Published in: International Conference on Enterprise Information Systems (ICEIS), 2002, Spain

A FRAMEWORK TO ANALYSE MOST CRITICAL WORK PACKAGES IN ERP IMPLEMENTATION PROJECTS

José Esteves
Department of Lenguajes y Sistemas Informáticos , Universidad Politécnica de Catalunya, Jordi Girona Salgado 1-3
08034 Barcelona, Spain
Email: jesteves@lsi.upc.es

Joan A. Pastor
Universidad Internacional de Catalunya,Immaculada 22
08017 Barcelona, Spain
Email: jap@unica.edu

Keywords: ERP systems, ERP implementation, critical success factors, criticality, work packages

Abstract: In order to achieve success in a software project, it is important to define and analyze the most critical processes within the project. A common approach to define most critical processes is the Process Quality Management (PQM) method. However, the process structure of the PQM method is too simple since it only provides one level of process analysis. Real cases imply project process structures that are more complex. We have improved the PQM analysis section to provide more depth to real project structures. This study attempts to analyze this issue in a specific type of software projects: Enterprise Resource Planning (ERP) implementation projects. We present a framework to analyze most critical work packages in ERP implementation projects. We then apply the result of the analysis to SAP implementation projects. The result is a list of critical work packages in each phase of a SAP implementation project. These results show the higher importance of work packages related with organizational and project management aspects compared with the technical ones. Therefore, these results evidence the need of project managers to focus on these work packages.

1. INTRODUCTION

In order to achieve success in a software project, it is important to define and analyse the most critical processes within the project. If organizations had unlimited resources, each process within the project could have equal attention for resources and management focus. But in practice, the time and resources of project managers are always limited. Therefore, it is important to pinpoint those activities that warrant the most attention (Hardaker and Ward 1987). We may define a project as "a unique set of coordinated activities, with definite starting and finishing points, undertaken by an individual or organization to meet specific objectives within defined time, cost and performance parameters" (Association of Project Management 2000) that "it is only completed when the deliverable has been produced to the satisfaction of the customer" (Shenhar and Wideman 2000). A software project is "the set of all project functions, activities, and tasks, both technical and managerial, required to satisfy the terms and conditions of the project agreement. A software project should have specific starting and ending dates, well-defined objectives and constraints, established responsibilities, and a budget and schedule. A software project may be self-contained or may be part of a larger project. A software project may span only a portion of the software product lifecycle" (IEEE 1998). According to Savolainen (1991), a software project is successful if the implementation of the resulting information system (IS) supports effectively the business goals defined by management, and fulfils the users' requirements, if the users are satisfied with the system, if the system is implemented within the

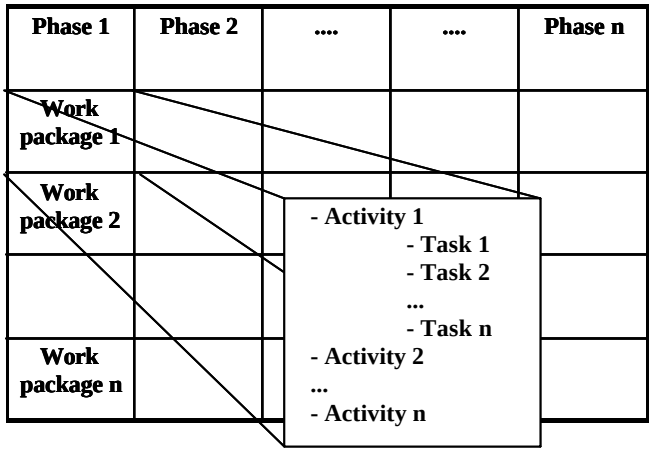

Figure 1 – A common software project structure.

given time and budget, and if the system maintenance is flexible.

Pinto and Slevin (1987) defined a model of project implementation success as $S=f(x_1, x_2,\ldots,x_n)$ where S is project success and each xi is a critical success factor i. A common approach to define most critical processes is the Process Quality Management (PQM) method developed by IBM (Hardaker and Ward 1987, Ward 1990). This method is based upon the critical success factors (CSFs) approach. According to Rockart (1979), CSFs are "the limited number of areas in which results, if they are satisfactory, will ensure successful competitive performance for the organisation".

The process structure of the PQM method is too simple since it only provides one level of process analysis. Since most cases imply project process structures that are more complex, here we propose the improvement of the PQM analysis section to provide more depth to complex project structures. This study attempts to analyse this issue in a specific type of software projects: Enterprise Resource Planning (ERP) implementation projects. We propose an extension to the standard PQM method, where we provide a new criticality indicator for complex implementation project process structures.

The paper is organised in the following way. First, we present a software project structure overview. Next, we describe the PQM method and its analysis and detail sections. Then, we present our extended analysis framework for more complex critical processes. We apply this framework to the case of SAP implementation projects and analyse the results found. Finally, we present some final considerations and future work.

2. SOFTWARE PROJECT STRUCTURE OVERVIEW

The structure of project processes is the basis for their planning, monitoring, control and success. According to the Project Management Body of Knowledge (Duncan 1996), a project process is a "series of actions bringing about a result". According to the IEEE standard for software lifecycle processes (IEEE 1996) a software process is "a set of interrelated activities, which transforms inputs into outputs". One of the steps in project planning is determining the sequence in which the processes will be performed. Most project process structures have more than one level (see figure 1). Most common project processes are structured in phases, work packages, work activities and work tasks (quotations from IEEE 1998, p. 3):

- A project phase is a collection of logically related project activities, usually culminating in the completion of a major deliverable (Duncan 1996).
- A work package "is a specification of the work that must be accomplished to complete a work task. A work package should have a unique name and identifier, preconditions for initiating the work, staffing requirements, other needed resources, work products to be generated, estimated duration, risk factors, predecessor and successor work tasks, any considerations for the work, and the completion criteria for the work package – including quality criteria for the work products to be generated". A work product is "any tangible item produced during the process of developing or modifying software".
- A work activity is a "a collection of work tasks spanning a fixed duration within the schedule of a software project. Work activities may contain

other work activities, as in a work breakdown structure. Typical work activities include project planning, requirements specification, software design, implementation, and testing".

- A work task is ”the smallest unit of work subject to management accountability. A work task must be small enough to allow adequate planning and control of a software project, but large enough to avoid micro-management. The specification of work to be accomplished in completing a work task should be documented in a work package. Related work tasks should be grouped to form supporting processes and work activities”.

During the project, work package units are the basic element of project management in terms of planning, control and monitoring (Fairley 1999). Therefore, we focus our analysis in the definition of criticality for this unit.

3. PROCESS QUALITY MANAGEMENT METHOD OVERVIEW

Process Quality Management (PQM) is essentially a top-down, business-lead approach. According to Ward (1990), the purpose of PQM is to enable the management team to:

- Identify the key requirements for improving the overall business performance of the enterprise;
- Conduct an audit on the current investment in IT-based applications and services;
- Identify the principal opportunities and priorities for future investments in IT-based applications and services;
- Review the relevance of current quality improvement projects (if a quality programme exists).

The PQM method has the following steps:

- First step - to define the mission. According to Ward (1990), "the word 'mission' is used in PQM very simply to mean the reason why the particular management team exists; what they are collectively being paid to do. The mission statement is the vision and guiding light for the team, and all those managed/influenced by them."
- Second step - to identify CSFs. Hardaker and Ward (1987) refer that in IBM they define CSFs (applied to PQM) as "what the team must accomplish to achieve its mission".
- Third step - to define the business processes. In this step the relationships between the CSFs and the business processes are established. Therefore, we will have a list of the most critical processes (MCPs). This is made through the creation of a matrix of CSFs versus business processes like the one shown in figure 2.
- Fourth step - to review the IT applications based in the knowledge of the matrix developed.

3.1 Building the matrix of CSFs vs. Business Processes

In the third step, with PQM we generate a matrix of CSFs vs. business processes that has two sections (see figure 2): a 'detail' section, relating CSFs to the business processes, and an 'analysis' section, which indicates the relative 'importance' to CSF performance of each business process, and forms the basis for the establishment of the most critical processes (MCPs). Next, we explain each section.

3.2 Detail Section

The matrix has two axes: one represents the business processes and the other represents the CSFs. The management team focuses in turn on each CSF and considers the following question: "which business processes need to be performed particularly well for us to be confident of achieving this CSF?" (Ward 1990).

Many processes influence the achievement of a CSF but the team must judge which are the truly critical ones. After the first pass a 'sufficiency test' is applied: “If the identified processes are performed well, are they sufficient to manage the CSF in question?” If the answer is 'no' then additional processes need to be defined. This analysis is repeated for all CSFs, each of which will have a different set of critical processes.

3.3 Analysis Section

After the definition of the processes which are most relevant for each CSF, there is the need to make an analysis of priorities, which is done by using three indicators:

- The count - the more important business processes are potentially those which impact most CSFs and a simple count is provided in a column in the analysis section matrix.
- Assignment of a 'quality' rating - A 'quality' rating for each process is provided in a column in the analysis section. This rating is normally the subject of considerable debate among the

	CSFs in ERP implementations / ASAP Processes	Sustained management support	Effective organisational change	Good proj. scope management	Adequate proj. team composition	Meaningful business reengineering	User involvement and participation	Adequate Proj. champion role	Trust between partners	Dedicated staff and consultants	Strong communication	Formalize Proj. Plan/schedule	Adequate Training Program	Preventive Trouble Shooting	Usage of Appropriate Consultants	Empower Decision Makers	Adequate ERP Implem. Strategy	Avoid Customization	.....	(A) Count	(B) Quality Rating	(C) Big burners	(D) Wp criticality
W	**Project Kickoff**																			6			2.8
A	Kickoff Meeting																			5			
T	Prepare for kickoff meeting							1				1								2			
T	Conduct kickoff meeting	1					1	1			1									4			
T	Company wide proj. introduction							1			1	1								3			
A	Project team standards meeting																			4			
T	Prepare for standard meeting							1				1								2			
T	Conduct standard meeting							1		1	1									3			
W	**Quality Check**																			2			1.5
A	Perform quality check and approval																			2			
T	Conduct quality check							1												1			
T	Signoff project preparation phase	1						1												2			

Figure 2- Example of the matrix CSFs versus ASAP processes for project preparation phase.

team. The 'basic' ranking that PQM uses to assess the current quality of the business processes is: A = needs no improvement, B = works well, room for minor improvement, C = functions, several areas for improvement, D = process in place but not functioning, E = embryonic.

- Identifying the 'big burners' - In addition to 'count' and 'quality' a number of teams also identify the 'big burners', those business processes that consume a significant proportion of the money, people or assets for which the team is responsible. The processes to which this test applies are designated with an asterisk in the 'big burners' column.

Now the matrix is complete and the most critical processes can be established.

3.4 Establishing the Most Critical Processes

According to Ward (1990), the most critical processes (MCPs) are those "processes whose performance must be improved if the CSFs are to be managed successfully." The identification of MCPs is made through the creation of a matrix of CSF counts versus process quality rating. The project team must decide which zones have the MCPs.

While all CSFs are equal in importance, the processes vary in their scope and the amount of the team's resources that is devoted to each of them. The general role is that under no circumstances must the quality rating of a 'big burner' process be allowed to slip; and where it has a current quality rating of 'D' or 'C', immediate attention is required in the form of improvement processes.

Rel ij - Relationship between task(i) and CSF(j)	$Rel_{ij} = \begin{cases} 1 & \text{if there exists a relationship} \\ 0 & \text{if there does not exist a relationship} \end{cases}$
Wp.n - Criticality of work package (n) within phase (p). Numtask is the number of tasks of work package n.	$W_{p.n} = \frac{\sum_{i=1}^{NumTask} \sum_{j=1}^{NumCSFs} Rel_{ij}}{Numtask}$

Table 1 – Formal definition of criticality indicator.

4. IDENTIFICATION OF CRITICAL WORK PACKAGES

To be able to identify critical work packages, we have extended the analysis section of PQM in order to deal with more complex project structures involving such packages. To accomplish software project complex structures in the analysis section, we introduce one indicator that we name 'work package criticality' indicator. We define work package criticality as "the importance a work package has with regard to the number of relationships between the software project CSFs and the tasks that compose that work package". The calculation of criticality indicator is worked out as described in table 1.

For each work package we sum all the relationships between CSFs and the tasks of that work package. As each work package has a different number of tasks, we divide the sum by the total number of tasks of this work package in order to compare the value obtained with the value of other work packages.

4.1 An Example

In this section we analyse two work packages of the ASAP preparation phase (phase one): project kickoff and quality check work packages. The matrix of figure 2 represents the different CSFs and the structure of these work packages with the relationship between tasks/activities/work packages and CSFs.

According to the ASAP (1999) documentation, the purpose of the *kickoff* work package "is to formally announce to the company the initiation of the SAP project, which includes the overall goals, detailed task plans, and processes. Consultants, steering committee, senior management, project managers from the company and SAP, and any other implementation partners must be involved. The kickoff meeting is focused on the company as a whole, while the project team standards meeting is focused on the project team". The purpose of the *quality check* work package is to provide final verification of all prior project planning and deliverables from this phase. All issues regarding scope, project environment, and initial technical setup must be addressed. The CSFs used in this analysis are taken from a CSFs unified model in ERP implementations proposed by Esteves and Pastor (2000).

The third step of PQM consisted in the creation of the matrix of CSFs versus ASAP processes (see figure 2). The matrix in figure 2 has been built in the following way. We focused on each CSF and asked this question: Which ASAP processes must be performed especially well for us to be confident of achieving this CSF? Then, we looked at all the processes and decided which ones were important for that CSF.

We used 'open coding' from the Grounded Theory qualitative research method to analyse the ASAP methodology documentation. Grounded Theory is a general methodology for developing theory that is grounded in data systematically gathered and analysed (Glaser and Strauss, 1967). For example, by applying our indicator for work package criticality described above we obtain 2.8 for project kickoff and 1.5 for quality check; thus, project kickoff is more critical, which means that more attention should be put in this work package. Table 2 presents the results of count (according to PQM) and criticality indicators to all the work packages of the *project preparation* phase within the ASAP methodology.

If we only focus on the count indicator we see that, for instance, project standards and procedures (W1.2) work package has a bigger number of CSFs impacts than initial project planning (W1.1) work package. Thus, we could think that the *project standards and procedures* work package is more critical in project preparation phase than the *initial project planning* work package. However, the analysis of the criticality indicator helps to refine the analysis. Through the calculation of the criticality based in our framework we evidence that initial project planning is the most critical work package.

Identifier	Work package	Count	Wp.n Criticality
W1.1	Initial project planning	15	3.63
W1.2	Project standards and procedures	16	3.18
W1.3	Project kick-off	6	2.8
W1.4	Technical requirements planning	7	3.14
W1.5	Quality check	2	1.5

Table 2 - Criticality indicators for the work packages of project preparation phase.

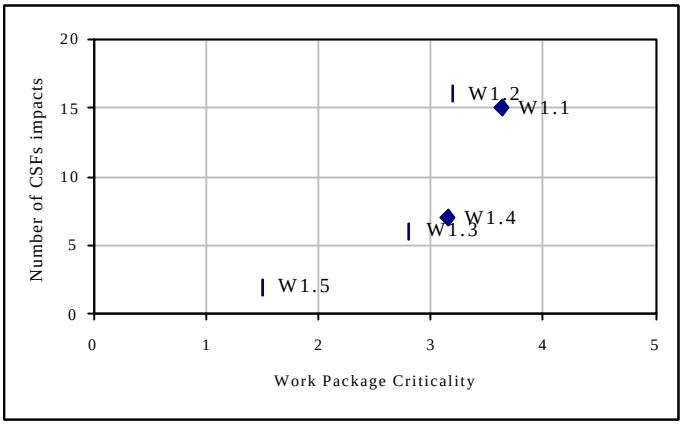

Figure 3 - Most critical work packages in project preparation phase.

Thus, managers should put more efforts in the realization of this work package in order to be successful.

Figure 3 represents graphically the count and the criticality indicator. A position of a work package in the graphic higher and right means this work package is more critical.

We would like to emphasize that this analysis does not mean that the other work packages are not important; we only evidence the importance of the work packages based in a CSFs point of view and relative to the number of tasks in each work package. In the next section, we analyse all the phases of ASAP methodology and the criticality of all its work packages.

5. CRITICAL WORK PACKAGES ALONG ASAP PHASES

In 1996, SAP introduced the Accelerated SAP (ASAP) implementation methodology with the goal of speeding up SAP implementation projects. ASAP was advocated to enable new customers to use the experience and expertise gleaned from thousands of implementations worldwide.

5.1 ASAP Implementation Methodology

The ASAP implementation methodology is a structured implementation approach that can help managers achieve a faster implementation with quicker user acceptance, well-defined roadmaps, and efficient documentation at various stages. This is specifically targeted for small and medium enterprises adopting SAP. The phases of the ASAP methodology, also known as the ASAP roadmap are (ASAP 1999):

1. Project Preparation - the purpose of this phase is to provide initial planning and preparation of SAP project. The steps of this phase help identify and plan the primary focus areas to be considered such as: objectives, scope, plan and definition of project team.
2. Business Blueprint - the purpose of this phase is to create the business blueprint, which is a detailed documentation of the results gathered during requirements workshops/meetings. It will allow the implementation project team to clearly define their scope, and only focus on the SAP processes needed to run the organisation business.

3. Realization - the purpose of this phase is to implement business and processes requirements on the business blueprint. The objectives are final implementation in the system, an overall test, and the release of the system for production (live) operation.
4. Final Preparation - the purpose of this phase is to complete the final preparation, including testing, end user training, system management and cut over activities, to finalize the readiness to go live. The final preparation phase also serves to resolve all open issues.
5. Go Live & Support - the purpose of this phase is to move from a pre-production environment to live production operation. A support organization must be set up for end users to provide long-term support. This phase is also used to monitor system transactions and to improve overall system performance. Finally the completed project is closed.

The structure of each phase is the following: each phase is composed of a group of work packages. These work packages are structured in activities, and each activity is composed of a group of tasks. For each task, a definition, a set of procedures, results and roles are provided in the ASAP roadmap documentation (ASAP 1999).

	Work packages		Count	Wp	Stens
Preparation	W1.1 - Initial project planning	Org	15	3.63	9
	W1.2 - Project standards and procedures	Org	16	3.18	8
	W1.3 - Project kick-off	Org	6	2.8	7
	W1.4 - Technical requirements planning	Tec	7	3.14	8
	W1.5 - Quality check	Org	2	1.5	4
Business Blueprint	W2.1 - Blueprint phase	Org	11	3.31	8
	W2.2 - Organisational change management	Org	11	3.67	9
	W2.3 - Project team training business blueprint phase	Org	6	2.5	6
	W2.4 - Establish development system environment	Tec	9	1.47	3
	W2.5 - Define the business organisation structure	Org	7	3.25	8
	W2.6 - Business requirements definition	Org	14	2.73	7
	W2.7 - Quality check	Org	2	1.5	4
Realization	W3.1 - Project management realization phase	Org	9	2.24	5
	W3.2 - Sustaining the organisation change management process	Org	6	2.54	6
	W3.3 - Conduct project team training	Org	6	2.4	6
	W3.4 - Baseline configuration and confirmation	Tec	8	2.19	5
	W3.5 - System management	Tec	6	1.26	3
	W3.6 - Perform the final configuration and co nfirmation	Tec	8	1.95	5
	W3.7 - Perform ABAP/4 development	Tec	1	1	2
	W3.8 - Develop conversion programs	Tec	3	1.85	4
	W3.9 - Develop application interface programs	Tec	6	1.85	4
	W3.10 - Develop enhancements	Tec	3	0.85	2
	W3.11 - Create reports	Tec	8	2.14	5
	W3.12 - Create forms	Tec	6	2.14	5
	W3.13 - Establish authorization concept	Tec	5	1.57	4
	W3.14 - Establish archiving management	Tec	3	1.75	4
	W3.15 - Prepare end-user documentation and training material	Org	7	2.56	6
	W3.16 - Final integration test	Tec	5	2	5
	W3.17 - Quality check	Org	2	1.5	4
Final Preparation	W4.1 - Project management of the final preparation phase	Org	9	3	7
	W4.2 - End-user training	Org	3	2.4	6
	W4.3 - System management	Tec	4	1.3	3
	W4.4 - Detailed project planning	Org	9	2.11	5
	W4.5 . Cutover to the production system	Tec	6	1.83	4
	W4.6 - Quality check	Org	2	1.5	4
Go Live	W5.1 - Production support	Org	9	3.5	9
	W5.2 - Project end	Org	8	3.2	8

Table 3 - Work packages criticality by ASAP phase.

5.2 Critical Work Packages per Phase

We applied the procedure described in section four to all the phases of ASAP methodology and the result is presented in table 3.

In the *Project Preparation Phase* the most critical work packages are: initial project planning, project standards and procedures, technical requirements. The outcomes of these work packages are the baseline guides for the overall project. The purpose of the initial project planning work package is to allow the start detailed planning for the project. The key elements of the project are defined and the project scope is defined, the project plan is prepared.

In the *Business Blueprint Phase* the most critical work packages are: organizational change management, blueprint phase, and the definition of the business organisation structure. The purpose of this phase is to primarily prepare the business blueprint document for the SAP implementation (Kale 2000) which details the TO BE processes, including the written and pictorial representations of the organisation's future structure and business processes. The purpose of the change management work package is to address the organizational and human resource factors that impact the SAP implementation. It includes a series of change processes that allow the change team to manage organizational risk, accelerate SAP implementation and optimize organizational processes.

In the *Realization Phase* the most critical work packages are: prepare end-user documentation and training material, sustaining the organisation change management process, conduct project team training, project management realization phase. The purpose of change management work package is to conduct periodic project team and organizational risk assessments and to expand the change communications, knowledge transfer, sponsorship and development processes that were initiated during the business blueprint phase. Then, we have training both project team and prepare the training of end-users. The purpose of the project management realization phase is to establish a cycle of project management activities to ensure that the implementation project is on schedule.

In the *Final Preparation Phase* the most critical work packages are: project management of the final preparation phase, end-user training and detailed project planning. The purpose of project management is to perform the established cycle of project management activities in order to keep the implementation project on target. Next come training end users and detailed project planning work packages. The purpose of the detailed project planning is to identify issues that impact the initial plan for production support and cutover prepared in phase 3, and to adapt the plan accordingly (ASAP 1999).

In the *Go Live Phase* the two critical work packages that composed this phase have almost the same criticality. The purpose of production support work package is to provide support to users and maintain optimal system performance. Project end work package is to officially close the project. Any open issues still pending resolution are reviewed and closed.

5.3 Overall ASAP Work Packages Criticality

We also categorized the work packages into organizational and technological work packages. Organisational work packages are related with concerns like organizational structure and culture, business processes and project management. Technological work packages focus on aspects related to the particular ERP product in consideration and on other related technical aspects, such as hardware and base software needs.

Next, we converted the criticality scores of table 3 into a normative scale of ten scores ('stens'). In such scale, range 1-3 ranks as low criticality, range 4-7 ranks as normal criticality and range 8-10 ranks as high criticality.

In general our analysis makes objectively evident that the most critical work packages are those related to organizational aspects. Figure 4 shows how all the organizational work packages have normal and high levels of criticality, while technological work packages have low and normal levels, except for work package W1.4 (technical requirements planning) which purpose is to identify the technical infrastructure needed to implement the SAP system and to clarify customer's expectations. The reason for these results lies in the fact that CSFs are mostly related to an organizational perspective rather than a technological one (Esteves and Pastor 2001). This result is very important for managers. In this respect, ERP implementations are not different from other complex IS projects, for which Felix and Harrison (1984, p. 161) already said: "technical problems can usually be detected and repaired before the system is put in jeopardy. The cost may be high in terms of either budget or schedule, but the repair can be made. Organizational and personnel problems often cannot be redressed, and continue to jeopardize the success of the system itself". Figure 5 shows the overall criticality analysis along the ASAP phases. From the figure we see that the first two phases, Project Preparation and Business

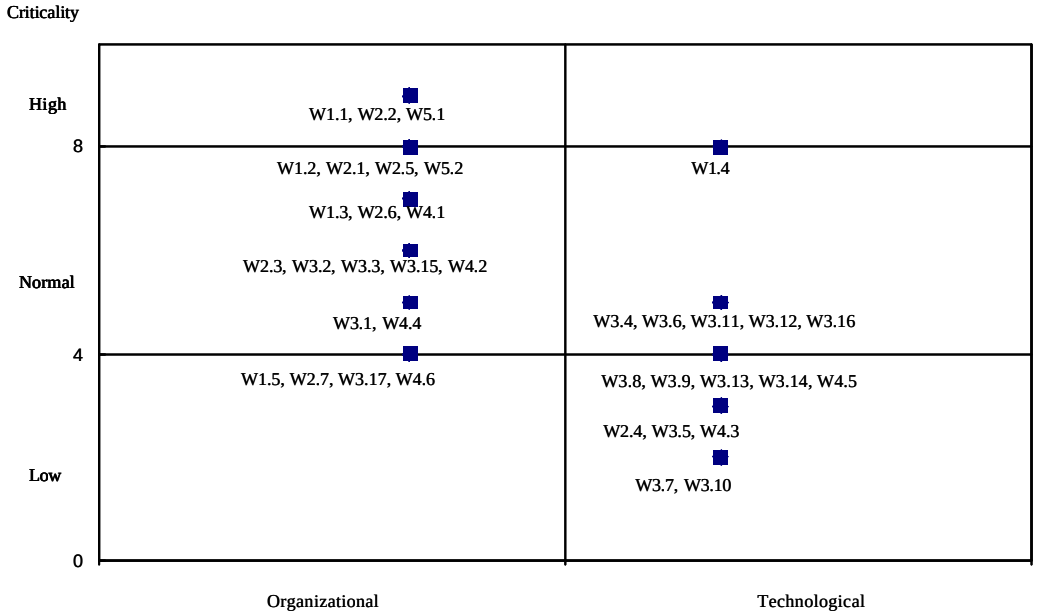

Figure 4– Analysis of organizational versus technological ASAP work packages.

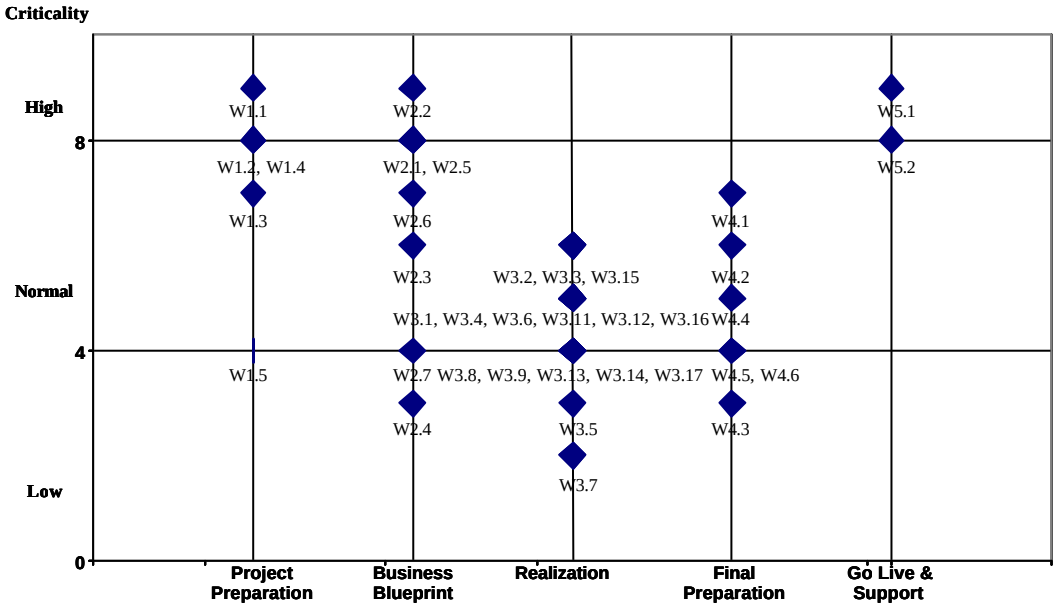

Figure 5- Criticality Analysis along ASAP phases.

Blueprint, together with the last phase Go Live and Support, are the ones with more criticality.

In the initial phases is when most of strategic decisions are made and the business model is defined. The last phase represents the last effort to customize the SAP system according to the need of the organization. Obviously, project managers should put more attention and effort in these phases in order to achieve a successful SAP implementation project.

6. CONSIDERATIONS AND FUTURE WORK

This study proposes an extension to the PQM method in order to analyse the most critical processes in complex project structures such as a SAP implementation project. We proposed a new

indicator for processes criticality analysis and then we applied it to the ASAP implementation methodology structure in order to define its most critical processes. Although we have applied the analysis extension proposed in this paper to a SAP implementation project, it can be applied to other ERP systems or other software projects that have a complex software project structure, as the one explained in this paper.

The analysis described here for work packages criticality can also be applied to activities, since they have the same dependence of tasks. In that sense, activity criticality analysis can be calculated using the same procedure of work packages. The criticality indicators described in this study are not intended to substitute the PQM analysis; instead, they attempt to give more analytical expression to the work done by the project teams when they define the most important tasks based in a CSFs approach. The analysis of this indicator:

- Will help managers to plot and prioritise for attention on those most critical components,
- Will help in the better allocation of organisation resources ,
- Will help to avoid 'bottlenecks' in software implementation projects and,
- Will help the development of new ERP implementation methodologies.

Future research will be focused in the validation of this indicator. We pretend to use the Case Study method to validate it (Yin 1994). At this stage of research, we consider that all the relationships between CSFs and the ASAP tasks have the same weight. In the future, we want to improve this by adding the relevance of each CSF in each phase (see Esteves and Pastor 2001). Finally, we also attempt to extend the results presented here to other ERP systems and implementation methodologies.

REFERENCES

ASAP 1999. “Accelerated SAP Methodology (ASAP) Roadmap”, ASAP documentation, version 4.5, SAP AG company, 1999.

Association of Project Management (UK) (2000). APMP syllabus, abridged glossary of project management terms, January 2000.

Duncan W. 1996. “A guide to the Project Management Body of Knowledge”, Project Management Institute (PMI) Standards Committee, 1996.

Esteves J., Pastor J. 2000. "Towards the Unification of Critical Success Factors for ERP implementations", 10th Annual Business and Information Technology (BIT) Conference, November 2000.

Esteves J., Pastor J. 2001. “Analysis of Critical Success Factors Relevance along SAP Implementation Phases”. Americas Conference on Information Systems (AMCIS), 2001.

Fairley R. 1999. ”Managing by the Numbers: A tutorial on Quantitative Measurement and Control of Software Projects”, International Conference on Software Engineering (ICSE), 1999, pp. 677-678.

Felix R., Harrison W. 1984. “Project Management Considerations for Distributed Processing Applications”, MISQ Quarterly, vol. 8, nº. 3, September 1984, pp. 161-170.

IEEE 1996. “IEEE/EIA Standard – Industry Implementation of ISO/IEC 12207: 1995, Standard for Information Technology – Software Life Cycle Processes”, IEEE Computer Society, 1996.

IEEE 1998. “IEEE standard for Software Project Management Plans”, IEEE Std 1058-1998, IEEE Computer Society, December 1998.

Glaser B., Strauss A. 1967. "The Discovery of Grounded Theory: Strategies for Qualitative Research", first edition, Aldine, 1967.

Hardaker M., Ward B. 1987. “Getting Things done: How to Make a Team Work”, Harvard Business Review, Nov/Dec 1987, pp. 112-119.

Kale V. 2000. “Implementing SAP R/3: The Guide for Business and Technology Managers”, Sams Publishing, 2000.

Pinto J., Slevin D. 1987. "Critical Factors in Successful Project Implementation", IEEE Transactions on Engineering Management, vol. EM-34, nº. 1, February 1987, pp. 22-27.

Rockart J. 1979. "Chief executives define their own information needs", Harvard Business Review, March - April 1979.

Savolainen V. 1991. "Definition of favourable atmosphere for effective IT decisions", Sol HG. and Vecsenyi J. (eds), Environments for supporting decision processes, Elsevier Science Publishers B.V. (North Holland), 1991, pp. 129-140.

Shenhar A., Wideman R. 2000. "Optimizing Project Success by Matching PM Style with Project Type", Project Management Forum, October 2000.

Ward B. 1990. "Planning for Profit", chapter 5, in "Managing Information Systems for Profit", edited by T. J. Lincoln, John Wiles & Sons Ltd., 1990.

Yin R. 1994. “Case Study Research, Design and Methods”, 2nd ed., Sage Publications, 1994.