A Survey of Software Project Managers on Software Process Change

Yuepu Guo
Department of Information Systems
University of Maryland Baltimore County
Baltimore, Maryland 21250, USA
+1-410-455-8842
yuepu.guo@umbc.edu

Carolyn Seaman
Department of Information Systems
University of Maryland Baltimore County
Baltimore, Maryland 21250, USA
+1-410-455-3937
cseaman@umbc.edu

ABSTRACT
Software project managers play an important role in selecting their software development process. In this study we conducted a survey of software project managers about software process change. The result of the survey revealed several factors affecting this type of decision making. It also revealed critical issues in software development projects. In particular, the findings point to the importance of a piloting strategy in technology transfer, as well as the importance of highlighting cost, quality, and schedule information in reporting evidence of a new technique's effectiveness. We expect that the findings of this study could facilitate research on technology transfer and adoption.

Categories and Subject Descriptors
D.2.9 [**Software Engineering**]: Management – *life cycle, Productivity, Software Process Models*

General Terms
Management, Human Factors

Keywords
Software development process, change, decision making, technology transfer

1. INTRODUCTION
The goal of a software development process is to apply repeatable, predictable procedures, thus improving the productivity of projects and the quality of software. There exist various software process models, such as the traditional waterfall approach and the newer collection of agile processes, each of which is applicable to different environments with its own advantages as well as limitations. Selecting an effective overall development process is critical to the success of software development projects, but it is not always easy. Further, changes in the development environment often necessitate changes in the development process, even a complete change of development paradigm. Competitive markets, pressure for increasingly shorter delivery times, tight budgets, changing technologies, a volatile workforce – all of these are motivations for decision makers to reexamine the software development processes that have served them well in the past.

We have begun a research initiative with an industrial partner to address the difficulties that decision makers (in particular software managers) face when contemplating major process changes. Our partner is a large multinational company providing office equipment and document management solutions to organizations of all types. It is among the top 200 companies according to Fortune 500 ranking. More than 50,000 employees are working at the company around the world. It is currently experiencing a gradual evolution in its software development projects from a traditional waterfall approach to a blend of several newer iterative and agile processes. This transition is being carried out largely organically, through the choices of individual software managers, rather than being dictated from upper management. Therefore, our joint research project has as a goal to collect a convincing and illuminating body of evidence that will assist software project managers in the company in making informed choices about software development approaches for their projects. Such evidence should go beyond a simple categorization of what approaches are in general "better", but should impart an understanding of what is "better" with respect to different criteria, in different contexts, and in view of various goals.

In the context of this objective, we have also studied the motivations of software project managers in their decisions about software process. The survey study described in this paper was intended to elicit the views of experienced software managers in our partner company about what constitutes convincing evidence, and what information would be most useful to them in making their software development process decisions. Our motivation for asking these questions was to inform our own joint research project, but we believe the results also offer some insight about what decision makers need from the software engineering research community.

2. RELATED WORK
In this section, we summarize several lines of previous research that have addressed how people decide to change the way they work. We pay particular attention to reports based on experience with software development, but we turn first to more classical theories of acceptance behavior. The most influential theory depicting users' attitude and behavior toward acceptance of a new technology is the Technology Acceptance Model (TAM) developed by Davis [1] and Bagozzi [2]. In TAM two major factors that affect user acceptance of a new technology were

ESEM'08, October 9–10, 2008, Kaiserslautern, Germany.

identified. One is *Perceived Usefulness*, and the other is *Perceived Ease of Use*. Since TAM was proposed, many researchers have been engaged in validating and extending TAM from different perspectives. In 2003 Venkatesh et al. [3] compared TAM and other well-known models based on the TAM, and then proposed a new model called Unified Theory of Acceptance and Use of Technology (UTAUT). Not only did UTAUT summarize and redefine the direct determinants of intention of usage, but also came up with four moderating factors. TAM and UTAUT have become the foundation of research in areas such as technology adoption, technology transfer, etc.

In [4] Kaindl et al. studied the causes of the difficulty in transferring research results about software requirements engineering to industrial practice. They also presented the incentives to implement requirements engineering in the IT industry. The research is based on the results of two panel discussions held at two requirement engineering conferences. In this study, the obstacles to technology transfer were classified as those related to the "technology producer" and those related to the "technology consumer". Technology producers are universities or other research institutes, while technology consumers are industrial companies and organizations. For the producer, obstacles include a lack of environmental infrastructure for hosting large long term projects, a lack of tools to support proposed new languages and methods, etc. For the consumer, the difficulty of technology transfer is due to frequently changing requirements, budget and time limits for trying new methods under a competitive business environment, concerns about lack of technical support [4], etc.

Rainer and Hall investigated the factors affecting Software Process Improvement (SPI) in [5]. They summarized the factors identified through a survey conducted by other researchers and themselves in their previous work [6]. These factors were then compared with those identified through a group interview. The purpose of the comparison is to identify the most relevant factors as well as to exclude factors that emerged with lower frequency or only through one research strategy. Thus they could increase the validity and reliability of the research result. In this study they identified the most relevant factors by calculating the frequency of the key words used in the group interviews. They also analyzed the words in context to ensure co-occurrence of words did indicate logical connections. Therefore this study is a good example of the combination of methodologies that could be applied in related research. Rainer et al. prioritized factors affecting developers to "buy in" to SPI in another study [7]. The most critical finding of this study is the contradiction between claiming that developers want to get empirical evidence showing the advantages of SPI and what the developers actually accept as evidence. This finding implies the importance of "local opinions" over "local empirical evidence" from the developers' perspective. Although they focused on eliciting developers' opinions, which is different from the subjects we are interested in, this work revealed the importance of the local context for implementation of SPI.

Based on a literature review of requirements engineering and software engineering research, Kauppinen et al. [8] pointed out that researchers in the software engineering domain have paid more attention to the effect of human factors on the implementation of requirements engineering processes than researchers in the requirements engineering domain. In this work they proposed a new model of the factors affecting requirements engineering process in organizations. The model defines three dimensions – characteristics of requirements engineering, human and organizational infrastructure. The factors are grouped into the dimensions. Moreover, they studied the relationships among the dimensions to reveal the root causes of the complexity of the organization-wide implementation of a requirements engineering process. From a research perspective, this model introduced a new dimension–organizational infrastructure such as training and support–to classify the factors, which is different from the basic categories defined in TAM [1] and UTAUT [3].

Pfleeger conducted a comprehensive literature review regarding technology transfer in [9]. The review involved general technology transfer models such as Rogers' theory of innovation diffusion [10] as well as specific case studies such as software engineering technology infusion within NASA [11]. Based on previous work by researchers in this area, Pfleeger argued that software engineering technology transfer requires more than just a new idea and some evidence showing it works. Beyond the idea and the evidence, a procedure is required for carrying out the technology transfer itself. Pfleeger then proposed a new model for software engineering knowledge transfer. This model defines a series of phases to implement technology transfer. In particular, Pfleeger's model addresses the goals of evidence researchers need to collect and how the body of evidence can be trusted and accepted by technology practitioners. Therefore Pfleeger's model is closely tied to the goals of our study.

3. SURVEY DESIGN AND IMPLEMENTATION

The design of the survey relied on the following high level research questions:

- What motivations do software managers have for considering changes in software development processes?
- What types of evidence do software managers want in order to make decisions about changes in the software process?

We also asked some questions about the level of adoption of iterative development processes in order to address some concerns of our corporate sponsor, but will not be reporting those results as they are proprietary.

The survey is composed of four sections. Section 1 contains demographic questions, such as the number of years that the software project manager has worked at the company and the number of projects they have managed. We included demographic questions because it is often the case that the demographic factors moderate relationships between other variables. Questions in Section 2 correspond most directly to the research questions. Questions were asked about the determinants of process changes, as well as the sources of information and evidence that were influential in making the decision to change. These questions were asked in the context of two different scenarios: a hypothetical change and an actual change in the past. We also asked the respondents to describe a proposed future change. In Section 3 we asked about the current software development process being used because we wanted to know if the current model in use has an effect on how decisions are made concerning software development process changes. Section 4

mainly addresses the adoption of the new iterative models being introduced across the company. As mentioned previously, this paper reports only on the results of the first three sections of the survey. The entire survey is semi-structured. Most questions were closed (multiple choice, multiple answer, or a Likert scale), but the survey also included some open-ended questions in order to identify the causes and motivations behind the more closed responses. With these open-ended questions, we could gather rich detailed data that we were able to analyze qualitatively.

The survey was refined several times, e.g. adjusting the order of the questions, before we sent it to the project managers. To further test the survey, we conducted a small pilot on three project managers who were randomly picked from the target population. We appended two questions to the end of the pilot survey, asking respondents if they had any difficulty in answering the survey questions and the time it took them to finish the survey. The feedback from the pilot showed that all the questions were phrased clearly and that they finished the survey within ten minutes. Then we distributed our survey to the rest of the population. The survey instrument was created and distributed online, in order to ease the data collection process, in July of 2007.

The subjects were sampled based on convenience. The survey population consisted of 38 project managers that were chosen by our contact in our partner organization. All potential respondents were software managers at our partner company and were located in the United States. The survey was distributed online, using SurveyMonkey, with an email invitation. SurveyMonkey is an online survey tool that enables people of all experience levels to create their own surveys quickly and easily [12]. Two reminders were sent (plus one from our main contact person at the company) before we closed the survey. A total of 23 project managers responded to the survey, for a response rate of 61%. Among the 23 responses, 19 project managers completed all the survey questions. The survey results presented in the next section are based on the responses of the 19 completed surveys.

4. SURVEY RESULTS

The survey results we present in this section largely follow the order of the survey questions. Starting with demographic questions, we report descriptive results such as the number of years the respondents have worked at the company, the size of their project groups, etc. Then respondents answered questions about the problems in their projects and the motivations for change. Finally and most importantly, they answered questions about the factors affecting their decisions about software development process change, in real and hypothetical situations. The details of the answers are elaborated in the following subsections.

4.1 Demographics

The survey results show that all respondents have 5 or more years experience in software development.

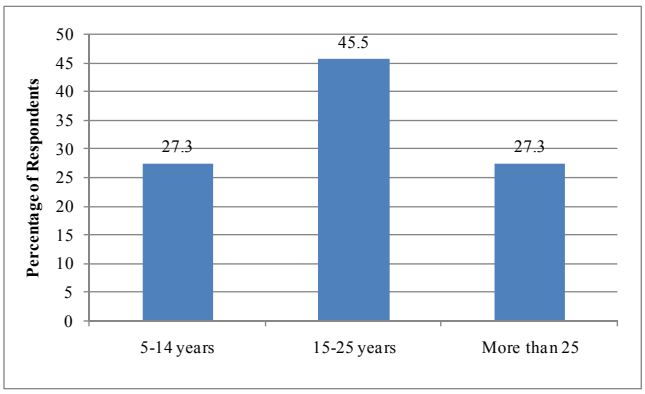

Figure 1. Experience of Software Development

Among them, nearly three quarters (73%) have more than 15 years experience, as shown in Figure 1.

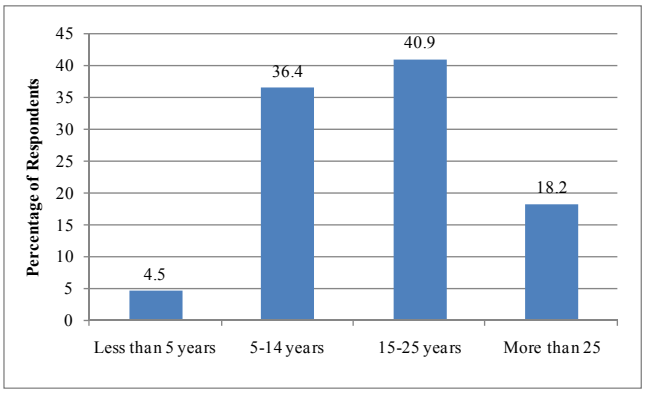

Figure 2. Work Experience at the Company

Almost all of the respondents have worked for the company for more than 5 years, with nearly 60% having tenure of more than 15 years, as shown in Figure 2.

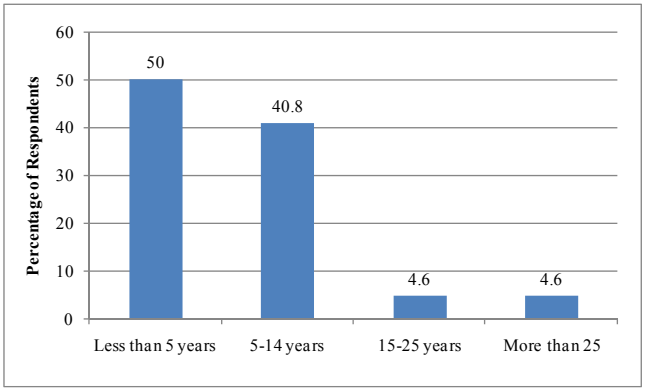

Figure 3. Work Experience at the Current Position

On the other hand, the respondents clearly have not spent most of their careers in their current position. Figure 3 shows that 50% of the respondents have been in their current position for less than 5 years. More than 90% of them have been in their current position for less than 15 years, as shown in Figure 3.

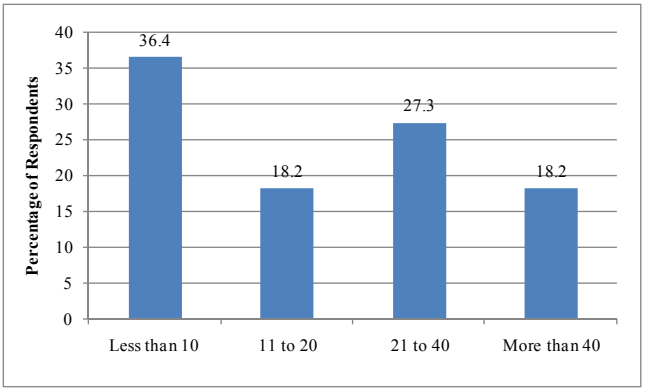

Figure 4. The Size of the Software Groups

When we asked about the projects and the groups they managed, 36% of the respondents said that their group has less than 10 members. Twenty seven percent (27%) of the respondents managed groups with 21 to 40 members, as shown in Figure 4.

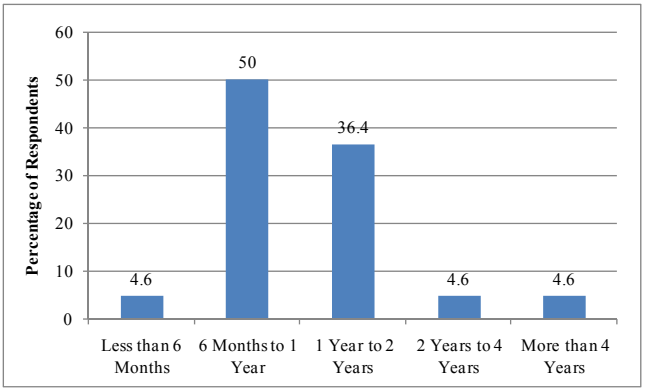

Figure 5. Project Duration

Fifty percent (50%) of the projects the respondents have managed lasted from 6 months to one year. Thirty six percent (36%) of the projects they have managed lasted from 1 year to 2 years, as shown in Figure 5. We also combined Figures 4 and 5 to identify possible correlations between group size and project duration, as shown in Figure 6. Visually, one might notice that very long (more than 4 years) and very short (less than 6 months) projects only occurred with larger (more than 20 persons) teams. However, a Chi-square test found no significant correlations between group size and duration.

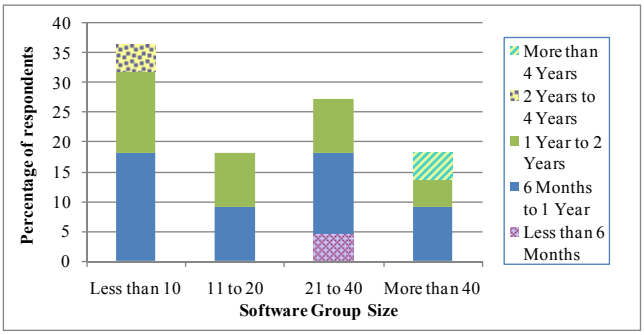

Figure 6. Project Duration by Group Size

These demographic questions exhibit a general view of the nature of the respondents. The majority of the project managers appear to be highly experienced, but have not necessarily been project managers for a long time. Team sizes are varied, but skewed towards small-to-medium teams. Project durations appear to be roughly in the medium range.

4.2 Problems in Projects and Motivations for Change

The respondents were asked how often (on a 5-point Likert scale) the following become problems in their software development projects: (1) fast pace of technology; (2) turnover of staff; (3) requirements changes; (4) tight budget/schedule; and (5) high rework cost. The results show that the most frequent problems that the software project managers encountered are the tight budget and schedule of the project, followed by requirements changes and high rework cost, as shown in Figure 7.

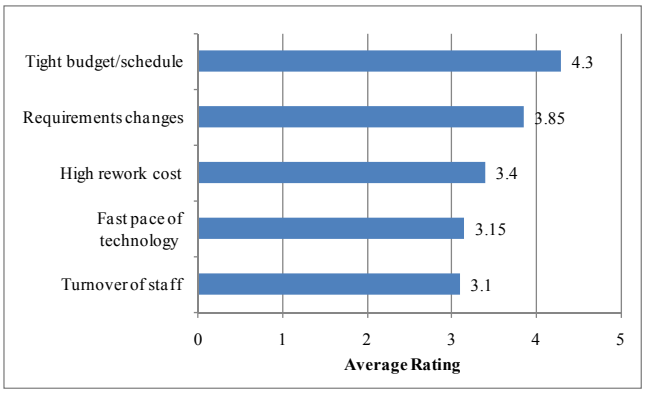

Figure 7. Rating of Problems in Software Projects

Following this question we asked the respondents what in general motivates major process changes in software development in their organization. This was a multiple-answer question (i.e. respondents could choose more than one response). We provided "desire to decrease cost", "desire to increase quality", "desire to manage schedule" and "environmental changes" as the options. At the same time, the respondents could describe their motivations by selecting the "other" option. The top three motivations, each of which was selected by 70% or more of the respondents, for changing the software development process are to manage project schedule, increase software quality and reduce the project cost. Other motivations which were mentioned by the respondents are pressure from the company's competitors, influence of younger leaders and increasing employee satisfaction.

4.3 Software Development Process Change

In the next section of the survey, we wanted to determine what types of evidence and information sources affect decisions about software development process changes. We presented the respondents with a list of options and asked them to indicate which would have some influence on their decision to make a major process change (e.g. adopting a new development paradigm, a new CASE tool, a new procedure for a major software development activity, etc.). The respondents were asked to select all that applied. The options are shown in Figure 8.

This question was followed by an open-ended question that asked the respondents to select the most influential factor listed in the

previous question and give the reason for their choice. We asked these two questions twice, each time in a different context. The results from these queries are presented in the subsections below. In each case, the results were consistent across respondents of varying demographics, e.g. experience and team size.

4.3.1 Hypothetical Changes

First, respondents were asked to answer the questions with respect to a hypothetical scenario ("Suppose they were considering making a major process change…"). In this case, trusted colleague experience, an empirical study using industry practitioners and trusted colleague opinion were the top three factors that would affect their decisions about software development process change, as shown in Figure 8. The respondents also gave the reason for their choices. They trusted colleagues' experience because "it contains real practical evaluation within a familiar environment". One respondent who prefers "trusted colleague experience" gave another reason – "they know the problems that I experience in my day to day job and if they found the new process beneficial, they could relate the benefits back to my needs". Three respondents mentioned that they can learn lessons from trusted colleague experience.

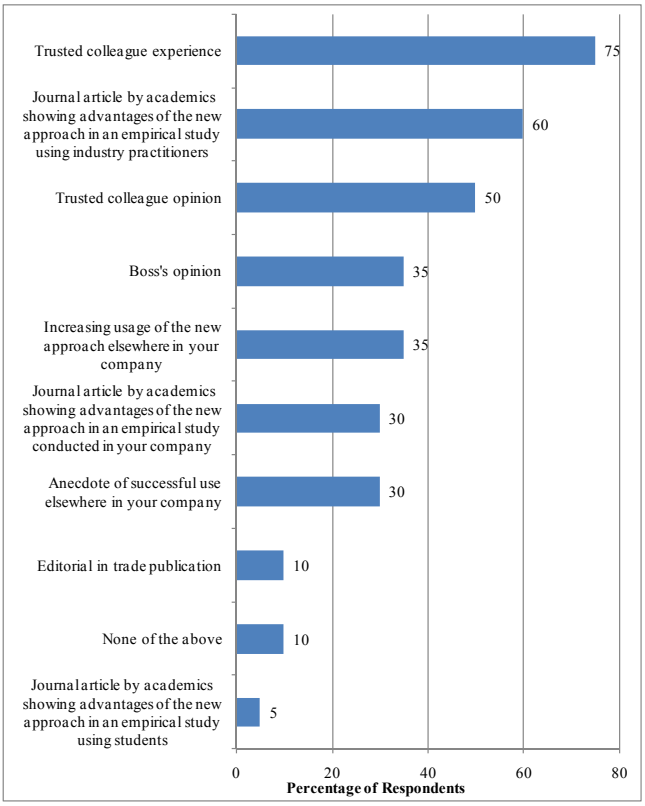

Figure 8. Factors Affecting the Decision in a Hypothetical Scenario

4.3.2 Real changes in the past

Demonstrating the advantages of a new technology does not necessarily lead to a successful technology transition. An effective procedure to carry out the change is also required [9]. In addition, studying the effects of changes can help us to discover the problems of the new approach and the procedure, thus facilitating the possible changes we will make in the future.

For this reason, we decided to ask our respondents questions about what influences their decisions about process change in a more concrete scenario, i.e. about process changes that they actually experienced. We asked the respondents the same two questions described above, but we changed the scenario to the past ("Think of a major process change that was made in the past either in a project you oversaw or that you worked on"). Again, trusted colleague experience took the top spot (35% of the responses) among these factors. Here we noticed that only 35% of the respondents selected the top relevant factor. This is much lower than the level of responses in the hypothetical scenario (compare the horizontal axes on Figures 8 and 9. This is due to the fact that most of the respondents selected fewer options for this question than in the hypothetical case. One might conclude that, hypothetically, managers would expect to consult a number of sources in making process decisions, but in reality only a few sources are actually influential.

Compared with the responses in the hypothetical scenario, a new factor – "Increasing usage of the new approach elsewhere in your company" entered the top-three list and was also tied as the most popular factor (35% of the responses). However, only one respondent thought this factor was the most influential one for decision making and this choice was not explained. "Boss's opinion" and "empirical study using industry practitioners" tie for the third spot in the top-three list (20% of the responses). Figure 9 shows the result.

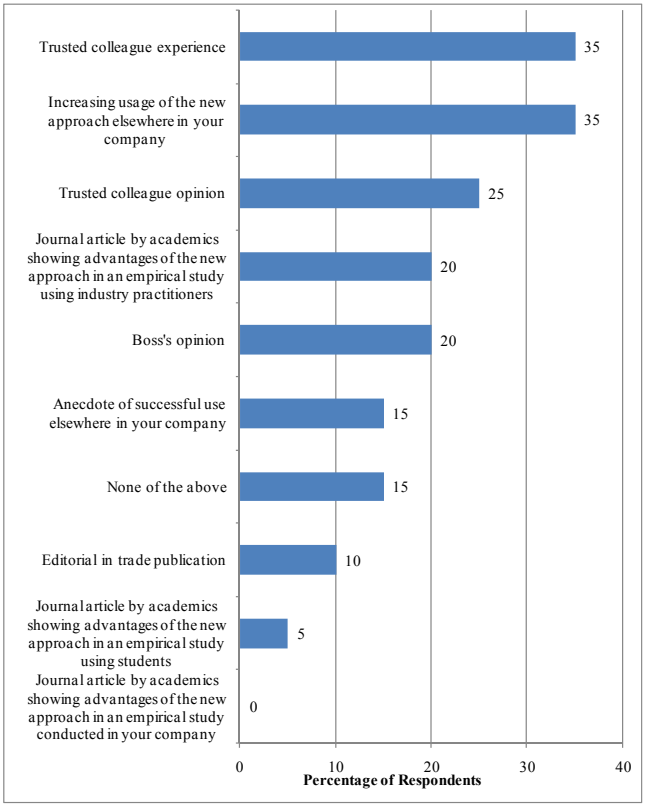

Figure 9. Factors Affecting an Actual Decision in the Past

In this section we also asked the respondents whether there had been a major change in the software development process in any of the projects they oversaw in the last 12 months and, if so, to please describe how that change happened and what effect it has

had on cost, schedule, quality, employees, and any other aspects of the organization or business. This was an open-ended question. The results show that more than half of the respondents experienced such a change within the last 12 months. Most of the responses referred to the evolution to an iterative or agile approach such as Lean or Scrum from the traditional waterfall. The positive effects that the respondents mentioned were improved quality and predictability, employee morale and customer satisfaction. However, some respondents also pointed out that the new approaches increased cost and schedule, that there was a lack of training and that it caused some inconsistency with the company's higher-level processes.

4.4 Potential changes

In this section the respondents were asked if they are currently considering a change in the software development process used in any of their projects and, if so, to select from the following factors those that are motivating the potential change: (1) desire to decrease cost; (2) desire to increase quality; (3) desire to manage schedule; (4) environmental change (e.g. change in competition, employees, technology, etc.); or (5) other. This was a multiple-answer question and the respondents could give factors other than the ones listed by selecting the "other" option. This question is the same as the second question in Section 4.2, but it focuses on the motivations of a specific, concrete potential change. The results show that 55% of the respondents are currently considering a change.

Among the factors that motivate the change, quality, cost and schedule still possess the top three positions. Nine percent of the respondents think environmental change is also a relevant factor. Figure 10 shows the result. "Employee satisfaction" is an example of other motivations.

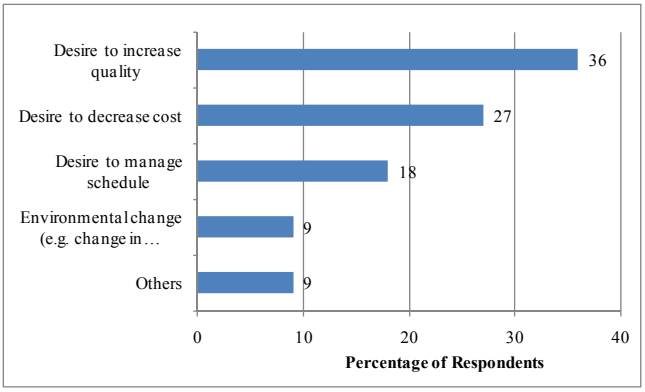

Figure 10. Motivations of the Potential Change

We also asked the respondents what part of the development process they were considering changing and what criteria they will use for making the decision. The parts of the process that the respondents are considering changing vary across a wide spectrum. In their responses almost every phase of the software lifecycle is mentioned, for example, requirement analysis, design, development, and testing. Some respondents, in their open-ended responses, indicated that there were other criteria for deciding what to change, e.g. whether or not it would improve customer satisfaction.

5. LIMITATIONS

Although this survey has a 61% response rate which, compared to other web-based survey studies reported in the literature, is very high, the target population has only 38 respondents. The small number of responses restricts us from conducting statistical analysis on the data in most cases. In addition, all respondents are from one company. Therefore, it is not easy to generalize the applicability of the findings from the survey. This survey must be combined with other methods to provide high external validity for the results. On the other hand, triangulation was used (between quantitative and qualitative responses) wherever possible to strengthen the findings, and the survey itself was piloted and tested in several different ways. Also, we believe our partner organization is representative of large, commercial manufacturers whose products contain a significant software component and who develop most of their software in-house.

6. DISCUSSION

In this paper we present the results of a survey study of software project managers in one large company. The purpose of this survey is to reveal the factors affecting decisions about software process change. Starting from demographic questions, we expanded on the research question by developing a series of survey items that covers general motivations behind changes, the evidence that informs decisions about change, and the effect of process changes in past, current, and hypothetical scenarios.

The most important factor that appears to affect project managers in their decision about software process change is local evidence of various kinds, consistent with the findings of Rainer et al.[5]. For example, the experience and opinions of trusted colleagues was one of the most often selected choices when the respondents were asked about the types of evidence they relied on. The analysis of the open-ended responses indicate that any evidence that is able to show its applicability in a particular setting can be considered "local", hence of high credibility. This conclusion has been confirmed in the different scenarios described in Section 4.3. For example, in the hypothetical change scenario, one respondent indicated a trust in colleagues' experience because "it contains real practical evaluation within a familiar environment". When asked about a real process change in the recent past, slightly different sources of evidence were cited, but the responses still point to the importance of local evidence. The "increasing usage of the new approach elsewhere" and "boss's opinion" were choices that were selected more often in the real change scenario.

Theoretical evidence, e.g. "journal article by academics showing advantages of the new approach in an empirical study using industry practitioners", also plays an important part in the decision making process. One respondent noted that such evidence "would be grounded in the scientific method". However, a "journal article by academics showing advantages of the new approach in an empirical study using students" was not often selected as a valuable source of evidence, further strengthening the conclusion that evidence that is closer to the decision makers' experience is more valued.

An important implication of these results for researchers is that gaining local experience with a new technology is key to increasing its adoption. When attempting to convince decision makers to try a new process innovation, the strategy suggested by our results is to start with a pilot study. The pilot study should be highly visible, so that the results will be more quickly

disseminated to other decision makers. It must also be rigorously designed and monitored, and the results reported with care, so that others will believe in its validity and applicability to their situation. A pilot study strategy is also supported, indirectly, by Kaindl's [4] results, which point to the lack of infrastructure in a research setting for experimentation at the level needed to provide practitioners with compelling evidence. However, Kaindl's work also points to the obstacles faced on the practitioner side, e.g. limited time and budget, which may also be an obstacle even to pilot studies.

When asked about the effects of an actual past process change, one respondent pointed out that the new approach had increased cost and schedule, that there was a lack of training and problems occurred with inconsistency with the company's larger processes. This is a normal part of the learning curve that occurs in the adoption of any new approach. But it also indicates another reason to follow a pilot study approach. A small pilot may have the opportunity to demonstrate the advantages of the new approach past the learning curve, giving new adopters evidence that they will, with patience, reap those benefits as well. This result also indicates the importance of organizational support for a change, another factor that is consistent with the pilot study approach. The piloting strategy is also supported by the comments of another respondent who described the process of a recent successful change. The change had started from recommendations of another group in the company and was based on their previous project-level success.

Our results show that quality, cost and schedule are always the top motivations for software project managers. One respondent used the term "the fire in the belly" to describe the effect of these top factors. This implies that any evidence presented to practitioners about a new process innovation must clearly address the innovation's effect on quality, cost, and schedule. Our results also show that software development processes are still immature in some sense and that process innovations are needed in all process areas. Almost every phase of the software development process appeared in the project managers' list of areas they are considering changing. In particular, most of the respondents indicated that requirement changes and high rework costs are problems they often encounter in software projects. While these problems could potentially be addressed by changes in many different process areas, they point to specific concerns (at least more specific than "quality, cost, and schedule") that should be highlighted when presenting any evidence of a technique's effectiveness. We notice that employee morale and customer satisfaction are also factors motivating software process changes. Some of the past changes described by respondents were said to improve employee and customer satisfaction to some degree. However, it appears that customer satisfaction is a minor issue in our partner organization because it was mentioned by only one respondent.

Effective technology transfer is a concern for all software engineering researchers who want to see their work put to practical use. It is also a concern for all practitioners who want their work to benefit from the most effective practices. However, the adoption of new research-based techniques in the industry is frustratingly slow and difficult. The study presented in this paper provides some empirical evidence for particular strategies and emphases in the technology transfer process, namely a pilot strategy and an emphasis on the innovation's effect on cost, quality, and schedule.

7. ACKNOWLEDGMENTS

We would like to thank Ryan Getek, Huijuan Wu and other students in the IS 805 class of fall 2006 at UMBC for piloting an early version of the survey. This research was supported in part by a research gift from our industrial partner.

8. REFERENCES

[1] F. D. Davis, "Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology", MIS Quarterly, vol. 13, pp. 319-339, 1989.

[2] F. D. Davis, R. P. Bagozzi, and P. R. Warshaw, "User Acceptance of Computer Technology: A Comparison of Two Theoretical Models", Management Science, vol. 35, pp. 982-1003, 1989.

[3] V. Venkatesh, M. G. Morris, G. B. Davis, and F. D. Davis, "User Acceptance of Information Technology: Toward a Unified View", MIS Quarterly, vol. 27, pp. 425-478, 2003.

[4] H. Kaindl, S. Brinkkemper, J. A. B. Jr, B. Farbey, S. J. Greenspan, C. L. Heitmeyer, J. C. S. d. P. Leite, N. R. Mead, J. Mylopoulos, and J. Siddiqi, "Requirements Engineering and Technology Transfer: Obstacles, Incentives and Improvement Agenda", Requirements Engineering, vol. 7, pp. 113-123, 2002.

[5] A. Rainer, T. Hall, "A quantitative and qualitative analysis of factors affecting software processes", The Journal of Systems and Software, vol. 66, pp. 7-21, 2003.

[6] A. Rainer, T. Hall, "An analysis of some 'core studies' of software process improvement", Software Process: Improvement and Practice, vol. 6, pp. 169-187, 2001.

[7] A. Rainer, T. Hall, and N. Baddoo. 2003. Persuading Developers to "Buy into" Software Process Improvement: Local Opinion and Empirical Evidence, In Proceedings of the 2003 International Symposium on Empirical Software Engineering (Rome, Italy, September 30 - October 1, 2003).

[8] M. Kauppinen, M. Vartiainen, J. Kontio, S. Kujala, and R. Sulonen, "Implementing requirements engineering processes throughout organizations: success factors and challenges", Information and Software Technology, vol. 46, pp. 937-953, 2004.

[9] S. L. Pfleeger, "Understanding and improving technology transfer in software engineering", Journal of Systems and Software, vol. 47, pp. 111-124, 1999.

[10] E. M. Rogers, Diffusion of Innovations, 4th ed. New York: Free Press, 1995.

[11] M. V. Zelkowitz, "Software Engineering Technology Infusion within NASA", IEEE Transactions on Engineering Management, vol. 43, pp. 250-261, 1996.

[12] SurveyMonkey, www.surveymonkey.com, 2008.