

Conceptual model and concepts for ERP system change implementation planning

Igors Mikulovs, Janis Grabis

Tele2 AB, Riga Technical University, Meza street 1/3, Riga, LV-1048, Latvia
igors.mikulovs@tele2.com, grabis@rtu.lv

Abstract: The paper investigates the existing software change management approach at Tele2 and uses accumulated experiences as a basis for conceptualizing the planning of changes in packaged applications. Tele2 is one of Europe's leading telecom operators, offering mobile services, fixed broadband and telephony, data network services, cable TV and content services with 37,6 million customers in 11 countries. IT operations and services are being supported within Tele2. Change management, especially scheduling of changes, is known to be one of the most challenging problems in managing of IT operations. The task of finding an optimal change implementation plan is difficult considering many variables which influence the decision on resource allocation to implementation of change requests. Many IT companies do planning manually, re-allocating and re-planning resources constantly. This paper presents the results from in-depth study of change management and planning practices with the aim to identify concepts and their attributes influencing the change implementation plan. The conceptual model is elaborated based on the collected concepts. The objectives of change planning such as implementation cost and time minimization are also represented in the conceptual model. The one of the most important features represented in the conceptual model is re-planning of change implementation activities in response to changing development priorities and external events such as new updates of the packaged software. The conceptual model and concepts are validated against four similar studies. Construction of the conceptual model also allowed to identify several directions for future improvement of change management and scheduling practices. The described conceptual model will serve as a basis for further work on design and implementation of change planning decision support system.

Keywords: Change management, conceptual model, change implementation planning, release planning.

Introduction

Enterprise resource planning (ERP) applications were one of the fastest growing and most profitable areas of the software industry during the late 1990s (Sprott, 2000). Companies worldwide have invested billions of dollars in implementing these systems (Bailor et al., 2006). The adoption of ERP systems has been the focus of substantial research in recent years (Chuck et al., 2010). Although early research has covered many aspects of ERP implementation, it has focused mainly on the implementation stages of the ERP lifecycle. As with other types of information systems, many maintenance activities must be carried out, and issues need to be resolved after an ERP system has become operational (Daveport, 1988).

Corrective, adaptive, preventive and perfective (Lientz et al., 1981) changes should be introduced during the ERP maintenance, and these changes are frequently initiated by Change Requests (CR). Implementation of the change is driven by a number of factors, for example: priority, business case, planned production dates, user needs. At some point of time there could be a large number of outstanding CR, however implementation of those is always limited by resources and other constraints.

The actual problem is to plan and distribute resources (human, time, financial) over the input CR list and to elaborate a time plan and release date for particular CR list considering different factors, which impacts decision on CR processing order. Stakeholders might be interested in quickest as possible implementation, or cost saving implementation, or only interested in precious CR release date communication. Currently many companies drive planning manually, re-allocating and re-planning resources when new CR priorities are changes, resource pool is changed, new urgent CR incomes, or low priority becomes not relevant. Due to changes in resources or CR priorities, some of CR can mismatch requested production dates should be reported and confirmed to next release date. Re-allocating work plan there should be considered already performed work, and interruption penalties when specialist break work in particular activity with following return to it. All this however becomes unmaintainable at certain point of time, due to significant amount of parameters to consider, downtime of the work increases, some activities become uncontrolled, as result overall efficiency of planned activities drops, IT companies do not perform anymore in optimal way.

The CR list needs to fulfil the interests of various stakeholders and takes many variables into consideration. Several scholars have presented lists of such variables, including: importance or business value, stakeholder preference, cost of development, requirement quality, development risk and requirement dependencies (Carlshamre, 2002; Firesmith, 2004; Greer et al., 2004; Weerd et al., 2006).

In order to automate this task, an algorithm for allocation of the CR list to available resource subject to the relevant factors is required. That would allow change/release managers to reduce manual work, to make more

efficient decision and to better control the CR flow over the implementation period. The valuable benefit would be a possibility to communicate correct production dates for the CR despite changes of the initial plan and other variations.

This paper aims to establish a common basis for future development of this CR allocation algorithm. That comprises definition of fundamental concepts utilized in the change planning during the software maintenance. This model is a result of study of the manual change request planning within Tele2 IT division. The subject researched organization has the market-driven requirement engineering processes (Carlshamre et al., 2001). The paper also defines requirements for CR processing list and discusses factors affecting ERP system change implementation. The conceptual model can assist in CR development; add value to automated change request planning. It can provide guidance to organization on building or improving their change management process.

Methodology

The analysis process consisted of two stages. Initially, we identified common properties of the system and process studied by conducting brainstorming sessions and collecting relevant data. CR planning concepts were individually analysed, and added to a draft conceptual model. All information was gathered using different sources, change/release planning documentation, system specifications, operational procedures and interviews. The conceptual model is represented using UML.

Collection and validation of data consisted of collection and scrutinizing of relevant documents (Tele2 internal documents, 2012). With the help of these documents and interviews, requirements for factors which should be considered during CR planning are stated. Each requirement was sent for approval to a representative change/release manager of the particular system. The missing information was supplemented by interviews. Consequently work was validated in the sense that every stakeholder was involved in study scrutinizing and finally the requirements and model was approved. Additionally, the conceptual model and findings were validated by cross-examination with the existing literature in order to check ability of the elaborated conceptual model to represent CR scheduling problems investigation by other researchers.

Requirements

Brainstorming and general requirements

During the first stage of analysis, concepts of ERP system change process and change planning were elicited. These are presented below in the brainstorm diagram (Fig. 1). This diagram was used as the basis for the next stage, where concepts were structured and supplemented with proper descriptions. All initial concepts are divided in five groups, namely, objectives of CR planning, constraints and factors affecting the planning process, variables influencing the decision-making as well as opportunities for improvements in the change management process.

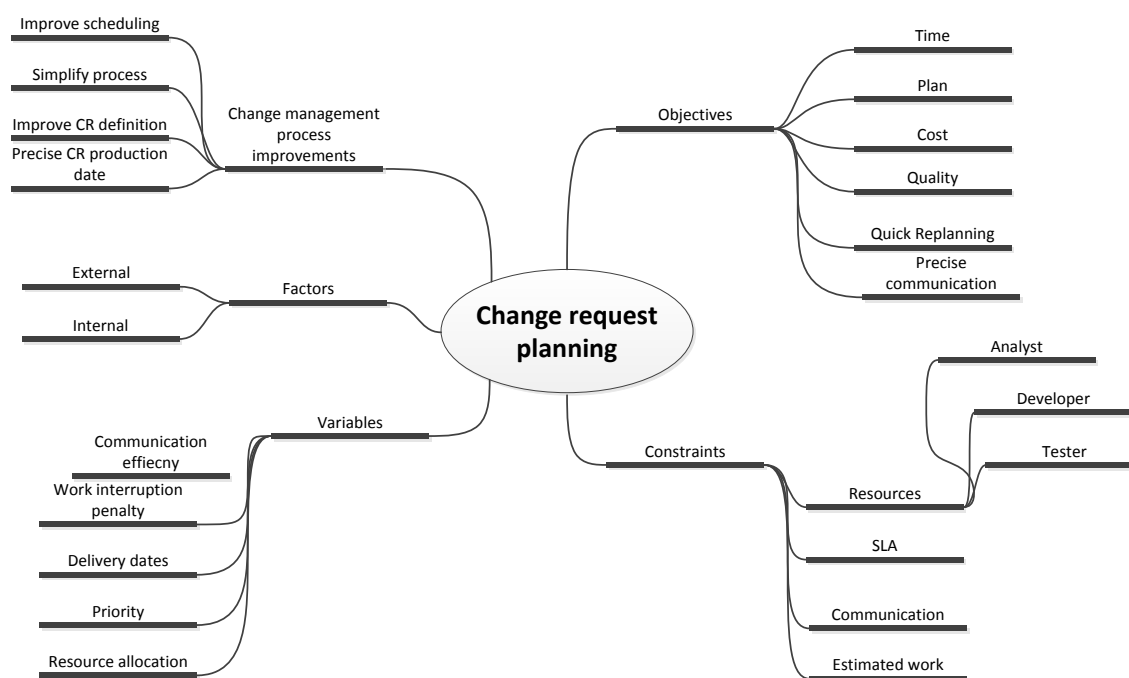


Fig. 1. Brainstorm diagram.

The main requirements towards the change management process and the CR planning algorithm are also identified:

1. The algorithm should support processing of the input CR list considering different parameters (factors) and yield the resource allocation and time plan. This is necessary to make the algorithm relevant to all possible factors which can happen in reality.
2. The algorithm should support re-planning at any point of time. This is necessary in case the priority or any of CR factors changes, new CRs are introduced or existing CR are removed from the list.
3. In re-planning, the algorithm should consider already performed work on the previously planned CRs. As the input CR list or resources can be changed during the time, such requirement is necessary.
4. The algorithm should support functionality to group certain CR from the pool; with intention to process implementation activity for the group of CRs. Regression testing could serve as example here: regression testing is performed for particular ERP software module or unit. If there are several CR related to one ERP software functional unit, the resource savings occur if all them are tested the same iteration.
5. The algorithm should support the following decision objectives: cost saving, plan accuracy with respect to delivery date, shortest implementation time.
6. Planned activities and resource allocation should be represented in a calendar time frame. This is necessary to be able to communicate the planning outcome to change managers or any stakeholders.
7. The algorithm should support human resource pool configuration. It should be possible to define efficiency, cost rate, availability of particular human resource.

Analysis of documents

Tele2 IT organization units have three main documents related to CR planning, namely, (1) the CR list collected from all requestors, (2) the list of available resources and (3) manual mapping of CR towards resources for building the change implementation plan.

The manual planning is driven by two main parameters. Requested production date is a primary parameter and priority is a secondary parameter. Requested production date is neither unique among CR nor mandatory, however it must be strictly observed. Priority is used to order CR with the same requested production date and to order CR without the requested production date. The market-driven requirement engineering processes (Daveport, 1988) have a strong focus on requirement prioritization (Carlshamre, 2002).

A sample CR list is given in Table 1 and a sample resource list is given in Table 2. In addition to production data and priority, each CR is characterized by effort required for change design, development and testing, changes description and functional module affected by the change. Each resource is characterized by its efficiency ranking, hourly rate and area of specialization.

Table 1

Example table of intake change request list

CR	Description	Design (h)	Development (h)	Tests (h)	Priority	Production date	Functional module
1.	Incorrect calculation formula	2	3	8	1	01.01.2012	Finance
2.	Incorrect description on invoice	3	1	12	2	01.02.2012	Invoicing
3.	Legal req. New VAT rate.	12	23	45	1	01.01.2012	Finance
4.	Adaptive change, new XML block in invoices.	12	45	0	1		Invoicing
5.	Low priority GUI fixes. Ordering in ComboBox.	2	2	12	5		GUI

Table 2

Example table of available resource list

ID	Type	Ranking (1..5)	Rate/hour (EUR/h)	Functional unit
1.	developer	1	25	CRM
2.	tester	5	50	Finance, CRM
3.	analyst	3	35	Reporting
4.	tester	2	25	Sales
5.	developer	1	60	Invoicing

Conceptual model

The conceptual model is elaborated according to the requirements gathered. These requirements allow to define the general approach to change implementation planning as shown in Fig. 2. The general approach shows that the planning algorithms receives CR list, resource list and other relevant factors as input data and the list of CR included in a current release and resource allocations are received as the planning outputs. The conceptual model formally defines these input and outputs used by the algorithm.

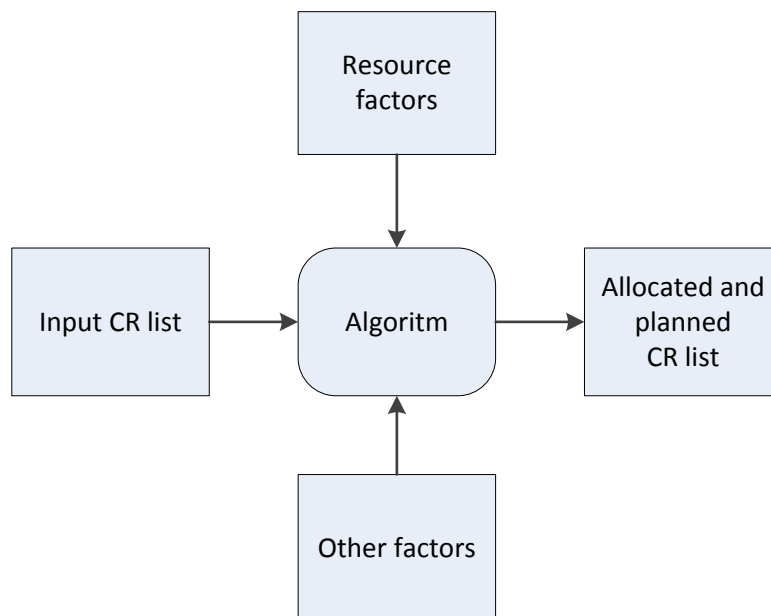


Fig. 2. **The high level description of ERP system change implementation planning.**

During research was identified limitation: each of CR should pass implementation time estimate analyses phase. Potentially, prerequisite for each CR should be provided resource requirements. There should be known how much time is needed from developer/designer/tester in order to fulfil request. The estimated hours might be provided based on average team efficiency (e.g. should not be build based on high performers in the team).

Description of concepts

The main concepts defining the CR planning problem are change requests, change request list, resource, resource list, objective, original requirement, functional software unit, change request plan and resource allocation plan. The change request is a document containing a call for an adjustment of IT system. The main attributes of the change request are:

- Requested production date - calendar parameter. It stands for primary CR processing intake to model. The most production date critical CR should be allocated, if several CRs have the same date, a secondary priority parameter should be used.
- Priority - Subjective unique parameter. It stands for secondary CR processing intake to model. Model should allocate resource to CR consequently based on priority.

- Design (h) - time in hours required for the design of particular CR.
- Development (h) - time in hours required for the development of particular CR.
- Testing (h) - time in hours required for the testing of particular CR.
- Dependency – CR represented by this attribute can have implication or exclusion dependencies with other CRs. There are six types of dependencies (Li, 2007). However, only the implication dependency as most frequently occurring (Carlshamre et al., 2001) is considered in the paper and conceptual model.
- Functional unit - the functional unit of ERP software CR relates to. (E.g. Financial, GUI, Reporting and etc). It is used to group CRs and to plan one regression testing.

The resource is defined as human resource – the individual who makes up workforce of organization. The main attributes of the resource are:

- Type – defines type of the resource, for example, developer, designer, tester.
- Ranking - subjective parameter, which is represents efficiency of specialist, and might reduce estimated time for each activity.
- Rate/hour- actual costs per hour for particular human resource.
- Communication multiplier - if different types of resources are involved in CR implementation some of the development effort is spent on communication. This attribute represents the communication effort, which is only one recourse is involved. This attribute is also used to represent communications with vendors of ERP system.
- Work interruption penalty - resources are often shifted from one task to another to complete new high priority requests. If work on current CR is interrupted, there is a resumption delay when resource returns back this CR. The work interruption penalty attribute accounts for this delay.
- Functional unit - specialization in functional units of ERP system which can be addressed to this resource. Maps to functional unit of CR.
- Availability - planned vacation days, sickness and etc.

Standard regression test estimates for functional unit - Standardized (agreed) time frame which is required to cover functional unit of ERP system with regression tests.

The objective attribute represents the aim of CR planning. The following objectives can be considered: cost saving, time accuracy, the shortest implementation time. Each of these modes can be selected running planning or re-planning.

Associations among concepts

Based on the finding in the research there was created conceptual model, considering found concepts and their attributes. Conceptual model was build using UML class diagram notation (Fig. 3) (Larman, 2005). The concept attributes are not represented on diagram, and are provide in Table 3.

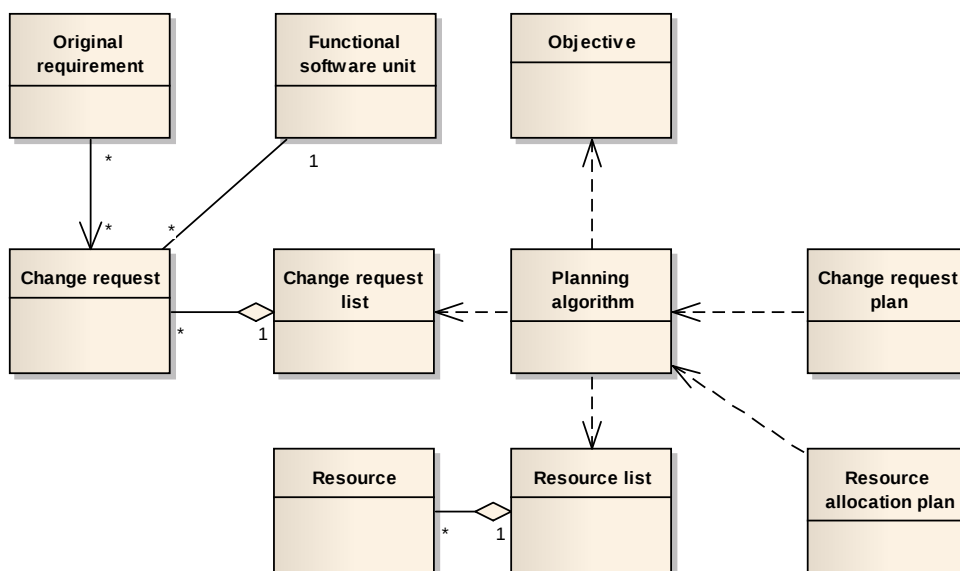


Fig. 3. Conceptual model.

Validation

The conceptual model, concepts and their attributes constructed during research were validated against several existing studies in this field. Four papers (Li et al., 2007; Reboucas et al., 2006; Sauv  l et al., 2006; Carlshamre

et al., 2001) dealing with CR implementation planning were selected for validation. Table 3 marks concepts and attributes from the conceptual model also discussed in the related papers. The validation result shows that the conceptual model is able to represent CR planning factors considered in the related studies and is more comprehensive than existing CR implementation planning models.

Table 3

Concept and their attributes validation

Concept and attributes	Li. C. et al., 2007	Reboucas R. et al., 2006	Sauvé J. et al., 2006	Carlshamre P. et al., 2001
Change Request	x	x	x	X
Requested_production_date	x	x	x	
Priority	x	x	x	x
Required_Design_Time	x	x		
Required_Development_Time	x	x		
Required_Test_Time	x	x		
Dependency on CR	x	x	x	x
Functional_Unit_ID				
Change request List		x		
Resource	x	x	x	x
Ranking	x		x	
Rate_hours	x	x	x	
Communication_multiplier				
Work_interruption_penalty				
Functional_Unit_ID				
ResourceList		x		
Functional units	x		x	
ID	x		x	
Name	x		x	
Required_regression_tests_hours				
Decision directions(cost saving, time saving, precise communication)	x	x	x	x
ID	x	x	x	
Name	x	x	x	
Change Request Plan		x	x	x
ID			x	x
Ordering			x	x
Planned production date			x	
Resource Occupation Plan		x	x	
ID			x	
Change_Request_ID			x	
Hours			x	
Occupation_Date			x	

The cross-evaluation also revealed several important and challenging aspect of CR implementation planning. Importance of CR planning as analysed in this paper is confirmed by Par Carlshamre (Carlshamre et al., 2001): “Release planning is a crucial activity in market-driven software development, because it decided what should be delivered and when. Release planning is matter of prioritization the requirements and selecting a number of top priority requirements depending on the available resources and delivery date.” Related research also

emphasizes importance of the “priority” attribute: “priority of requirements is a major determinant in incremental planning” (Carlshamre et al., 2001).

The further study by Li C. et al. (2007) focus on such attributes as priority and efforts (costs/time): “lists of such variables, including: importance or business value, stakeholder preference, cost of development, requirement quality, development risk” (Li et al., 2007). The study also confirmed on-time-delivery as driver for the model, where the same attribute was selected as major in this paper. However, their model does not consider the decision objective of cost saving, which is one of the main decision-making objectives in release planning.

Re-planning of CR after changes in the CR list is also investigated by Jacques Sauvé et al. (2006): „we identified the main challenges in IT change management as planning and scheduling of changes; high number of emergency changes”. ITIL’s change management process recommendations ranked the first 3 most important change management challenges as being (Sauvé et al., 2006):

1. Scheduling/planning changes (with 47 points out of a maximum 55, or over 85%);
2. High number of emergency changes (43 points or 78%);
3. Request for change scope ill-definition (40 points or 72%).

The first and second points are covered within this paper by providing the re-planning feature.

Discussion and conclusion

According to the web survey (Sauvé et al., 2006), the main challenges in IT change management are 1) planning/scheduling changes, 2) high number of emergency changes and 3) ill-definition or wrong scoping of request for changes. In this paper we addressed the problem of planning and scheduling change requests, taking in account numerous relevant factors.

Analysis of the change implementation process at Tele2 IT division was performed and described in this paper. Main CR planning concepts and their attributes as well as the planning approach were identified.

These are represented in the CR planning conceptual model. The model was validated against the existing CR planning methods reported in literature, and synergies and gaps were identified. The conceptual model serves as the basis for further elaboration of the CR planning algorithm and decision support system to be used by change managers.

The main contribution of this paper is conceptualization of the CR planning problem. Design and implementation of the planning support tool is the subject of the next phase of our research.

The current conceptual model is limited by an assumption that all parameters characterizing CR and resources are constant throughout the planning horizon. However, it is common in different IT companies that some of these parameters might change. For example, a resource changes its original specialization during implementation in order to meet the deadline.

Acknowledgements

The authors would like to thank the Tele2 change/release managers that participated in the research questionnaire.

References

- Bailor and Coreen, 2006. "For CRM, ERP, and SCM, SAP Leads the Way".
- Boehm, B.W., Parikh, G., 1988. Techniques of Program and System Maintenance, QED Information Sciences, Inc. pp. 51.
- Carlshamre, P., 2002. Release Planning in Market-Driven Software Product Development: Provoking an Understanding. Requirements Engineering 7(3). pp. 139–151.
- Carlshamre, P., Sandahl, K., Lindvall, M., Regnell, B., Natt och Dag, J., 2001. An industrial survey of requirements interdependencies in software release planning. In: Proceedings of the 5th IEEE international symposium on requirements engineering, pp. 84–91.
- Carlshamre, P. and Regnell, B., 2000. Requirements Lifecycle Management and Release Planning in Market-Driven Requirements Engineering Processes. International Workshop on the Requirements Engineering Process: Innovative Techniques, Models, and Tools to support the RE Process, 6th-8th of September, Greenwich, UK, the DEXA Conference.
- Chuck, C.H., Law, Charlie, C., Chen, Bruce, J.P., Wu, 2010. Managing the full ERP life-cycle: Considerations of maintenance and support requirements and IT governance practice as integral elements of the formula for successful ERP adoption, Journal Computers in Industry 61 (3). pp. 297-308.
- Daniel, L. and Moody, 2005. Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions, Gerstner Laboratory, Czech Technical University, Prague 13000, Czech Republic.
- Daveport, T., 1988. Putting the enterprise into the enterprise systems, Harvard Business Review 76 (4). pp. 121–131.
- Firesmith, D., 2004. Prioritizing Requirements. Journal of Object Technology 3(8). pp. 35–47.

- Florae, W.A., 1992. Software quality measurement: A framework for counting problems and defects, Technical Report CMUX.EI-92-TR-22, Software Engineering Institute, Carnegie-Mellon University, Pittsburgh, PA Sprott D, 2000.
- Greer, D. and Ruhe, G., 2004. Software release planning: an evolutionary and iterative approach. *Information and Software Technology* 46. pp. 243–253.
- Larman, C., 2005. Applying UML and patterns: an introduction to Object-Oriented Analyses and Design and Iterative Development. - 3rd ed. pp. 134-159.
- Li, C., van den Akker, J.M., Brinkkemper, S., and Diepen, G., 2007. Integrated Requirement Selection and Scheduling for the Release Planning of a Software Product
- Lientz, B. and Swanson, E., 1981, Problems in application software maintenance, *Communications of the ACM* 24 (11). pp. 763–769.
- Reboucas, R., Sauve, J., Moura, A., Bartolini, C., Trastour, D., 2006. “A decision support tool to optimize scheduling of it changes,” in *Proceedings of IFIP/IEEE Network Operations and Management Symposium, Munich, Germany*.
- Sauvé, J. [Rebouças](#), R., Moura, A., Bartolini, C., Boulmakoul, A., Trastour, D., 2006. [Large Scale Management of Distributed Systems Lecture Notes in Computer Science](#) Volume 4269. Business-Driven Decision Support for Change Management: Planning and Scheduling of Changes. pp. 173-184
- Sprott, D, 2000. Componentizing the enterprise application packages. *Communications of the ACM*, 43(4). pp. 63–69.
- Tele2 internal documents., 2012.
- Weerd, I., van de Brinkkemper, S., Nieuwenhuis, R., Versendaal, J.M., Bijlsma, A., 2006. Towards a Reference Framework for Software Product Management. In: Glinz, M., Lutz, R.R. (eds.) 14th IEEE International Requirements Engineering Conference, Minneapolis/St. Paul, Minnesota, pp. 319–322. IEEE Computer Society, Washington.
- Yang and Xin-She, 2010. *Engineering Optimization: An Introduction with Metaheuristic Applications*, Wiley Hoboken, NJ, USA.
- Zia, L., 2006. Optimizing Change Request Scheduling in IT Service Management, IEEE International Conference on Services Computing.

Appendix

Concepts and their attributes

Concept	Attributes
Change Request	ID Number Title Text Requested_production_date Date Priority 0..1 Number Required_Design_Time Date Required_Development_Time Date Required_Test_Time Date Dependency_on_CR Number Functional_Unit_ID Func_Unit
Change Request List	ID Number
Resource	ID Number Type Text Ranking Number Rate_hours Number Communication_multiplier Date Work_interruption_penalty Date Functional_Unit_ID Func_Unit
Resource List	ID Number
Functional units	ID Number Name String Required_regression_tests_hours Number
Decision directions	ID Number Name Dec_directions
Change Request Plan	ID Number Ordering Number Planned production date Date
Resource Occupation Plan	ID Number Change_Request_ID Number Hours Number Occupation_Date Date