XP In A Research Lab: The Hunt For Strategic Value

Victoria Bellotti*, Richard R. Burton*, Nicolas Ducheneaut*[¥],Mark Howard*, Christine Neuwirth[†], Ian Smith*

*Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA 94304 USA
+1 650 812 4000
bellotti; nicolas; beki; mahoward; iansmith@parc.com

[¥]School of Information Management & Systems
Berkeley 102 South Hall University of California
Berkeley, CA 94720-4600
+1 510 642 1464
nicolas@sims.berkeley.edu

[†] Department of English and Human-Computer Interaction Institute
Carnegie Mellon University
Pittsburgh, PA 15213
+1 412 268 8702
cmn@cs.cmu.edu

ABSTRACT
We describe our experience of using XP to develop experimental software in a research setting. We focus on the ways that a research project is different from a commercial project. Our experience leads us to conclude that XP practices can be extended to allow a new understanding of value in a software project. We draw a distinction between strategic and tactical value and explore how different settings affect the type of value a project seeks to develop. We suggest ways that research projects might benefit from the use of XP.

Keywords
Extreme Programming, metaphor, strategic value, tactical value, software process, research setting.

1 INTRODUCTION

According to Ron Jeffries, the Extreme Programming (XP) model is one in which "customers define value and engineers create value" [9]. This paper will explain why we believe the notion of "value" needs further refinement. Our work in a research laboratory has led us to conclude that there are two kinds of value in a software project, *tactical value* and *strategic value*. Tactical value in an XP project is manifest in the features and function of the software system and its timely delivery. Strategic value in a XP project, however, is drawn from the underlying business model that was the inspiration for conducting the software development effort itself. We believe this distinction has gone unnoticed in the literature because in a commercial setting the strategic value of a project is generally tacitly understood by all parties and in fact dictated by a paying customer. In a research environment, however the goal of a project is often to ascertain what, if any, strategic value is present.

As an example consider Alice, the manager of a small print shop. In an effort to improve interactions with her suppliers she engages an XP development team to build a web-based supply chain management application. The XP team creates "value" for her through the XP process in the development of the desired application. This value is tactical because it is measured in terms of the software's contribution to the explicit goal of improving supplier relations.

What is not considered in this example is any question of whether improving relations between Alice and her suppliers will make a positive contribution to the future of her business. Strategic value addresses this type of concern. If Alice's business was in a period of rapid contraction she may have to address issues of strategic value. For instance should she migrate to the sign-printing business or to the books-on-demand business?

In the research setting it is common for the only questions at issue to be ones of strategic value. For instance it is often the case that researchers will have a "technology in search of a business" and will conduct studies and trials to try and determine if a compelling application can be developed using their technology. Our research process uses XP in two ways. First to develop software artifacts that help illustrate or measure strategic value. Second, to provide a catalyst for discussion among group members as to the strategic value of the project.

The XP practice of "metaphor" [1] comes closest to capturing the nature of strategic value. Kent Beck describes metaphor as "a simple shared story of how the whole system works." XP expects metaphor to express a sense of system cohesion. We believe that the notion of metaphor must be extended to also be a simple shared story about *why the whole system is interesting* and later in this paper we propose a specific technique for developing this shared story. Our hope is that an extension of the XP notion of metaphor can be used to focus discussions of strategic value in the same way that XP planning processes focus discussions of tactical value.

This paper is organized as follows. The next section will detail the particulars of our team and setting. The third section will give a detailed description of a five-month experiment in searching for strategic value using XP. The fourth section will include problems and success we encountered using XP in a research environment. The final section will give our conclusions about this project.

2 OUR XP SETTING

This project was a continuation of the work on designing

software to embed personal information management resources in an email tool conducted in 2000 and 2001 [3,4]. The XP portion of the project began in August 2001. Our project team consisted of three fieldworkers, two engineers, and one manager. The project's "goal," during XP phase that we report on in this paper, was to build software on which an experiment to test some of our research hypotheses could be conducted. As will be shown in later sections, we developed our understanding of the strategic value of the project through the development of this test apparatus.

Fieldworkers conduct studies using methods derived from sociology and anthropology. They conduct *ethnographies* (in-situ interviews and observations) strongly oriented to support the design process for software. In addition, all the fieldworkers in our group have extensive experience in the area of HCI (Human Computer Interaction). This combination of study of actual practice plus experience with user interface design made them excellent advocates for customers of a new software system. We refer to them throughout this paper as "customer representatives" rather than customers to emphasize the distinction between true customers and those that study customers.

The two engineers were motivated by their own interest in XP and were anxious to try out the methodology in our setting. Further, previous projects in our group had lacked timely feedback about the value of software developed [3]. On reflection these previous software development efforts did not have a cohesive understanding of the project's strategic value. Rather there were many disparate ideas about the project's strategic value. At the beginning of our XP activity we hoped that XP might help us create a shared understanding of the strategic value. Note that although the fieldworkers were playing the role of customer representatives in our project, this was a notional distinction and all project members were primarily motivated by their role as researchers first, and either engineers or customer representatives second. We will show later in the paper that this seemingly small distinction in our roles had important consequences.

We chose to use short XP iterations—about one- to one-and-a-half weeks—in an effort to increase the frequency of feedback.[2] Since we have only two engineers, obviously all pair programming occurs within this pair. We attempted to conform to "standard XP" [7,8] as much as possible.

The focus of our design effort has been a tool that improves support for the kind of intensive multitasking and interpersonal coordination work that we have observed users attempting with existing email tools. We believe that the customer value of our design ideas can only be assessed by observing users handling their real email with our prototype, rather than trying out look-and-feel evaluations of superficial user-interface features. XP has proved to be an excellent discipline for maintaining a fully functional prototype that works with real data throughout our project.

[2] The team is now interested in trying longer iterations as an experiment to test our ability to accurately estimate.

3 EXPERIENCE

In this section, we detail our experience using XP. We report three types of experience:

- Anecdotal experiences interesting to the XP community at large.
- Experience unique to using XP in a research environment, or introducing XP into such an environment.
- Experience using XP to develop strategic value.

Pre-project

In line with the team's previous engineering methodology, the customer representatives drew up a two-and-a-half page requirements document. An initial planning meeting was scheduled with the whole team to review this document and improve the requirements it contained. The document was typical of traditional software processes, it included a great many features with little thought to how the system in general would behave, or how the features would interact with each other. This document was not 'wrong' as such but neither was it helpful in guiding engineering decisions. We believe a traditional software process starting from this document would not have achieved the same quality of results that were accomplished using XP.

Given that some project spikes had revealed the feasibility of the basic design and that the engineers were committed to the XP process, the engineering staff hijacked this initial "requirements meeting." Rather than spend time lobbying for the adoption of XP as a development process before the project began, engineering simply asked the other members if they wouldn't mind writing their ideas for features down on story cards, then picking a set of those cards that they thought were of highest priority. Before engineering had fully explained the XP method, the customer representatives generated an initial story set of 34 stories in a two hour meeting.

It will come as no surprise to experienced software developers that this first set of stories revealed several miscommunications in the original requirements document. In particular, different customer representatives meant different things by design suggestions in the original requirements document. These disagreements lead to excellent discussions and design work to hammer out what the customer representatives really meant. The customer representatives were both surprised and pleased to have discovered these disagreements at this early stage. They also enjoyed the initial XP hijacking of the planning meeting and were consequently receptive to the suggestion that XP be used as the projects engineering methodology.

Iteration 1 and 2

Due to our inexperience with the method there was a high ratio of planning to engineering effort at this stage. In addition, the team spent significant time in meta-discussions about the utility of XP in a research setting

and about features and assumptions of the methodology itself, e.g. it was paraphrased by one engineer that "XP coding practices are about building the software right, whereas XP planning practices are about building the right software" [5].

Immediately before iteration one, engineering introduced the customer representatives to the XP process (and gave them copies of the books) in order to play the planning game. During this first planning game the customer representatives had their first major realization of the possibly harmful consequences of a design decision. This realization came about due to intense discussions with the engineers regarding story cards, and resulted in the customers accepting that one of their possible user communities would be disadvantaged by their decision. In later iterations, these planning game discussions would prove to be a critical part of our strategic decision making process. At this stage, our understanding of the strategic objectives of the project was not yet well formed.

Iteration 3

This was the first iteration whose initial estimates were lower than the actual work completed. In only two XP iterations, engineering had learned the value of caution. Also at this point, engineering began giving story estimates in "points" rather than "idealized hours." Expressing accomplishment in terms of idealized hours was misleading because it gave the impression that the engineers were only working a small number of hours per week. This problem was particularly acute for those not familiar with the XP process; the word "hours" had specific connotations for them, and it was hard to establish in the minds of others that there was a distinction between "ideal" hours and "real" hours. An abstract point system separates the planning and review process from any real-world connotation.

Iteration 4

During this iteration we noticed an increasing laxity in the customer representatives in developing acceptance tests. This is in contrast to the planning game that was accepted and adopted easily. We believe that there are three reasons for this:

- The planning game, "driving the car" in XP parlance, gives immediately positive feedback to the customer. The "customer is king" maxim here gives the customer immediate and explicit value in that they see their decisions have effect on the direction of the project. Conversely, the value of having comprehensive acceptance tests is deferred and implicit in that it saves time and effort by avoiding future problems.
- Fieldworkers spend immense amounts of time designing, implementing, and analyzing the field studies of users. This is a large contribution to the tactical value of the project as a whole. Normally one would expect these experiments to be their primary such contribution and acceptance test generation seems to be an additional burden. We feel this is a consequence of the XP maxim 'the customer is king.' The king sometimes has to perform onerous acts on behalf of the kingdom [3].
- Our project had no dedicated tester to guide the customer representatives in the design and implementation of acceptance tests. Thus the role of dedicated tester [6] was thrust on the customer representatives with little support from engineering.

Iteration 5

The fifth iteration brought the first observed instance of serious bargaining to the planning game. The customer representatives were faced with a choice between implementing a set of features that would contribute significantly to the effectiveness of their experiment or conducting some much needed software installation and hardware system housekeeping that would allow experimental design to proceed apace. In fact the customer representatives chose to undertake the four-point housekeeping story card themselves rather than spend engineering resources on them. This bargaining session clearly illuminated for the customer representatives that not all requests are appropriate for engineering resource expenditure.

Iteration 7

The XP planning game for this iteration exposed a significant conflict. Over the preceding weeks, two separate theories regarding the project's strategic value had emerged:

- Plan A: The fieldworkers maintained their focus on an experiment as the strategic objective. This goal had undergone refinement as a result of their involvement in the XP process, but was in line with their original strategic thinking.
- Plan B: Engineering had developed an alternative focus as a result of their ongoing exposure to a working prototype—due to the XP practice of continual integration. Engineering's exposure led to many new, innovative ideas and a desire to explore those ideas.

These ideas come into conflict during the planning game because they imply different prioritizations of features. If plan A was accepted as the strategic direction of the project then a very different set of story cards would be recognized as of the highest priority than those that would be chosen if plan B was to be the projects strategic focus. In particular, for an experiment under plan A to justify the amount of effort required to conduct it, many resources must be spent to build (and test) well-established features that study participants would expect to find in real, production applications.

Our group's fieldwork methods insist that in order to conduct a realistic evaluation, the test subjects must use the software in realistic settings for realistic tasks. Implicit in these methods is the belief that "toy systems" cannot deliver realistic feedback about value without including standard features (such as 'print,' 'import legacy data' and 'undo'). Basic usability problems would 'muddy' the results of the evaluation of the new features of interest. However, repeating the efforts of others to develop these features would seem to be a misuse of

resources under plan B, where exploration of new, innovative features would be more productive.

It was felt that this disagreement was significant in our setting because although the fieldworkers were acting as our customer representatives, this was not as concrete a role for them as might be found in a commercial setting, all project members were equally focused on the question of what might be the most strategically productive goal of the project, and each project members opinion held considerable weight in the minds of their colleagues.

The XP planning process can be thought of as an exercise in "steering the car" [1] of software engineering which is extremely effective when someone has a map and a destination in mind, in our research setting however our car was sitting in the middle of a desert, and choosing the best destination out of that desert was not obvious.

In the end this disagreement was resolved in favor of plan A. The pragmatic reason for this decision was that significant resources had already been committed to plan A, i.e. many hours of preparing for the experiment, recruiting subjects, etc. and further, doing Plan A would not prevent future work on Plan B should that be deemed worthwhile. Fortunately for our experiment in the use of XP, this pragmatism coincided with the XP practice of making "the customer king." Following this meeting, many debates about strategic choices were resolved quickly with the catchphrase, "we're doing plan A."

Thus, only 2 months into our project, XP helped to uncover an important distinction between two hitherto implicit and confounding research strategies (plan A and plan B), which manifested themselves as a clash of equally important but different values between engineering (software designers) and customer representatives during planning game sessions. From this distinction we were better able to understand the implications of the "customer is king" axiom in a research setting as opposed to a commercial one:

Customer is king if your research strategy is to test the *value of your design concepts to customers*.

Customer is not king if your research strategy is to *reify, explore and demonstrate innovative design concepts*.

Iteration 8 and 9

Things went awry in iteration eight. One member of the engineering team went on a two-week vacation in the middle of this iteration. Compounding the problem was the fact that the estimate for one of the story cards in this iteration was wildly optimistic. As it turned out, this story card implied a virtually complete rewrite of the system and its accompanying test machinery[3]. The engineer that was left to work on this section commented later, "This started to feel like a normal, out-of-control software project." Iteration eight was such an unmitigated disaster that the decision was made to make the goal of iteration nine simply to be cleaning up the mess from iteration eight.

Iteration 10

At this point, both the customer representatives and the engineering staff were quite comfortable with XP. By this time XP "story cards" and the meetings around their generation and selection had become the primary vehicle for communication within the group. For example, as engineering discovered an area of under-specification in the stories, two XP-inspired actions occurred. First, engineers would "do the simplest thing that could possibly work" to meet the stated objectives of the cards. Second, with the strategic objectives of project in mind, the engineers wrote new (possibly improved) stories and left them on the chairs or desks of customer representatives. These stories were considered in the next planning meeting. This practice continues to the present.

Iteration 13

For the first time, someone outside the project group began using the software.

4 OUR PROBLEMS AND SUCCESSES WITH XP

Problem 1: Fluid roles

XP is extremely effective at aligning developer interests with customer interests, based on the assumption that the customer's strategic vision *is* the vision of project. (If the customer is paying for the development this is always true.) In a research setting, however, this assumption does not hold. As was seen in iteration seven of our experience, engineering priorities become difficult to reconcile when there are multiple competing models of the strategic project value.

To remedy this difficulty, we recommend an extension of the XP notion of metaphor. In order to bring to the foreground questions of strategic value, we propose the *metaphor game*. We suggest that all team members play this game at the time of release planning. The metaphor game contributes to strategic value by making explicit the team's shared vision. Analogously, the planning game contributes to tactical value by making explicit the team's immediate priorities.

We propose that the metaphor game's rules be as follows:

- Team members generate metaphor cards. Metaphor cards are statements of philosophy about the entire system. These statements can be guidelines, constraints or actual metaphors.
- A set of metaphor cards is chosen by consensus to represent the project's strategic goals. These cards must all be logically consistent and this decision is again by consensus. Should metaphor cards become inconsistent as understanding changes over time, the cards must be removed or refined.
- Engineers must consider the "simplest thing" to implement a given story card in a way that is consistent with the current set of metaphor cards.
- During the planning game, engineers can give an infinite point rating to story cards when they are inconsistent with the current set of metaphor cards.

[3] Engineering learned a valuable lesson in this iteration, namely factoring your test suite is at least as important as factoring your production code, if not more so.

The authors appreciate the reticence of some readers to accept our reliance on achieving consensus in the metaphor game. Indeed, XP uses specific roles in the planning to avoid the need to establish such a consensus—the customer is king. In our experience, the presence of explicit vision statements—even conflicting ones—makes the process of achieving consensus far easier than one might think.

Problem 2: Focusing on the details
A tenet of XP is continuous integration and the constant development of working software. This presents problems when trying to establish strategic value because it is common for evaluators of a piece of software that appears professional (and working!) to focus on details like fonts, scrollbars, and icons rather than the strategically important factors [2,10].

For example, customer representatives have prioritized adding drag and drop features to our prototype system in addition to already completed keyboard accelerators that fulfill the same function but are not intuitive. It is difficult to avoid the tendency to think of the system as a finished, shrink-wrapped application when it works as well as it does!

It could be argued that had we applied standard XP's recommendation of developing the application with a "spartan interface," this problem might be ameliorated. However, in our case our customer representatives placed a high priority on usability issues.

Success 1: A catalyst for strategic value analysis
This is the flip side of problem 2. Having a continuously working prototype keeps discussions of strategic value tied to realistic issues. In previous non–XP projects we believe strategic discussions took place in a "blue sky" environment without significant grounding in actual experience.

For interacting with people outside the immediate team an additional benefit is seen. The prototype acts as a focus for discussing ideas that are murky and not yet well articulated. Without such a focus, it is easy for discussions with external parties to be sidetracked about what the project *might* be.

In a research setting, conversations of this form are frequent. We feel many other projects in a research setting could benefit by having a continuously integrated artifact to demonstrate their ideas.

Success 2: Improved communications
As would be expected with XP, communication between project team members has improved dramatically. Our team members have radically different backgrounds, computer science, psychology, HCI, business, philosophy, and more. Having a common vocabulary of stories, points and iterations, allows all team members to contribute effectively.

For instance in previous projects it was impossible for non-technical team members to contribute to discussions of a technical nature. With XP however, this occurs frequently since those non-technical individuals can reason about the relative point-costs of particular proposals. While the tasks required to complete a particular story may appear arcane to the lay person, it is always obvious that a card estimated at six points is twice as difficult as a card marked three points, and that if changing the definition of a story on a card reduces it's estimate, then it is easier to accomplish the revised story.

5 CONCLUSIONS
Although our experience with XP was fruitful, it exposed the difference between strategic and tactical value. We believe that this difference is exacerbated in a research environment because strategic value questions are paramount. In the future, we plan to experiment and evaluate the metaphor game as a means to make explicit strategic value discussions among project members.

It is our belief that the issues around strategic value we discuss are relevant not only in the research setting, but also in at least some commercial settings. For instance, it is often the case that in a small "start-up" company it is not always clear what the best strategic direction for the company will be, it is not always clear who the company's customer will be, and it is often the case that everyone in the company feels a strong desire to be involved in the process of defining the company's strategic direction. We believe that in these circumstances also, many of the issues we raise will be significant.

Acknowledgements
We would like to thank our management for supporting our maiden voyage into XP-land and Trevor Smith for his insightful comments about XP, in particular XP in a commercial setting. We would also like to thank the members of the Bay Area XP Group for their valuable comments.

7 REFERENCES
1. Beck, K. (2000) *Extreme Programming Explained*. Reading, Massachusetts, Addison Wesley.
2. H. Beyer and K. Holtzblatt, Contextual design: Defining Customer-centered Systems. Morgan Kaufmann, 1998.
3. Bellotti, V., Ducheneaut, N., Howard, M., Neuwirth, C. & Smith, I. (in preparation). Innovation in Extremis: Evolving an Application for The Critical Work of Email and Information Management. Submitted to *Conference proceedings on designing interactive systems: processes, practices, methods, and techniques*, 2002.
4. Bellotti, V., & Smith, I. (2000). Informing the design of an information management system with iterative fieldwork, *Conference proceedings on designing interactive systems: processes, practices, methods, and techniques* (pp. 227-237). August 17 - 19, 2000, Brooklyn, NY United States.
5. Barry W. Boehm. Software Engineering Economics. Prentice Hall, Englewood Cliffs, NJ, 1981
6. Crispin, L., House T., & Wade C., The Need for Speed: Automating Acceptance Testing in an

Extreme Programming Environment, *Extreme Programming Examined (pp. 96-104)*. May 20-23 2001, Villasimius, Italy.

7. Extreme Programming. http://www.extremeprogramming.org
8. Jeffries, R., Anderson, A. & Hendrickson, C. (2001). *Extreme Programming Installed*. Reading, Massachusetts, Addison Wesley.
9. Jeffries, Ron. Webinar Customer Program (online presentation).http://groups.yahoo.com/group/extremeprogramming/files. 20 Nov 2000.
10. Landay, J.A. and Myers, B.A. Interactive sketching for the early stages of user interface design. In Proceedings of CHI '95: Human Factors in Computing Systems, Denver, CO, May 1995, pp. 43-50.