

Customer Involvement in Extreme Programming

— XP2001 Workshop Report —

Arie van Deursen

CWI

P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

<http://www.cwi.nl/~arie/>

Abstract

This paper covers customer involvement challenges in lightweight software development processes. The report summarizes the presentations and discussions of the *Workshop on Customer Involvement* held during XP2001, the *Second International Conference on Extreme Programming and Flexible Processes in Software Engineering*, Cagliari, Italy, May 21, 2001.

1 Customer Involvement

Active customer involvement is crucial to any software development project. For that reason, extreme programming (XP) insists on an *on-site customer*, who has many different tasks:

- Understanding customer wishes, maintaining regular contact with end users, and balancing their potentially conflicting interests.
- Talking to developers, clarifying feature requests when needed, and understanding some of the developer's technical concerns.
- Specifying functional tests for user stories, and verifying that these tests run correctly.
- Participating in the planning of iterations and releases.
- Maintaining good contact with management, explaining progress, and justifying the time spent with the development team.

Success of an XP project strongly depends on the way in which the on-site customer is able to fulfill these roles.

Compared with an XP developer, the on-site customer only has a small bag of methods and techniques he or she can use to deal with the various problems encountered. For improving the design, for example, the XP practitioner has a catalog of smells and refactorings available that enable him to recognize danger signals and

attack them in a systematic manner. Similarly, the on-site customer would be much helped by systematic ways for diagnosing problematic situations, and explicit guide lines for dealing with recurring problems.

To address these topics, we initiated a *Workshop on Customer Involvement*, which took place during the XP2001 conference on Extreme Programming. The purpose of this workshop was to bring together people interested in customer involvement in software development projects, in order to share experiences and collect guidelines for successful customer involvement. This report summarizes the main results from this workshop.

The schedule of the workshop is listed in Table 1. The workshop was attended by 20 participants, of which 2 were from academia, and 18 from industry. The format of the workshop included plenty of space for discussion, both at the start and the end of the workshop, as well as before and after each presentation and session. This report covers the topics raised in the presentations as well as the key issues addressed in the discussions.

2 The C3 Project

Ron Jeffries (who considers himself “the most extreme of the three extremos”) reported on the role of the on site customer in the well-known *Chrysler Comprehensive Compensation (C3)* project. In this very first XP project, two full payrolls were developed for Chrysler. The project began with approximately 145 large user stories, with a total estimated duration of one year.

The on-site customer in this project had a vision of the perfect system she wanted to develop. She was able to provide user stories that were easy to estimate. Moreover, she was with the development team everyday, answering any business question the developers had.

Half way the project, several things changed, which eventually led to the project being cancelled. One of the changes was the replacement of the on site customer, showing that the actual way in which the customer is involved is one of the key success factors in an XP project. The new on-site customer was present most of the

09:00	Welcome
09:00	Introduction
09:30	<i>The Customer Role in the C3 Project</i> Ron Jeffries (ObjectMentor, USA)
09:45	<i>Does XP Need a Professional Customer?</i> Malte Finsterwalder (Hamburg, Germany)
10:00	<i>Customer Involvement Experiences in a Software Product Line</i> Arie van Deursen (CWI, Amsterdam)
10:15	<i>Pairing the Customer: Software as Wanted versus Software as Needed</i> Jürgen Ahting (AMECO, Germany)
10:30	Discussion
10:45	Coffee break
11:00	<i>The Customer is Always Right?</i> Steven Fraser (Nortel, USA)
11:15	<i>The Role of the XP Customer in a Product Company: Toward a Customer Team</i> Kay Johansen (Verio Web Hosting, USA)
11:30	<i>Getting the Customer Off Site and On the Team</i> Tom Ayerst (Dresdner Kleinwort Wasserstein, London)
11:45	Discussion
12:00	Wrap up
12:30	Lunch

Table 1: Program of the Workshop on Customer Involvement

time, just like the previous on site customer, and available to the development team for questions. Unfortunately, the requirements and user stories were not as crisp as they were before.

In retrospect, the team could have improved the transfer from one customer to another in the following ways:

- In order not to depend on a *single* person too much, create a *customer team*;
- Explicitly introduce a *customer training*, so that the on-site customer knows better what to do and what techniques to use.

3 The Professional Customer

Malte Finsterwalder argued for professionalizing the role of the customer. Being a customer requires a number of skills that are independent of the application domain. These include balancing potentially conflicting end-user needs, experience in requirements gathering, reporting to upper management, controlling the budget, and checking for forgotten requirements.

A customer trained in these skills with relatively little domain knowledge could be a valuable resource for an XP project. This trained person should ensure that the different stake holders talk with one voice. This implies that the professional customer must have the authority to make decisions in case of stake holder conflicts.

A problem discussed during the workshop was that such a professional customer will act as a mediator. This may block direct contact between end users and developers. Part of the training of

such a professional customer should be teaching him how to facilitate and coordinate direct contacts. It is his responsibility to manage the customer contact and make sure that the appropriate persons and different stakeholders communicate with the development team directly.

4 Problems Encountered

Arie van Deursen reported on his customer involvement experiences during the (successful) development of a product line called DocGen for the *Software Improvement Group*.

A first observation during this project was that it involved not just one customer, but with many different ones. These include the end user (and even he comes in many categories), the system maintainers, and management, as well as the marketing people and the developers responsible for implementing particular customizations. Making these “speak with one voice” involved a lot of creativity during the project.

Another observation was that the project requirements changed frequently. This was partly due to the fact that new people were brought into the project, bringing new requirements. Another reason was that in several cases the people involved in the project easily changed their minds. XP in itself is capable of implementing non-stable requirements. However, the resulting system may be an incoherent collection of features, ultimately leading to project failure.

Furthermore, technology and potential customers experienced a significant semantic gap when trying to talk to each other. In the DocGen case, the developers had a compiler construction back-

ground, whereas the end users came from a mainframe background. This difference was amplified by the fact that neither the developers nor the customers considered talking to each other as their “real job”, easily considering time invested in talking to each other as wasted.

The following solutions were discussed:

- If there are many different customer categories, consider performing a classical stake holder analysis. Construct a customer team in which it is clear by whom each stake holder is represented.
- Find a way to teach the customer the costs of incoherence.
- Explicitly introduce a training concerning customer involvement, both for the customers and the developers.
- Consider using a technique adopted by Connextra (UK) asking the customer to add a “... so that ...” clause to each user story, which forces him to think about the higher level goal achieved by the given story.

5 The Customer Pair

Jürgen Ahting raised the issue of building the right system versus building the system right. Many of XP’s practices help the development team to build the system right. It is the responsibility of the customer to ask for the right system, being helped by feedback from the frequent releases of the system.

Ahting, however, found that the biggest fear of his clients was that the “software contractor will deliver software as we want it, not as we need it”. To address this concern, he proposes the notion of *customer pairs*, consisting of a domain-expert also having a programming background, and a client-customer responsible for “selling” the emerging system to the client stake holders. Having a customer pair has a number of advantages:

- Having two persons represent the customer increases the customer’s accessibility to the development team.
- The package “domain expert + XP development team” can be sold more easily;
- Selling pure XP is like trying to sell “individual travel” to “package tourists”.

6 The QA Team

Ahting’s proposal for the customer pair triggered a discussion on the role of the Quality Assurance (QA) team to support the customer. The QA Team helps the customer specify the acceptance tests for a story, thereby gathering more detailed requirements for the story and making sure it is testable. In particular Workshare (UK) explained the way in which they organize their customer involvement. Workshare has hired several full time customers, who were active product users in their previous jobs. There is a ratio of approximately one customer upon five developers. For each user story, a developer, tester, and product manager is assigned.

The tester comes from the QA team, and is responsible for writing acceptance tests for the stories. The overall development is guided by a steering committee, consisting of senior developers and senior customers.

The tasks of the QA team at Workshare include the following:

- Straight QA, ensuring that a tester, developer and customer is allocated for a story, that the tests are written, and that the tests are run;
- Setting up and maintaining dedicated machines for automated acceptance tests, run by QA;
- Ensuring that QA activities are factored into the process, conducted on a story by story basis rather than iteration by iteration.

7 Is the Customer Right?

Steven Fraser acted as a customer in an XP project involving the construction of a web-based conference registration system. His position, as a customer, was that *the customer is always right*. In his experience three key customer involvement factors are essential:

- Knowledge of the domain in order to identify solution opportunities (buy versus build, validation of operational requirements)
- Mutual understanding of schedule constraints in order to deal with system requirements as well as a 40 hour week (in this project, there was a 24 hour week for some staff).
- Clear feature prioritization, in order to have a (useful) system up and running in a 24 * 7 service from day one onwards.

This project also displayed the importance of the customer being able to provide incentives (bonuses or penalties) to the development team in order to meet certain deadlines.

8 The Customer Team

Kay Johansen reported on the role of the on site customer in a product development environment. In one of her projects, two people were selected to define the features and user interface: a product manager and a user interface designer. The product manager had been hired from one of the main customers. He had spent over a year in the field doing on-site product training for end-users. Therefore, he was in tune with the reactions of all of the significant customers, and understood their business processes as well as their front-line employees’ data entry styles. The product manager was willing to take the responsibility of coordinating the product rewrite, was an excellent meeting facilitator, and was willing to spend significant time with the development team.

An important success factor in this project was the ability of the development team to predict how much work the team could get done. The product manager could use this information to prioritize, make hard decisions for scope reduction if necessary, and to negotiate with senior management.

Based on the experiences of this project Kay Johansen made the following recommendations:

- Form a product management team, including at least two people performing the following roles: marketing, project management, and technical product management. The technical product manager would be the primary XP customer.
- Form a requirements team, combining the QA and the product management team, to be responsible for capturing, defining, and automating the product requirements in acceptance tests.

Although XP projects will try to attract star customers, setting up a team of customers will help to avoid depending too much on the qualities of a single person.

9 The “Development Offsite”

Tom Ayerst explained how his team at the Dresdner Kleinwort Wasserstein investment bank in London manages to show sceptical, high cost, busy business customers the benefits of continuous involvement. Many of these customers get paid through a bonus system, in which time spent not directly resulting in bonuses is costly to the customers themselves.

Involving such customers is done through a so-called “Development Offsite”. Given a business with a development requirement, this includes the following steps:

- Take around 40 developers and the business customers to a remote location.
- Split the developers into teams.
- Get the customers to explain their requirements.
- Let the teams compete to build a fully functional solution in three days, making presentations of work in progress at the end of each day.
- Let the customers stay with the teams to answer any questions the developers may have.

The customers determine which team has built the best system. This team gets a prize, conditional on the winning solution going into actual production within a set period after the end of the Offsite.

Such an Offsite has a number of effects:

- An actual product is built and deployed in production.
- The developers and the customers really get to know each other in three successive 14 hour days. The atmosphere is competitive and full of excitement.
- Customers get an appreciation of how hard development is, and how much more the developers can achieve when they can ask questions on demand.

- On return to the office the ice has been broken. Customers usually make arrangement to make themselves more available to developers to answer questions, and developers find it easier to approach customers.

During the discussion three other aspects were raised. After the Development Offsite, the customers were willing to spend two hours a day talking with development instead the zero hours that were usual. Moreover, the Development Offsite can be used to teach customers that (1) it takes effort to build software; (2) they get what they ask for; and (3) they had better watch carefully what they ask for, and invest some time in talking about it. Last but not least, the Development Offsite can be an ideal way to bootstrap an XP project.

10 Summary

The workshop consisted of 4 hours packed with discussion and exchange of ideas and experiences on the topic of customer involvement. The main results from the workshop include:

- It makes a difference to have a customer on site with the development team. Don't compensate for not having one – try harder to get one.
- The XP on-site customer is not a single person: it is a role best played by a team of people representing the various system stake holders.
- The QA team is quintessential in supporting the customers in defining acceptance tests for the user stories.
- Customers are responsible for asking for the right system to be built. They need to be helped in every possible way with this hard task. One simple technique for helping the customer to see his long term goals is to add the phrase “so that ...” to every user story.
- Explicit customer training is needed to help customers take maximal advantage of the short iterations, rapid feedback, and commitment to estimate of XP.

The full versions of the papers are available on the workshop's website at <http://www.cwi.nl/~arie/wci2001/>.

Acknowledgements Thanks to all the presenters and participants for bringing up numerous questions and answers during the workshop. I would like to thank Malte Finsterwalder for his feedback on this report.