Exploring the role of requirements engineering in improving risk management

James Chisan and Daniela Damian
Dept of Computer Science, University of Victoria
PO Box 3055, Victoria, BC V8W 3P6, Canada
{ chisan, danielad} @ cs.uvic.ca

There is limited evidence [1,4,5] which supports claims in literature that Requirements Engineering (RE) can have positive effects on an organization's ability to manage project risk. Such evidence can serve to motivate industrial adoption of requirements engineering techniques, but it can also contribute to our understanding of the role of RE in supporting project management activities. This paper presents an analysis of the causal relationship between RE practice and benefits in risk management in one software organization that revised its RE process and enjoyed improvements in its project management activities. In particular, we present insights from our investigation of the interaction between the RE process and the processes of project planning and SQA. We found that elements of the RE process such as feature decomposition, structure documentation and traceability links improved feature sizing and enabled effective change management, contributing to improved risk management activities. These findings are useful to practitioners in similar software organizations whose product release objectives originate from a remote business unit and who currently suffer from significant feature creep.

1. Research Design

This research was conducted at the Australian Center for Unisys Software (ACUS), a 130 employee software development center within an international multi-site organization with product management and marketing divisions in the US. Prior to improving its RE practice, ACUS faced significant challenges in aligning stakeholder expectations. Projects suffered from significant requirements creep, schedule and cost overruns. ACUS had difficulty in understanding the requested features and providing reasonably accurate development estimates.

Their new RE process included (1) *feature decomposition* into technical requirements, (2) *requirements workshops* (typically attended by 6-10 engineers) that involved (3) *cross functional teams* to analyze these features and (4) definition of *test scenarios* for each technical requirement, and (5) *traceability links* from requirements to rationale, and test scenarios.

Our previous research [2,3] indicated that the revised RE practice at ACUS contributed to improvements in effort estimations, negotiations with customers, controlling requirements creep and ultimately managing project risk. In addition, we also became aware that the relationship between RE practice and these payoffs had to be understood in the light of possible interactions with development other processes. We present here the results of an investigation that analyzed the impact of the revised RE process (REP) on other processes such as project planning and SQA, and the relationship of the particular RE process component that contributed to that impact.

1.1 Data collection

A total of fifteen managers, team-leads and senior engineers from the software engineering, product management and product information departments participated in the study. Their positions as managers or team-leads made these participants uniquely qualified to comment on the subtle process interactions which we sought to understand. Interviews and an online questionnaire were designed (http://vigilant.segal.uvic.ca/acus) to determine the impact (positive or negative) of the revised REP on other major development process areas at ACUS, and to determine which component of the REP most contributed to that impact. The questionnaire asked respondents to rate the impact of the RE process on seven risk-management related processes on a scale from -3 to 3, indicating detrimental or beneficial impact respectively (see Figure 1). Secondly, for each process, respondents could also specify which component of the revised RE process they felt was particularly responsible for the effect on that process.

2. Discussion

The analysis of data indicates not only that the revised RE process had a positive impact on processes of project planning and SQA (see Figure 1), but also allows

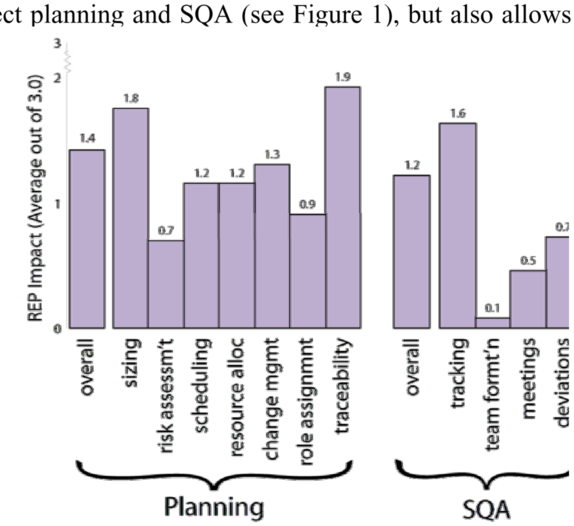

Figure 1. Average impact of REP on planning and SQA processes

Proceedings of the 2005 13th IEEE International Conference on Requirements Engineering (RE'05)
0-7695-2425-7/05 $20.00 © 2005 IEEE

us to describe ways in which the RE process has interacted with processes of change management, feature sizing, and project tracking in contributing to payoffs such as more effective negotiations, more accurate estimates and reduced requirements creep:

Through feature decomposition and structured documentation, the RE process enabled better change management and feature sizing, ultimately leading to more effective negotiations. The decomposition of features in the revised RE process was critical to making sizing estimations, which subsequently enabled ACUS to effectively assess resource requirements. Structured JAD-like workshops held at regular intervals throughout the requirements phase lent credibility and significance to project negotiations. Structured requirements documents aided the change management process which by its very existence threatened to restrict late-changes, motivating stakeholders to align their expectations.

Further, the systematic analysis of requirements led to developers' improved comprehension of features during sizing. That subsequently enabled the project team to make more accurate effort estimations. Those estimations were critical inputs to negotiations so as to align ACUS' development capacity with the market priorities. One team-lead commented that "having requirements done early, it became obvious we could not deliver all of the expected functionality, so we agreed to cut them. Previously we would not have known until it was too late and then everyone would have to go into a mad rush."

It was understood by all stakeholders at the outset of the project that the revised REP would prevent major feature changes after the negotiated requirement set had been committed to. A well-defined change management process motivated ACUS to seek detailed information before commitment, and motivated marketing to prioritize their needs in light of ACUS' finite capabilities.

Through feature decomposition, the REP enabled better sizing and change management, ultimately leading to more accurate estimations. Accurate project estimations were primarily the result of accurate feature effort sizings conducted by engineers who could make extensive use of technical requirements (as derived from features received from marketing). Interestingly ACUS achieved this improvement without the use of elaborate estimation methods, such as function-points. By bringing clarity to its requirements in the revised RE process, ACUS was able to leverage its engineers' extensive technical knowledge and experience of the product to produce far more accurate project estimations. Again, the change management process prevented unexpected changes and helped to ensure original estimations stayed on target.

Individual technical requirements, estimated by the engineers assigned to analyze and implement them, were aggregated to construct feature estimations and guide project planning and project estimations. The revised REP facilitated a thorough understanding of requirements such that "time spent during RE to get finer detail made it possible to have better estimates earlier on."

Once schedules and resource allocations were established, careful control of change requests prevented project corrections while ensuring the currency of the original estimations. As one manager said, estimations remained on-target via change management because "the process reduced the number of changes sneaking into the product."

Providing enhanced feature understanding enabled better change management and project tracking, ultimately leading to reduced requirements creep. Change management was instrumental in preventing requirements creep. The success of ACUS' software change control board, an integral part of this achievement, confirms the board's role as a means to control software change. Merely by virtue of implementing a formal change management process, engineers were dissuaded from making discretionary changes. Traceability established within the requirement specification which were used by project tracking to monitor resources, helped to prevent creep from significantly affecting progress.

The structured nature of requirements artefacts, helped enable change management, giving project managers control of requirements creep. Requirements churn that had been so common in past projects was controlled by relying on a rigorous requirements change process that limited all but the most critical changes. One engineer reported that the approval process itself was significant: "[it] had a big impact: it made people analyze and think twice about the changes they were considering."

When changes were necessary, change requests were considered in the context of project progress. Respondents indicated that improvements in project tracking that had occurred because of the revised REP enabled "identification of schedule risks" making it "easier to forecast resource crunches." Managers said they could effectively assess change requests and that the management process provided "firm control and visibility."

References

1. Brooks, F. No silver bullet: Essence and accidents of software engineering, *Computer* 20(4), April, pp. 10-19. 1987.
2. Damian, D., Zowghi, D. Vaidyanathasamy, L. and Pal, Y. An Industrial Case Study of Immediate Benefits of Requirements Engineering Process Improvement at the Australian Center for Unisys Software. *International Journal of Empirical Software Engineering* 9 (1-2), 45-75, 2004
3. Damian, D., Chisan, J., Vaidyanathasamy, L. and Pal, Y. Bridging the gap between theory and evidence: the impact of requirements engineering process on downstream development. *International Journal of Empirical Software Engineering*, in print, 2004.
4. Fosberg, K. and Mooz, H. System Engineering Overview, in Thayer, R. and Dorfman, M. (Eds.): *Software Requirements Engineering*, pp. 44-72, 2000
5. Robertson, S. and Robertson, J. Mastering the Requirements Process, Addison-Wesley, 1999

0-7695-2425-7/05 $20.00 © 2005 **IEEE**