“Effective Customer Requirements Management using an Information Supply Based Model”

SECTION 3. General issues in management

Ofer Zwikael (New Zealand), Oleg Tilchin (Israel)

Effective customer requirements management using an information supply based model

Abstract

Managing customer requirements is a crucial process in any project, especially when these requirements usually keep changing throughout a project life cycle. Despite the high importance of customer requirement management in any project, there is no available customer requirements management tool that has an effective interface with project management tools. This paper presents a new information supply model for integrating customer requirements management with project management. The model proposed, an Information Supply Tool (IST), can be used to manage customer functional requirements for product design process, and to improve communication among managers, project team members, customers, and other stakeholders. The aim of IST model is to supply information about the product design processes for managers to support decision making. It also provides information about the design process status on all granulation levels. The benefits of this model include using resources rationally for projects, exclude duplication of jobs, increase quality of project decisions, and better support achieving customer requirement. Senior managers will be able to evaluate the current state of the project against desired product functions, and to analyze the impact of managerial decisions. Managers will be able to model their managerial actions and receive information about the possible consequences to improve customers benefit management. The model is introduced in the paper, and a case study is presented.

Keywords: project analysis, IT management, management of technological innovation and R&D.
JEL Classification: O22, M15, O32.

Introduction

The outcome of any project is a new product or service (Kerzner, 2006). This outcome has to satisfy customer requirements, as defined at the beginning of a project and keep changing during its execution. Requirement definition starts with the customer, who defines his first octopi wishes, going through realizing the limitations and narrowing them into possible functional requirement, and ends with detailed technical requirements developed by the project team. Managing the requirement changes is performed during the project execution (PMI, 2004).

In order to provide the customer with a product that satisfies his requirements, two major processes must be executed. The first process includes the transformation of customer functional requirements into technical detailed requirements. This process is performed by the project team in the beginning of the project, and is then approved by the customer. The second process includes configuration management and change management during project execution in order to deal with new and changing requirements. Due to its complexity, requirement management has to be computerized and aligned with the project management software.

In order to investigate the existence of these processes in project management software packages, an analysis of the most popular project management software packages, such as Microsoft Project (Stover, 2004), Super Project, Plan View, Micro Planner, AEC Software has been conducted. It has been found that some drawbacks of these tools exist while dealing for implementing requirement management principles:

1. Narrow content – project management software packages are usually limited to scheduling, scope, and cost management. These software packages do not assist project managers with forming necessary product functions and characteristics.
2. Lack of information supply – project managers cannot always get information about the project on lower levels of its granulation. This may lead to a significant duplication of jobs, irrational use of resources or inefficient project decisions.
3. Project control – managers get the information about various states of readiness of tasks exclusively from project team members. Such information is not always correlated with the state of design of functions and characteristics of the product and its blocks or with the state of the product design at the lower (relative to the level of tasks) levels of granulation. As a result, managers cannot juxtapose the current state of the project with desired functions and characteristics of the product.

4. Impact of managerial decisions – a project manager cannot model his managerial actions and receive information about the results of his managerial actions. Hence, he/she cannot evaluate the accuracy of his own managerial decisions.

These drawbacks allow us to reach a conclusion that the indicated project management software packages do not sufficiently assist in the collection of information from project participants regarding product progress. Information supply from project participants, which is highly aligned with the product structure, is crucial for achieving a quality product. Hence, a new requirement management approach is needed, in order to provide an effective requirement management model for new product development. This paper presents a new model for the collection and maintenance of requirements information in projects, which is useful for both project monitoring and control and product progress analysis. The introduction of the new approach will follow a literature review.

1. Literature review

A project is defined as any series of activities and tasks that have a specific objective to be completed within certain specifications, have defined start and end dates and funding limits (Kerzner, 2006). Project success is usually measured according to four dimensions: 1) schedule overrun (from the approved due date), 2) cost overrun (from the approved budget), 3) project performance, (in comparison to the agreed upon objectives/goals), and 4) customer satisfaction of project outputs (Lim & Mohamed, 1999; Zwikael & Globerson, 2006; Kerzner, 2006; Voetsch, 2004; Kerzner, 2006). Procaccinoa and Verner (2006) found a consensus among the project managers, indicating that delivering a system that meets customer/user requirements and works as intended through work that provides a sense quality and personal achievement are important aspects that lead to a project being considered a success. Na et al. (2004) remind that empirical studies of US software firms support the adoption of user requirement analysis techniques in software project management.

Requirement management is included in project management and software development literature. For example, the Project Management Body of Knowledge (PMBOK®) identifies nine knowledge areas in which a project manager has to act during the project, including integration, scope, time, cost, human resources, communications, quality, risk and procurement management. Requirement management is covered in the scope management knowledge area (PMI, 2004). Out of these nine, the scope knowledge area makes sure that the project includes all and only the necessary work required to complete the project successfully. This knowledge area includes knowledge and tools relating to product scope (features and functions that are to be included in a product or service) and project scope (work that must be done in order to deliver a product as specified). Processes within this knowledge area are: scope planning, scope definition, Work Breakdown Structure (WBS) creation, scope verification and scope control.

Tools included in this knowledge area are: a scope statement and WBS, which describes all work packages to be performed during the project (PMI, 2004).

Requirement management is critical especially in software development projects. For example, Han and Huang (2007) found that the "requirement" risk dimension is the primary area among six risk dimensions. Requirement management includes the following areas: continually changing system requirements, system requirements not adequately identified unclear system requirements and incorrect system requirements.

Requirement management starts during the initiation phase of a project, when the customer has to introduce his best wishes and the project manager or salesman has to set the technical limitations. Dag et al. (2005) claim that the manual linkage between customer requirements and product requirements is routinely performed, cumbersome, time-consuming, and error-prone. Once an agreement was reached regarding the expected outcome, a project charter or a contract may be signed. Andriole (1998) claims political forces are highly involved in the creation of the initial requirements list.

Requirement management then continues with the planning phase of the project, performed by the project team. Zwikael and Globerson (2004) found that scope planning is one of the processes performed with the highest extent of use during the planning phase of a project. Requirement planning includes the developing of a Work Breakdown Structure (WBS) which identifies all project activities that should be performed in order to achieve customer's requirements. Reifer (2001) suggests not to completely spelling out requirements before development.

Requirement control is performed in the execution phase of a project. This process contains the update of the planned requirements and the adding of new ones. The accomplishment of project activities influences the progress of the project.

Dick et al. (2005) suggest a new approach for project control by specifying process in such a way that progress can be measured in terms of achievement rather than effort expended. The outcome of each process activity is specified in terms of the status of information items and the relationships between them.

Frame (2002) claims that project execution is impossible without a project management software package. Moreover, the requirement management processes require a tool that is interfaced with the project management software. The measurement of project progress has also been highlighted by Dick et al. (2005). They suggest a new model according to which progress can be measured in terms of achievement rather than effort expended. The outcome of each process activity is specified in terms of the status of information items and the relationships among them.

According to the stated literature we can claim that any type of requirement management tool has to provide the following capabilities:

1. Support a complex structure – the complex project and product structure consists in the coordinated management of the design process at various levels of its granulation and also at various levels of product structure.
2. Support dynamics management – an automated analysis of the project status, including requirement changes, collected and presented in online regime.
3. Improve management decision making – the ability to provide the required functions and characteristics of the product during the optimization of the necessary resources.
4. Increase depth (working out in detail) of management – determined by the granulation levels of the project works and the structure levels of organization which executes the project.
5. Increase independency – independent realization of a certain part of project management activities.

Based on this list, a new model will be introduced and demonstrated in the following sections.

2. The approach to effective requirements management

This section introduces a new requirement management model, which is aligned with both project management tools and product structure. This approach proposes the introducing a new model which represents a product and a process of its designing. The creation of a functional requirement control mechanism enables controlling functions and characteristics formation of a product or a service.

2.1. The project model. The model provides completeness of information supply for all project management processes (Globerson and Zwikael, 2002). From this point of view, it is possible to talk about the completeness of such representation. The project model includes three components which are as follows: 1) the simplified format of product design goal, 2) the product model, and 3) the product design process. The following paragraphs describe three components of the suggested representation.

2.1.1. The simplified format of product design goal. The input for this model is customer requirements for the project output, as has been requested by the customer and agreed by the project team. Project requirements are included in the project goal. Product goal sets out what and how the product must do and which resources are necessary for it, according to the following format:

$$g = \langle F, C, R \rangle, \quad (1)$$

where g is a product design goal (project goal), F is a set of the Product Functions (what should be done), C is a set of the Product Characteristics (how it should be performed), $\langle F, C \rangle$ includes the quality component of the goal. Functions and characteristics express product functional requirements, R is a set of resource kinds (denomination and quantity required for product's design). R is quantity component of the goal.

The specified format may serve as a constructive basis for efficient product scope management. The format provides understanding of which product will be produced at the end of the project. This understanding is similar among all members of the project team, as well as among the project manager, stakeholders and the customer (Barkley and Saylor, 2001).

2.1.2. The product model. The second component of the model defines the product's structure. Let us formally present the model P of the product being designed as a graph:

$$P = (B, W, G), \quad (2)$$

where $B = \{b_0; b_{11,\dots}, b_{ik,\dots}\}$ is the nodes to which the product and the product blocks correspond; b_0 is the node to which corresponds the product; $\{b_{11}, \dots, b_{ik}, \dots\}$ is the set of nodes to which corresponds the set of IDs of product blocks; W is a set of edges, $(b_{ik}, b_{jl}) \in W$, $i < j$, if block b_{jl} is subordinated to the block b_{ik}; i, j are levels of the model hierarchy; G is a set of nodes weights. We will describe node weight, accordingly (1), as:

$$g(b_{ik}) = \langle F_{bik}, C_{bik}, R_{bik} \rangle,$$

where $g(b_{ik})$ is block b_{ik} design goal. The block b_{ik} design goal is sub goal of product b_0 design goal.

F_{bik}, C_{bik}, R_{bik} are lists of denominations of block functions, block characteristics, and resources required for design, respectively. The product's model is described in Figure 1.

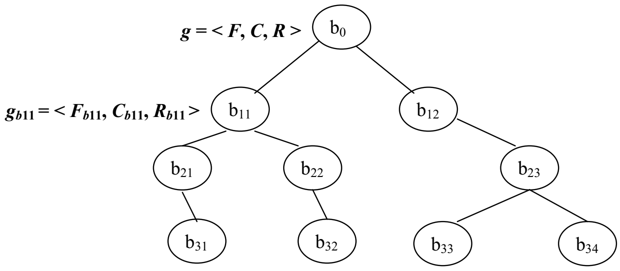

Fig. 1. The product's model 3.1.3.

2.1.3. The product design process. The goal of product design (1) is achieved by means of product design process. The product design process represents the project scope development (Schalbe, 2002). The project scope is presented as Work Breakdown Structure (PMI, 2004) including the following characteristics described in the WBS development standard (PMI, 2001):

- it is representative of work as an activity and this work has a tangible result;
- it is arranged in a hierarchical structure. For example, it is possible to distinguish the levels of tasks, subtasks;
- it has an objective or tangible result, which is referred to as a deliverable.

According to this standard, the model of product design process is represented by the following oriented graph:

$$E = (Z, W, V_Z), \tag{3}$$

where Z is a set of tasks identifiers; W is a set of arcs, $(z_i, z_j) \in W$, if the result of solving task z_i is used by task z_j, then the arc reflects the logical connection of the tasks; V_Z is a set of weights of nodes. The weight of each node contains: ID product block, during the design of which the task is being solved, g_z is the goal of solving task z.

The goal of task solving is the part of the corresponding product block goal.

According to the introduced format (1) the goal of task solving has a form:

$$g_z = < F_z, C_z, R_z >, \tag{4}$$

where: $< F_z, C_z >$ is a quality component of task goal g_z; F_z, C_z is a list of denominations of block functions and characteristics, received as a result of solving a given task; $F_z \subset F_b, C_z \subset C_b$.

R_z is resources (denomination and quantity), required to solve the task; $R_z \subset R_b$.

Resources for task solution are a part of resources necessary for the design of the corresponding block. For example, the model of the product design process for block b_{11} (Figure1) is presented in Figure 2.

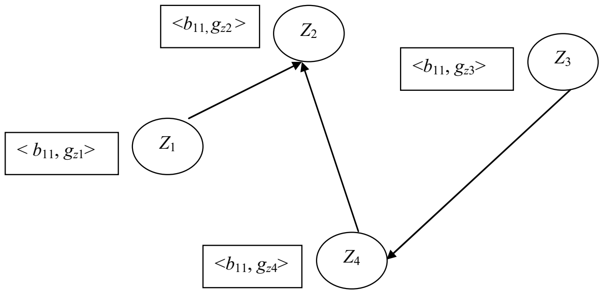

Fig. 2. The model of the product design process for block b_{11}

Thus, the project model is the representation of a product (Figure 1) and its process design (Figure 2).

2.2. The functional requirement management model. Let us introduce the concepts of "event" and "situation" that characterize the execution state of the design process according to interdependence of tasks. These concepts provide event-driven specific of the suggested requirement control mechanism.

Events characterize execution state of a separate task. The necessary condition for occurrence of an event "a task is realized" is a task readiness to execution. The task readiness to execution is determined by realization of some tasks. These tasks are used by a given task. For example, a necessary condition for occurrence of an event "a task is realized" for task z_2 is realization tasks z_1 and z_4 (Figure 2).

Situations characterize the state of execution of all tasks. Necessary condition for occurrence of situation "the tasks are realized" is occurrence of an event "a task is realized" for every task. For example, a necessary condition for occurrence of situation "the tasks are realized" is occurrence of an event "a task is realized" for tasks z_1, z_2, z_3, z_4.

The term "task result" is a measure of achieving a quality component $< F_z, C_z >$ of goal (4) of solving the task. This dynamic indicator characterizes the state and changes during the task solving as well as after its completion. In this connection, we distinguish between:

- Intermediate result (progress) reflecting a state – contains a list of denominations functions and characteristics of the task which are designed during its solution by the reviewed moment in time.
- Intermediate result (progress) reflecting changes – contains a list of performed new functions and characteristics that were not initially set in the task goal.

- The result is achieving at a certain moment in time the quality component $< F_z, C_z >$ of goal (4) of solving the task, or another result satisfying the designer.

Thus, evaluation of the task solving process in time is done by juxtaposing the result and quality component of the goal. For example:

a. The intermediate result, reflecting the state:

$Rez_{\ state(z)} = < F^*, C^* >, F^* \subset F_z, C^* \subset C_z$

b. Intermediate result, reflecting the changes:

$Rez_{\ change(z)} = < F_{new}, C_{new} >$

c. The end result:

$Rez_{\ end(z)} = < F_z, C_z >$,

or results satisfying the designer:

$Rez_{\ end(z)} = < F_z, C_z, F_{new}, C_{new} >$,

$Rez_{\ end(z)} = < F^*, C^*, F_{new}, C_{new} >$,

$F^\wedge = F^* \cup F_{new}$, $C^\wedge = C^* \cup C_{new}$; $F^\wedge \neq F_z$, $C^\wedge \neq C_z$.

Parameter "task result" characterizes sufficient condition for occurrence of an event "a task is realized". Sufficient condition for occurrence of an event "a task is realized" is receipt of a task result. Sufficient condition for occurrence of situation "the tasks are realized" is receipt of a task result for every task. The degree of task readiness is determined according to necessary and sufficient conditions for occurrence of an event "a task is realized". The degree of all task readiness is determined according to necessary and sufficient conditions for occurrence of situation "the tasks are realized". Introduced concepts of "a task is realized" event and "the tasks are realized" situation are background of construction of the functional requirement control mechanism. The functional requirement control mechanism carries out control of the tasks readiness degree at the reviewed moments in time. It opportunely informs designers about the results derived from the control process. Consequently responsiveness and quality of functional requirements management are provided.

3. Implementation example – the development of the information supply tool

In order to demonstrate the use of this model, we chose the development of this tool as a simple example. The aim of the Information Supply Tool (IST) is a provision of information on the product design process. The basis of the IST design is its project model. Construction of the IST project model is executed according to suggested approach. It involves the construction of the IST model and the formation of the design process model, which will be described in the following sections.

3.1. Construction of the IST model. *A. Structuring of IST blocks.* The IST model is shown in Figure 3.

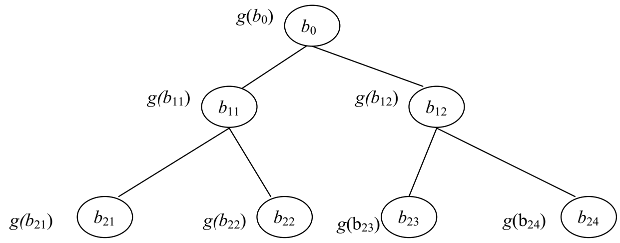

Fig. 3. The IST model

Determination of the functions for IST and for its blocks: $F(b_0)$ – information about design process of some product and of its blocks (the functions of IST); $F(b_{11})$ – information about state of the design process of some product; $F(b_{12})$ – information about results of managerial actions for change of the product design process; $F(b_{21})$ – information about state of the design process related to quality goal component; $F(b_{22})$ – information about of the design process resource state; $F(b_{23})$ – information about design process change which is executed by project management tools (PMT); $F(b_{24})$ – information about change of a design process which is executed by project participants.

C. Determination of IST characteristics structure. The IST model has three levels (Figure 3). According to this structure three levels of the characteristics structure are determined. The characteristics are divided into base and derivative ones. The characteristics every level (except characteristics of the base level) are result composition of the previous level characteristics. The characteristics structure is shown in Figure 4.

The base characteristics (first level) are: c_1 is flexibility, c_2 is velocity, c_3 are mutuality and number of address, c_4 is fullness, c_5 is simplicity, c_6 is accuracy.

The derivative characteristics (second level) are: c_7 is responsiveness, c_8 is usability, c_9 is transparence of the design process.

The derivative characteristic of the third level is: c_{10} is appeal of the IST use.

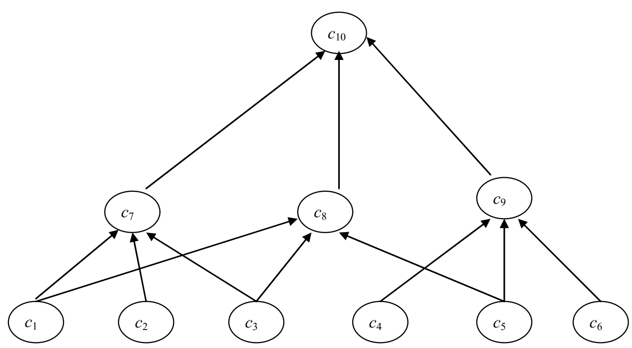

Fig. 4. The characteristics structure

D. Determination of the quality goal components for IST blocks. The quality goal components for IST blocks (Figure 4) are:
$< F(b_0); c_{10} >$
$< F(b_{11}); c_7, c_8, c_9 >$
$< F(b_{12}); c_7, c_8, c_9 >$
$< F(b_{21}); c_1, c_2, c_3, c_4, c_5, c_6 >$
$< F(b_{22}); c_1, c_2, c_3, c_4, c_5, c_6 >$
$< F(b_{23}); c_1, c_2, c_3, c_4, c_5, c_6 >$
$< F(b_{24}); c_1, c_2, c_3, c_4, c_5, c_6 >$

3.2. Formation of the design process model for block b_{21}. The quality goal component for block b_{21} is $< F(b_{21}); c_1, c_2, c_3, c_4, c_5, c_6 >$ = < Information about state of the design process related to quality goal component; flexibility, velocity, mutuality and number of address, fullness, simplicity, accuracy>.

The task z_1 name is "Information in on-line regime of managers about state design process of some product relatively quality goal component".

The task z_1 goal is $g(z_1) = < F_1(b_{21}); c_2, c_4, c_6 >$, $F_1(b_{21}) \subseteq F(b_{21})$.

The task z_2 name is "Information of managers about state design process of some product relatively quality goal component over time period".

The task z_2 goal is $g(z_2) = < F_2(b_{21}); c_4, c_5, c_6 >$, $F_2(b_{21}) \subseteq F(b_{21})$.

The task z_3 name is "The mutual information in on-line regime of designers about state design process of some product relatively quality goal component"
The task z_3 goal is $g(z_3) = < F_3(b_{21}); c_1, c_3, c_5 >$, $F_3(b_{21}) \subseteq F(b_{21})$.

The fragment of the design process model for block b_{21} is shown in Figure 5.

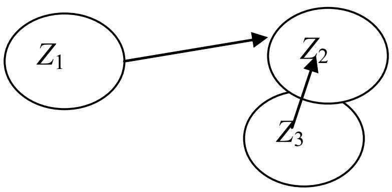

Fig. 5. The fragment of the design process model for block b_{21}

4. The model's alignment with project management approaches

The proposed approach demands realization of the following procedures defined for all phases distinguished by the PMBOK (PMI, 2004). Each project goes through four stages – initiation, planning, execution, and closure. The next paragraphs demonstrate the benefits from using the new model and its alignment to each of the project phases.

4.1. Initiation. Initiation is the phase of formally authorizing a new project. This phase links the project to the ongoing work of the performing organization. Projects are typically authorized as a result of one or more of the following: a market demand, a business need, a customer request, a technology advance or a social need. In this phase, this model assists with formation of strategic intention with estimation of resource capabilities.

4.2. Planning. Planning processes define and refine objectives and select the best of the alternative courses of action to attain the objectives that the project was undertaken to address. Planning is of major importance to a project because the project involves doing something that has not been done before. In this phase this model assists with:

- construction of the project model being designed;
- formation of design process model;
- planning of distribution of project tasks by organizational divisions;
- building of specific dynamic schedule of design process. This building provides dynamic distribution of interrelated tasks into a sequence of time intervals with simultaneous forming of their dynamic groups. The sign of grouping is the proximity of tasks relative to the necessary resources (Tilchin, 2004);
- resource planning.

The aforementioned planning functions correspond to the nomenclature of planning processes suggested in Globerson and Zwikael (2002).

4.3. Execution. Executing processes coordinate people and other resources to carry out the plan in order to perform the project. In this phase this model assists with:

- information supply of managers and designers about state requirements to product at the reviewed moments in time;
- multi-criteria dynamic optimization of the design process model;
- dynamic scheduling of the task groups taking into account proximity of tasks and changes of resource capabilities;
- dynamic redistribution of the resources.

This tool can also assist with controlling processes, such as designing an organizational process for the project and with formation of functions and characteristics of the product being designed.

4.4. Closure. Closing processes formalize acceptance of the project and bring it to an orderly end. In this phase the model assists with evaluation of project completeness and success.

Conclusion

This paper introduces an information supply approach for effective requirements management in projects. This approach provides the possibility of 1) automation of project requirement information supply management, and 2) high-level communication among designers and managers, project team members and stakeholders.

A design process of an information supply tool which provides functional requirements management is presented. This process is realized according to the suggested approach. In frame of the suggested approach, the content of the project management jobs is determined. The jobs are performed at every stage of the project process.

This approach presupposes 1) detailed development of each of IST blocks, 2) extension of possibilities of the suggested IST for realization of all project management processes, and 3) use of knowledge management means for improvement of "managerial abilities" of the proposed IST.

Research limitation includes the model being theoretical, with no real applications working in organizations. In order to deal with these limitations future research may include the development of a requirement management tool, which is based on the principles introduced in this paper, and the implementation of this tool in a pilot organization.

References

1. Andriole, S. (1998). The politics of requirements management. IEEE Software, 15, 6; p. 82.
2. Barkley, B.T., Saylor, J.H. (2001). Customer Driven Project Management: Building Quality into Project Processes McGraw- Hill Professional, 2nd edition.
3. Dag, J.N.; Regnell, B.; Gervasi, V., Brinkkemper, S. (2005). A linguistic-engineering approach to large-scale requirements management. IEEE Software, 22, 1; p. 32.
4. De Mello, L.S.H., Sanderson, A.C. (1990). AND/OR Graph Representation of Assembly Plans, IEEE Transaction on Robotics and Automation 6 (2), 188-199.
5. Dick, A.J., Hull, M.E.C., Jackson, K. (2005). Specifying process and measuring progress in terms of information state. The Journal of Systems and Software, 76, 3; p. 311.
6. Dick, A.J.; Hull, M.E; Jackson, K. (2005). Specifying process and measuring progress in terms of information state. The Journal of Systems and Software, 76, 3; p. 311.
7. Frame, J.D. (2002). The New Project Management: Tools for an Age of Rapid Change, Complexity and other Business Realities, 2nd Edition, Jossey-Bass.
8. Globerson, S., Zwikael, O. (2002). The Impact of the Project Manager on Project Management Planning Processes. Project Management Journal, 33, 3.
9. Han, W., Huang, S. (2007). An empirical analysis of risk components and performance on software projects. The Journal of Systems and Software, 80, 1; p. 42.
10. Kerzner, H. (2006). *Project Management: A Systems Approach to Planning, Scheduling and Controlling,* 9th edition, John Wiley and Sons.
11. Lim, C.S., Mohamed, M. (1999). Criteria of project success: An exploratory re-examination. International Journal of Project Management, 17, 4, 243-248.
12. Na, K., Li, X., Simpson, J.T., Kim, K. (2004). Uncertainty profile and software project performance: A cross-national comparison. The Journal of Systems and Software, 70, 1-2; p. 155.
13. PMI Standards Committee. (2004). A Guide to the Project Management Body of Knowledge, third edition, Newtown Square, PA: Project Management Institute.
14. Project Management Institute (2001). Practice Standard for Work Breakdown Structures. Newtown Square, PA: Project Management Institute.
15. Reifer, D.J. (2000). Requirements management: The search for nirvana. IEEE Software. Los Alamitos: May/Jun. Vol. 17, Iss. 3; p. 45.
16. Schalbe, K. (2002). Information Technology. Project Management, second edition. Course Technology.
17. Stover, T. (2004). Microsoft Office Project 2003 Inside Out, Microsoft Press.
18. Tilchin, O. (2004). Formation of Time Task Packages Proceeding of the Winter International Symposium on Information and Communication Technologies, Cancur, Mexico. Published by the Computer Sciences Press, Trinity College, Dublin, Ireland. ISBN: 0-9544145-3-5.
19. Voetsch, R.J. (2004). The current state of project risk management practices among risk sensitive project management professionals. The George Washington University PhD, 559 pages; AAT 3112236.
20. Zwikael, O. & Globerson, S. (2004). Evaluating the Quality of Project Planning: A Model and Field Results. International Journal of Production Research, 42, 8, p. 1545-1556.
21. Zwikael, O., Globerson, S. (2006). From Critical Success Factors to Critical Success Processes. International Journal of Production Research, 44, 17, 3433-3449.