Guiding requirements scoping using ROI: towards agility, openness and waste reduction

Krzysztof Wnuk
Department of Computer Science
Lund University
Lund, Sweden
Krzysztof.Wnuk@cs.lth.se

David Callele
Department of Computer Science
University of Saskatchewan
Saskatchewan, Canada
callele@cs.usask.ca

Björn Regnell
Department of Computer Science
Lund University
Lund, Sweden
Bjorn.Regnell@cs.lth.se

***Abstract*—We present a model for supporting scoping decisions that is based on an analysis of the ROI for a given feature. Employing a ROI threshold value for making scoping decisions, the utility of the model was investigated using data from a single large project and identified a group of outlying features responsible for a disproportionate wasted investment. These initial results are promising and indicate that further investigation and validation efforts are warranted.**

Keywords: Requirements visualization, process evaluation, scope, efficiency, cost estimation, economics

I. Introduction

In rapidly changing markets, the scope of a project must adapt to competitive pressures and remain responsive to changing conditions - some degree of flexibility is necessary. Responding in a timely manner and making appropriate scope decisions is a vital part of developing software systems that meet stakeholders' needs and expectations [1]. However, the prioritization techniques reported in the literature (*e.g.* [2]) typically model the scoping decisions as discrete temporal events that are binary (keep or cancel) and final. This approach does not capture the reality of dynamic development methodologies – other options (such as investigate further, reschedule, decompose or refine) are not supported and the resulting model(s) may not adequately address the needs of highly dynamic markets. The model presented here represents the decision-making criteria as temporal functions, not absolute values, which facilitates their use even in dynamic situations with uncertain scoping decisions.

II. An Economic Basis for Scoping Decisions

In this model, the estimated market value $V(t)$ of a feature is expressed as follows:

$$\int_a^b V(t)\,dt \qquad (1)$$

where $t = a$ is when the feature begins to have market value and $t = b$ is when the feature ceases to have market value.

The development costs $C(t)$ is the summation of the individual cost factors over time. The costs for the feature accrue at different rates, at different phases of the project, and are expressed as follows:

$$\int_a^b C(t)\,dt \quad \textit{where} \quad C(t) = \sum_{k=1}^{n} \int_a^b C_k(t)\,dt \qquad (2)$$

where a is the beginning of expeditures on the feature, b is the end of expenditures on the feature (either the feature is implemented or the feature was removed from the scope) and k is the number of cost factors included in the estimate.

The cost factors supported in the model are as follows; (C_1) is a direct cost, all other identified costs are indirect costs.

- C_1 *Requirements development* The cost of product and feature definition and associated requirements engineering efforts.
- C_2 *Production personnel* Canceling a feature can lead to inefficiencies such as the overhead associated with changing focus or, in worst case, idle capacity.
- C_3 *Other production resources* Canceled features can also incur costs such as those associated with capital equipment, carrying costs, *etc.*
- C_4 *Project Management* Significant costs are incurred when project plans are changed. Planning for, capturing, and implementing the changes can result in the loss of management focus on critical tasks or areas.
- C_5 *Estimated Cost of Development* Summary estimate for preparing the feature for production.
- C_6 *Estimated Cost of Production* Summary estimate for production cost of feature, aggregated over all units.

Given the value of a feature and the associated costs, we can calculate the Return On Investment (ROI). We propose that features should be kept within project scope until an ROI threshold level is reached – once the feature ROI drops below the ROI threshold, the feature is canceled.

$$\frac{\int_a^b V(t)\,dt}{\int_a^b C(t)\,dt} \leq \delta \qquad (3)$$

The value of the ROI threshold, δ is value/cost only in the simplest models – δ generally includes other business factors such as probability of market success based on past performance and analyses of projected market conditions at the time of product release. The flexibility inherent in the development process can be manipulated by adjustments to the value of δ – increasing the value of δ increases the probability of feature cancellation.

Feature Impact: The overall impact of a set of features within a development cycle can be calculated using the following equation:

$$\sum_{k=1}^{K} \left(\frac{\int_a^b V_k(t)\,dt}{\int_a^b C_k(t)\,dt} - \delta_k \right) \tag{4}$$

where K is the number of features analyzed, a is the point in the project when the feature was added and b when the feature was canceled (the values for a and b are not necessarily the same for different k). If the feature impact is negative for the entire set of features in a product cycle then the project manager can not expect that the product cycle will be profitable. A negative value for feature impact is generally acceptable only if there are enough infrastructure features (whose true value is deferred) to dominate revenue generating features.

III. Investigating Model Utility

The utility of this model was investigated using data from a large industrial project with 223 features, 120 of which were canceled during the analysis period. The data set contains data for the time from feature inclusion to cancellation and estimated effort data for each feature. However, the data set does not contain all of the cost function and value function data necessary for the model; the missing data was simulated according to the following characteristics.

- The value of effort exerted on requirements definition (C_1) was based on actual effort data records from the project. The effort was prorated on a per-milestone basis (from Milestone 1 (MS1) to Milestone 4 (MS4)). Effort before MS1 (feature definition phase) was depreciated by 50%, between MS1 and MS2 was at full cost (requirements definition should be complete by MS2), and after MS2 was inflated by 50% (to penalize late requirements definition efforts).
- The other costs C_2, C_3, C_4 were assumed to grow linearly at a cumulative rate of 15% per milestone (*e.g.* 15%, 30%, 45%) and were assigned a base cost equal to C_1.
- The market value was synthesized from a random distribution seeded by the cost data. Market value was calculated as a weight ((between 0 and 10 units)*(cumulative cost)) that was then depreciated at the rate of 20% per milestone until the feature was canceled.

In Figure 1, we see a graph of the resulting ROI values for the individual features that were canceled. The absolute magnitude of the peak ROI is a function of our simulated data, and in this case we can see that the threshold ROI value for the canceled features was approximately 4. The dispersion along the y-axis is a function of the effort invested in the feature and the time delay associated with the eventual decision to cancel the feature. The data point ID is arbitrary.

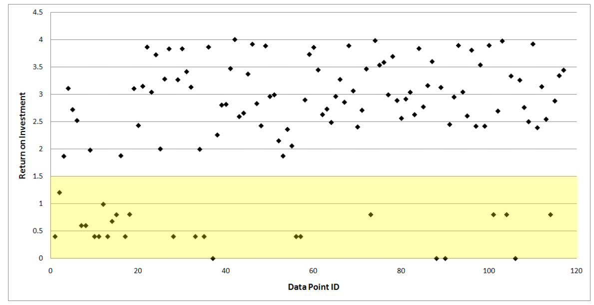

Figure 1: ROI per canceled feature, the yellow region denotes the outlier area of interest.

Upon further inspection of Figure 1, we note that the ROI data is clustered in two regions. The upper region represents those features that had greater potential to be kept in the project, based on their projected ROI. If any of these features are canceled, the investment in these features may be lost – but there was a sound basis for making the investment. The lower region, below the ROI threshold value of 1.5, contains the outliers and they represent approximately 21% of the canceled features. However, the effort spent on the features in this region corresponds to 77% of the total investment on features that were cancelled. The data indicates that the wasted investments in these features are significantly larger, on a per feature basis, than the features in the upper region. It appears that this model and technique may facilitate identifying these outlier features, enabling project management to cancel the features earlier and reducing wasted effort.

IV. Conclusions

In this work, we present a new model for supporting scoping decisions. Based on an analysis of the ROI for a given feature, and employing a ROI threshold value for making scoping decisions, the utility of the model was investigated using data from a single large project. Initial results are promising and indicate that further investigation and validation efforts are warranted.

Acknowledgments

This work is supported by VINNOVA (Swedish Agency for Innovation Systems) within the UPITER project.

References

[1] J. Karlsson and K. Ryan. A cost-value approach for prioritizing requirements. *IEEE Software*, 14(5):67–74, 1997.

[2] J. Karlsson, C. Wohlin, and B. Regnell. An evaluation of methods for prioritizing software requirements. *Information and Software Technology*, 39(14-15):939–947, 1997.