IDEA GROUP PUBLISHING

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITP4586

Situational and Task Characteristics Systematically Associated With Accuracy of Software Development Effort Estimates

Magne Jørgensen
Simula Research Laboratory, magne.jorgensen@simula.no, Telephone: +47 924 333 55, Fax: +47 67 82 82 01

Kjetil Moløkken, Simula Research Laboratory

ABSTRACT

Estimation skill is only one, out of many, factors that potentially impacts the accuracy of software development effort estimates. In this paper we examine how the size of the development task, the contract type (fixed-price versus per hour payment), the task priorities (time-of-delivery, quality, or cost), and the difference between estimating own and other peoples work, impact estimation accuracy. We found that an understanding of these factors could be important to explain the variance of estimation accuracy and, consequently, important when deciding on estimation improvement actions based on estimation accuracy measurement.

1 INTRODUCTION

Organizations developing software have, in general, a bad reputation for effort estimation. According to a survey carried out in 1998 by Standish Group[i] only 26% of the software projects completed on time, on budget and with the originally specified functionality. The characteristics of software projects, such as "one of a kind" activities, dynamic environments, changing requirements and carried out by humans, mean that we cannot expect zero effort overruns. There is however little doubt about that the improvement potential regarding estimation accuracy is large.

Most software effort estimation research studies seem to have a strong focus on the factors impacting the actual use of software development effort. Those factors are essential when building estimation models or providing the estimators with relevant information, but may not be sufficient for a proper understanding of the factors impacting the estimation *accuracy*. Standish Group, for example, reports that the main source of estimation accuracy improvement from 1994 to 1998 was the shift towards smaller projects, i.e., the estimation tasks were on average easier in 1998 compared with 1994.

The study reported in this paper is a follow-up study on our previous study of project experience reports at Ericsson Design Center in Norway (Jørgensen, Løvstad et al. 2002). In that study we found that a meaningful interpretation of the measured effort estimation accuracy required information about the project priority and properties of the requirement specification. For example, we found one project with a low priority on product quality (the software was meant to be a "demo") and a strong focus on not exceeding the cost budget. That project had very high effort estimation accuracy. However, it would be incorrect to attribute this high effort estimation accuracy only to good estimation skills. The experience report of that project indicated that the quality and completeness of the functionality had been adjusted to fit the available time and effort, i.e., that the requirement specification had been quite flexible. Similarly, there are several studies reporting different types of impact from the estimate on the project behavior, e.g., (Abdel-Hamid and Madnik 1986; Abdel-Hamid 1990). One type of project behavior impact from the estimate is the so-called "self-fulfilling prophecy" effect of software effort estimates, e.g., that an over-optimistic initial estimate and a high focus on estimation accuracy lead to actions that make that estimate more realistic.

To increase the confidence in the reported results and further extend our understanding of factors impacting the estimation accuracy we conducted a study of 60 software development tasks in a Web-development company. The hypotheses, which were based on our experience from earlier studies, tested were:

- H-1: Small tasks are typically over-estimated and large tasks under-estimated.
- H-2: Fixed-price tasks are typically over-estimated and tasks paid per hour typically under-estimated.
- H-3: Tasks where the customer prioritize quality or time-of-delivery have less accurate effort estimates compared with those with priority on cost.
- H-4: Tasks were own work is estimated are more accurately estimated than those where other developers' work is estimated.

The motivation for each individual hypothesis is described in Section 3. A study with the same goal, i.e., to increase our knowledge about factors impacting estimation accuracy, is described in (Gray, MacDonnell et al. 1999). That study found that over-estimation was connected with changes on small modules and development of screens, while under-estimation was connected with changes on large modules and development of reports. A potential explanation of these observations is that easy tasks (small modules and screens) typically are over-estimated, while complex tasks typically are under-estimated, i.e., an explanation consistent with our hypothesis H-1.

2 DESIGN OF STUDY

The company that participated in our study is the Norwegian branch of a large international web-development company. The role of the company is that of a contractor (McDonald and Welland 2001), producing web-solutions for its customers. Over a period of approx. 10 months we collected information about 60 software development tasks, i.e., most of the small and medium sized development tasks conducted by the company in that period. The median size of a task was 45 work-hours.

All tasks were estimated without the support of estimation models or databases of previous projects, i.e. "expert estimates". The practical difference between expert and model-based effort estimates is, in our opinion, much smaller then the "theoretical" difference. While expert estimates are based on non-explicit and non-recoverable reasoning processes, i.e., "intuition", the steps leading to a model based effort estimates are "in theory" explicit and recoverable. However, most estimation processes applied in practice have both intuitive and explicit (model based) reasoning elements (Blattberg and Hoch 1990). In fact, most formal software development estimation models requires expert estimates of important input parameters (Pengelly 1995), i.e., they require non-explicit and non-recoverable reasoning. The expert estimation-based accuracy results reported in this paper may, therefore, be valid for more model-based effort estimation.

Information collected immediately before the design and implementation of the task started, for each task:

• Name of the estimator
\• Short description of the task (max 10 lines)
• Type of contract (fixed price, per hour). 43% of the tasks were fixed-price, 42% per hour, and 15% of unknown contract type.
• Customer priority (cost, time-of-delivery, quality). The customer prioritized in 22% of the tasks the cost (given an acceptable level of quality), in 48% the quality, and in 30% the time-of-delivery (given an acceptable level of quality).
• Proportion of the task planned to be completed by the estimator (zero, between 1% and 50%, more than 50%). 22% of the tasks were planned to be completed more than 50% by the estimator, 30% between 1% and 50%, and in 48% of the tasks the estimator was not supposed to participate in the development, at all.
• Estimated effort in work-hours. The estimated effort should be the "most likely" use of effort, not the effort accepted by the customer in the contract (the "price-to-win" effort). From the answers in the "reasons for high or low estimation accuracy"-field (see field description below) it is clear that there were tasks were the estimators were influenced by "price-to-win" effort when providing the effort estimates, i.e., there were situations with a poor separation of most likely and price-to-win effort estimates. The same lack of separation is reported in (Jørgensen and Sjøberg 2001) and (Jørgensen, Løvstad et al. 2002), i.e., this problem may be typical for software organizations.

Information collected immediately after the task was completed, to avoid hindsight bias (Stahlberg, Eller et al. 1995):
• Actual effort in work-hours.
• Unexpected problems during the task execution (free text). 18% of the tasks experienced at least one major unexpected problem. This is less than reported, on maintenance tasks of similar size, in (Jørgensen 1995). That study reported a proportion of 30% tasks with major unexpected problems
• Reasons for high or low estimation accuracy (free text). Almost all estimators wrote 20-200 words describing estimation accuracy causes. We use this information in the discussion of the data analysis presented in the result section.

3 RESULTS

3.1 Size of Task vs Estimation Accuracy

H-1 hypothesizes that small tasks are typically over-estimated and large tasks under-estimated. An argument for the size impact is the so-called "regression-toward-the-mean" effect (Jørgensen, Indahl et al. 2002). This effect implies that estimates tend to move closer to the mean[ii] effort with increasing uncertainty. This means that tasks smaller than the average task may be over-estimated and tasks larger than the average task will be under-estimated. In the extreme case, where the estimator knows very little about the effort usage of the new task, a rational estimation approach is to estimate effort usage close to the effort of the average task. The regression-toward-the-mean effect was first described by Sir Francis Galton (Galton 1997). There are studies (Kahneman and Tversky 1973; Nisbett and Ross 1980) demonstrating that the regression-toward-the-mean effect in real life situations can be large and that people tend to overlook it.

The median size of the tasks in our data set was 45 work-hours. We use this value as an indication on the effort usage on the average task in this analysis. To test H-1 we divided the tasks into two categories: SMALL (< 45 work-hours) and LARGE (>= 45 work-hours). A Kruskal-Wallis test on the difference in median relative estimation deviation, defined as *MRE0 = (actual effort – estimated effort)/Actual effort*, shows a significant difference (p=0.02). The median MRE0 for the small tasks was 0%, i.e., under-estimation was just as frequent as over-estimation, while the median MRE0 for the large tasks was 21%, i.e., the typical large task was under-estimated with 21%. The general tendency towards under-estimation (median MRE0 for all projects was 8% under-estimation[iii]) means that, although the hypothesized estimation deviation tendency is correct, our hypothesis is only partly supported. A better formulated hypothesis may be that the likelihood of underestimated tasks is much higher for large tasks compared with small tasks.

3.2 Contract Type vs Estimation Accuracy

H-2 hypothesizes that fixed-price tasks are typically over-estimated and tasks paid per hour typically under-estimated. An argument for H-2 is that, when a company is paid per hour for a task, this induces less focus on not exceeding the estimate compared with the fixed-price situation. In the fixed-price situation the company loses money when exceeding the estimate, while the opposite may be the case in the payment per hour situation.

We tested H-2 applying the Kruskal-Wallis test on the median estimation accuracy (MRE0) of the tasks of the two contract types. The results were in the opposite direction of what hypothesized in H-2. The fixed-price tasks were more, not less, under-estimated (median under-estimation of 18%) than the tasks paid per hour (median under-estimation of 9%)[iv]. The hypothesis H-2 is therefore not supported.

An examination of the descriptions of unexpected problems and reasons for high or low estimation accuracy suggests that fixed price task estimates were frequently impacted by how much the customer was willing to pay. In particular, there were several examples of fixed-price tasks were the customer negotiations had pressed the estimate down very much. The consequence of this pressure was inaccurate, much too low, estimates. The relationship between contract-type and estimation accuracy is therefore more complex than we hypothesized. On one hand, there is a stronger incitement for not exceeding the estimate in the fixed-price situation. On the other hand, the customers' negotiation impact towards lower estimates may also be larger. We observed similar effects in the study reported in (Jørgensen and Sjøberg 2001).

3.3 Priority vs Estimation Accuracy

H-3 hypothesizes that tasks where the customer prioritizes quality or time-of-delivery have less accurate effort estimates compared with tasks where the customer prioritizes cost precision. The main argument for the hypothesis is that the developers try to optimize their behavior in accordance with the task priority, e.g., if time-of-delivery is priority one, there is less focus on actions to reduce the probability of exceeding the cost budget (Weinberg and Schulman 1974).

We applied the Kruskal-Wallis test on the median absolute relative estimation deviation (*MRE = |actual effort – estimated effort| / actual effort*). The median MRE was 11% on tasks with a priority on quality, 30% on tasks with a priority on time-of-delivery, and 18% on tasks with a priority on cost. The difference was significant (p=0.09). The difference was particularly large between the tasks with a priority on time-of-delivery and the other priorities, i.e., time-of-delivery seems to be an important indicator for high estimation deviations. The estimation accuracy (MRE) of tasks with a priority on quality was lower than those with a priority on cost. A closer examination of the described unexpected problems and reasons for high or low estimation accuracy suggests that the customers with a priority on cost were "more demanding" than the others, i.e., they negotiated lower fixed-price estimates or required more functionality than the developers had assumed when they estimated the task. An analysis of the MRE0, shows that a priority on cost had an impact on the level of under- and over-estimation. The median estimation deviation of the tasks with a priority on cost was 3% over-estimation, with a priority on quality the MRE0 was 11% under-estimation, and with a priority on time-of-delivery the MRE0 was 25% under-estimation. Clearly, the priorities of the task impact the estimation accuracy.

3.4 Estimation of Own Work vs Estimation Accuracy

H-4 hypothesizes that estimating own work leads to more accurate estimates than estimating other developers work. The main argument for this hypothesis is that the estimators estimating own work is more likely to use the estimate as a goal and that it is more difficult to predict the productivity on other people. This argument is supported by the software study reported in (Lederer and Prasad 1998). However, results reported other domains than software, e.g., in (Buehler, Griffin et al. 1994), suggest that estimating own work typically lead to more over-optimism compared with the estimation of other peoples work. A potential reasons for this over-optimism is the "I am above average"-bias (Klein and Kunda 1994), i.e., that far more than 50% of people believe they are above average skilled in work related tasks. The direction of the difference, stated in our hypothesis H-4, is based on the belief that the software study described in (Lederer and Prasad 1998) is the more relevant for our purpose than the studies reporting results from different estimation domains.

A Kruskal-Wallis test of the median MRE of the categories "Estimation of other developers work", "Less than 50% of the work conducted by one-

self", and "More than 50% of the work conducted by one-self" resulted in a weakly significant (p=0.12) difference. The lowest estimation accuracy (median MRE of 31%) was, as hypothesized, achieved for those tasks where other software developers' work was estimated. The two other categories had median MRE of 16% ("Less than 50% of the work conducted by one-self") and 20% ("More than 50% conducted by one-self"), i.e., no large estimation accuracy difference between these two categories. The hypothesis H-4 is supported.

4 CONCLUSION

This paper reports that:

- Large software development tasks were typically under-estimated, while small tasks were just as frequently over-estimated.
- There was a complex relationship between contract type and estimation accuracy. Fixed price tasks experienced more frequently that a customer negotiation induced estimation pressure leading to less realism of the effort estimates. On the other hand, fixed price estimates led in some situations to more awareness of the importance of not exceeding the estimate, i.e., to better estimation accuracy.
- Tasks with a priority on time-of-delivery had lower estimation accuracy than those with a focus on quality or cost.
- Estimating own work led to more accurate estimates compared to estimating other peoples work.

There are several applications of these results. In order to understand the reasons for high or low estimation accuracy, and not automatically attribute it to good or poor estimation skills, it is important to know and apply information about the factors reported in this paper. Another application of the reported results is an increased awareness of the situations leading to high or low estimation accuracy. This is, for example, important information when assessing the uncertainty of an estimate or when deciding on actions to improve the estimation accuracy.

NOTES:

i) http://www.standishgroup.com/sample_research/chaos1998.pdf. Describes results from a survey of US companies.

ii) An estimator's interpretation of "mean effort" may depend on the size of tasks the estimator is used to estimate and on the "reference class" of task, i.e., the tasks that the estimator believes are relevant to compare with when estimating the new task.

iii) An earlier study in the same company (Moløkken 2002) found that the average under-estimated effort was 15%, as opposed to our 8%. That projects analyzed in that study, however, were on average larger than the tasks included in our study. This supports the finding that the size of the projects is an important factor when explaining differences in estimation accuracy.

iv) Amongst the tasks were the estimator did not know the contract type, the median MRE0 was as low as -6%, i.e., the median task was over-estimated.

REFERENCES

Abdel-Hamid, T. 1990. Investigating the cost/schedule trade-off in software development. *IEEE Software* 7(1): 97-105.

Abdel-Hamid, T. K. and S. E. Madnik 1986. Impact of schedule estimation on software project behavior. *IEEE Software* 3(4): 70-75.

Blattberg, R. C. and S. J. Hoch 1990. Database models and managerial intuition: 50% model + 50% manager. *Management Science* 36: 887-899.

Buehler, R., D. Griffin and M. Ross 1994. Exploring the "Planning fallacy": Why people underestimate their task completion times. *Journal of Personality and Social Psychology* 67(3): 366-381.

Galton, F. 1997. *Natural Inheritance*. New Mexico, Genetics Heritage Press (originally published in 1889 by Macmillan and Company).

Gray, A., S. MacDonnell and M. Shepperd 1999. Factors systematically associated with errors in subjective estimates of software development effort: the stability of expert judgment. *Sixth International Software Metrics Symposium*, IEEE Comput. Soc, Los Alamitos, CA, USA: 216-227.

Jørgensen, M. 1995. An empirical study of software maintenance tasks. *Journal of Software Maintenance* 7: 27-48.

Jørgensen, M., U. Indahl and D. Sjøberg 2002. Software effort estimation and regression toward the mean. *Accepted for publication in Journal of Systems and Software*.

Jørgensen, M., N. Løvstad and M. L 2002. Combining quantitative software development cost estimation precision data with qualitative data from project experience reports at Ericsson Design Center in Norway. *Empirical Assessments of Software Engineering (EASE)*, Keele, UK.

Jørgensen, M. and D. I. K. Sjøberg 2001. Software process improvement and human judgement heuristics. *Scandinavian Journal of Information Systems* 13: 99-121.

Kahneman, D. and A. Tversky 1973. On the psychology of prediction. *Psychological Review* 80(4): 237-251.

Klein, W. M. and Z. Kunda 1994. Exaggerated self-assessments and the preference for controllable risks. *Organizational behavior and human decision processes.* 59(3): 410-427.

Lederer, A. L. and J. Prasad 1998. A causal model for software cost estimating error. *IEEE Transactions on Software Engineering* 24(2): 137-148.

McDonald, A. and R. Welland 2001. Web Engineering in Practice. *Proceedings of the Fourth WWW10 Workshop on Web Engineering*: 21-30.

Moløkken, K. 2002. Expert estimation of Web-development effort: Individual biases and group processes (Master Thesis). *Department of Informatics*, University of Oslo.

Nisbett, R. E. and L. Ross 1980. *Human inference: Strategies and shortcomings of social judgment*, Englewood Cliffs, NJ: Prentice-Hall.

Pengelly, A. 1995. Performance of effort estimating techniques in current development environments. *Software Engineering Journal* 10(5): 162-170.

Stahlberg, D., F. Eller, A. Maass and D. Frey 1995. We knew it all along: Hindsight bias in groups. *Organizational Behaviour and Human Decision Processes* 63(1): 46-58.

Weinberg, G. M. and E. L. Schulman 1974. Goals and performance in computer programming. *Human Factors* 16(1): 70 - 77.