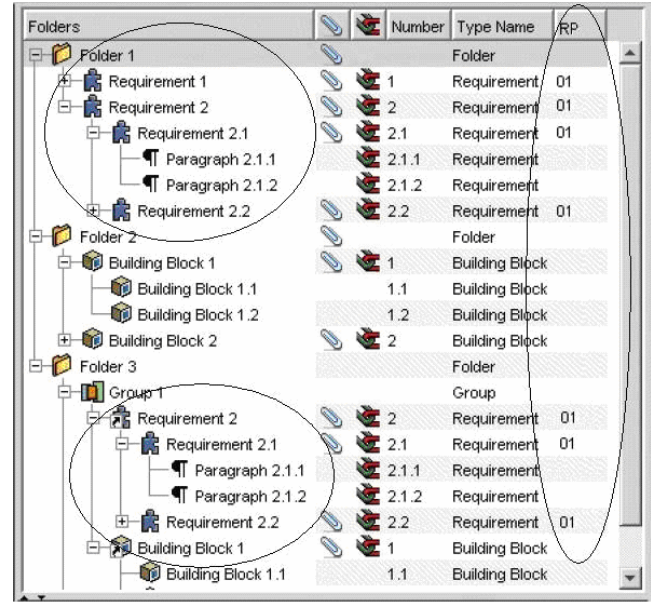


Figure 6. Example of Organizational Change part of Requirements Positioning within Teamcenter Requirements

The Requirements Positioning in this example shows a low value for structural change but a high value for the software customization. The Requirements Positioning can be visualized as follows. How to interpret and analyze the RP result is explained in the next section.

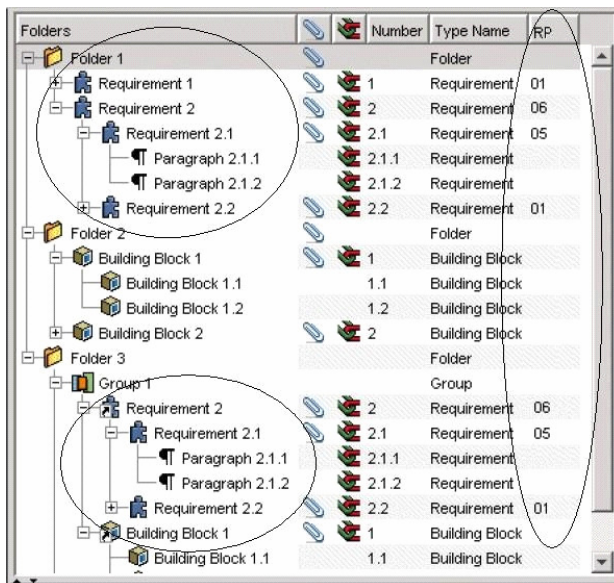


Figure 5. Example of Customization part of Requirements Positioning within Teamcenter Requirements

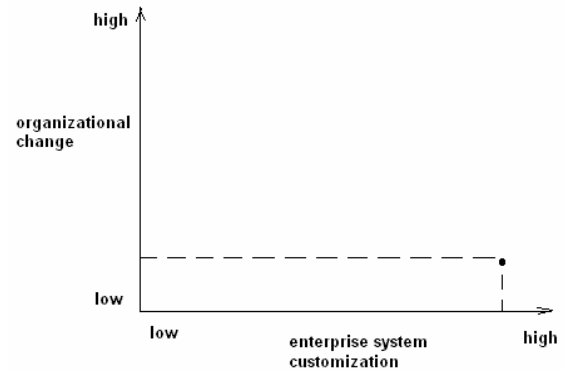


Figure 7. Visualizing Requirements Positioning RP

INTERPRETING THE RESULTS OF EXPERIENCING THE REQUIREMENTS POSITIONING

We can analyze and interpret the Requirements Positioning results through the related product lifecycle management projects of UGS, formerly EDS PLM Solutions.

Four Main Groups of Requirements Positioning

Requirements Positioning gives a company or department specific, detailed results. Only by simplifying and grouping the results can we gain a high level perspective. We simplify by grouping all Requirements Positioning possibilities into four main groups. For each of these four main groups we will develop an analysis and interpretation of results.

The first and fourth positioning requirement situation visualizes the typical balance between the enterprise system customization and the structural change. The second and third cases are typical unbalanced requirements between the enterprise system customization and structural change.

First Case of Requirements Positioning

The objective description of the first positioning possibility would be to say this is the typical case of light change for all stakeholders involved in the enterprise system project. The company does not have to restructure its organization and the software vendor does not have to customize its enterprise system. Data model, functionalities, workflows are implemented “out-of-the-box”. In the first case, the analysis of Requirements Positioning RP1 in term of implementation time and costs can be summarized as low investment cost and short implementation time (Figure 8).

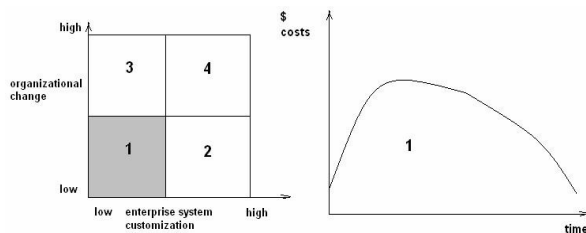


Figure 8. Analysis of the First Case of Requirements Positioning

Second Case of Requirements Positioning

The second group of Requirements Positioning is the typical case where the software has to be customized and adjusted to the company organization. This case occurs when a company does not wish to change the organizational structure, as is typical for a traditional management style or when it is not allowed for the company to change any structure. This is the case for strong documented and controlled organizations and standardized industries like defense. The only way to implement the enterprise system is to customize it if

it does not already support the organization structure. The time and cost analyses depend on each other and have a typical behavior. The software vendor has to customize a lot so the early costs are covered by the software vendor. As soon as the new functionality can be installed, tested and accepted by the customer, the invoicing can be carried out and only then the company implementing the enterprise system has to incur the costs. The analysis case for Requirements Positioning RP2 is represented in Figure 9. So the costs appear late but are high compared to the first case. Packages are customized and installed for a continuous transition.

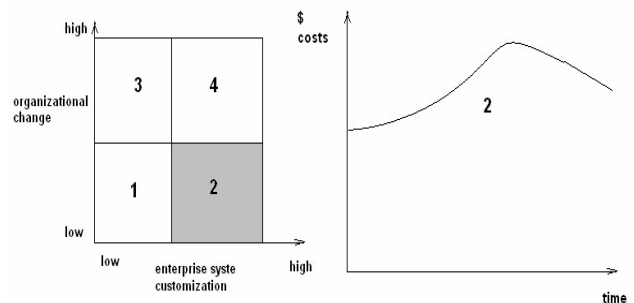


Figure 9. Analysis of the Second Case of Requirements Positioning

The Third Case of Requirements Positioning

According to the third Requirements Positioning we recognize a typical case of low customization and high structural change. This may occur in companies which use the standard enterprise system for supporting the new targeted organization structure or maybe the software as such a monopolistic economical position on the market that it acceptable to be inflexible. We speak about extensive rather than moderate process reengineering [3]. In this situation the analysis should take into account that the organization change will be costly in the early implementation phase. The adaptation of the work organization and workflow in a company to new structures takes time and requires training to acquaint people with the new corporate work philosophy. The development of costs depending on the implementation time of Requirements Positioning RP3 can be shown as illustrated in Figure 10.

Structural change in an organization costs time and money. The leaders for change have to face to several problems like a bad acceptance or resistance from the employees for the new structure. This is due to the effect of being afraid against the unknown new structure or because of the risk to loss power [2, 8].

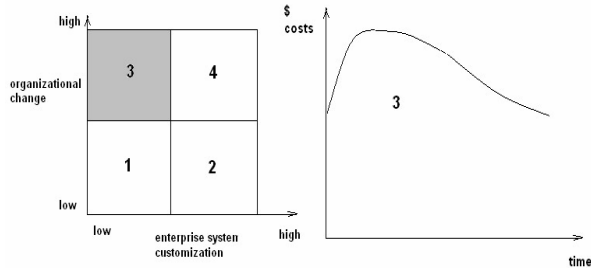


Figure 10. Analysis of the Third Case of Requirements Positioning

The Fourth Case of Requirements Positioning

The fourth case is a combination of situations two and three. By combining both figures, we get the typical analysis of the fourth Requirements Positioning. We have illustrated the case of Requirements Positioning RP4 in Figure 11.

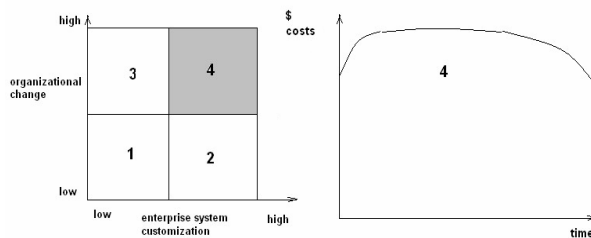


Figure 11. Analysis of the Fourth Case of Requirements Positioning

The company has to face a radical structural change in its organization and carry very high enterprise system customization costs. This means high costs occurring early. These costs are going to stay steady until the last implementation of customizations is completed.

CONCLUSIONS

In this paper, we proposed a way to visualize the Requirements Positioning. We use a graphic to represent the relationship between the requirements of customization of enterprise systems and structural change placed in a specific project situation. The purpose of the Requirements Positioning is to give management a tool to visualize a very complex requirements situation and to describe the differences in the requirements between all the departments involved. Expressing the Requirements Positioning helps to lead an enterprise system project to its success. It synthesizes and summarizes the relevant information resulting from the reports into a

clarifying form that can be analyzed and interpreted in order to carry out the best plan. Future research should give us more information about the time and change management within the Requirements Positioning. We will include decision pattern corresponding to the four typical Requirements Positioning results

REFERENCES

1. Broeze, H. (2004). The value of UML use-case-modeling in product life cycle context, *INCOSE 2004*, Toulouse.
2. DeMarco, T. & Lister, T. (1999). *Peopleware: Productive Projects and Teams*, 2nd Ed., 1999.
3. Esteves-Sousa, J. & Pastor-Collado, J. (2000). Towards the unification of critical success factors for ERP implementations. In *10th Annual Business Information Technology (BIT) 2000 Conference*, Manchester.
4. Fuchs, E., K.H., & Hauri, H. (2002). *Requirements Engineering in IT*, Bertelsmann Springer Wiesbaden, 43-49.
5. Gulla J.A.. (2004). Understanding requirements in enterprise systems projects, p178-184, *12th IEEE International Requirements Engineering Conference, 2004*.
6. Hagge, L. & Kreutzkamp, J. (2003). A benchmarking method for information systems, Deutsches Elektronen-Synchrotron, *11th IEEE International RE Conference, 2003*
7. Karlsson, J. & Ryan, K. (1997). Prioritizing software requirements in industrial setting, *ICSE*.
8. Markus, M. (1983). Power, politics, and MIS implementation. *Communications of the ACM*, 26 (6), 430-444.
9. Regnell, B, Karlsson, L, Höst, M. (2003). An analytical model for requirements selection quality evaluation in product software development, *11th IEEE International Requirements Engineering Conference, 2003*.
10. Rolland, C., Prakash, N. & Benjamin, A. (1999). A multi-model view of process modelling, *RE Journal*, 1999.
11. Karlsson, L, Regnell, B, Karlsson, J, & Olsson, S. (2003). Post-release analysis of requirements selection quality, 1-9. In Salinesi, C., Regnell, B. & Kamsties, E. *Proceedings of the 9th Intl. Workshop on Requirement Engineering Foundation of Software Quality (REFSQ '03) Essen, 2003*.