

# Reflections on Participatory Design: Lessons from the Trillium Experience

*Jeanette L. Blomberg*

Xerox Palo Alto Research Center  
3333 Coyote Hill Road  
Palo Alto, CA 94304  
415-494-4751, Blomberg@parc.Xerox.com

*Austin Henderson\**

Fitch RichardsonSmith  
10350 Olentangy River Road  
Worthington, OH 43085  
614-885-3453, Henderson.chi@Xerox.com

## ABSTRACT

In recent years system engineers, product designers, and human interface designers have become increasingly interested in developing ways of involving users in the design and evolution of computer-based systems. Some have turned for guidance and inspiration to an approach to systems design pioneered in Scandinavia and often referred to as Participatory Design. In this paper we examine the development of a computer-based design tool, Trillium, which on the surface *looked like* an example of Participatory Design in that users were directly involved in the development of the technology. Our analysis leads us to conclude, however, that Trillium's development departed in critical ways from our current model of Participatory Design and to suggest that the manner in which users are involved in the development effort plays an important role in the success of the endeavor.

**KEYWORDS:** systems development, technology in use.

## INTRODUCTION

There is a growing interest among system engineers, product designers, and human interface designers in developing ways of involving users in the design and evolution of computer-based systems. User involvement is seen as critical both because users are the experts in the work practices supported by these technologies and because users ultimately will be the ones creating new practices in response to new technologies. This interest has been sparked, in part, by a growing awareness of and appreciation for an approach to systems design pioneered in Scandinavia. This approach, frequently referred to as Participatory Design, advocates the active involvement of

users throughout the design process (Bjerknes and Bratteteig 1987, Bodker et al 1987, Ehn and Kyng 1987, Ehn 1988, Floyd 1987, Floyd et al 1989). A few years ago, the two of us were involved in helping to evolve a computer-based design tool called Trillium<sup>1</sup> which has been used to successfully design often complex user interfaces to Xerox products. At first glance, it appeared that the experiences we had with Trillium resembled the Participatory Design approach in that users were directly involved in the development and evolution of Trillium. However, upon further reflection we came to appreciate the ways in which our experiences with Trillium departed in critical ways from the model of Participatory Design we now understand. In this paper we ask the question, "Why was the design and development of Trillium *not* an example of Participatory Design?" Our answers to this question suggest that *how* users are involved in the design of new technology is critical to the success of the endeavor. Before discussing our experiences with Trillium, we will briefly describe the Participatory Design approach as we understand it.

---

\*Current address is Xerox Palo Alto Research Center, 3333 Coyote Hill Road, Palo Alto, CA 94304, 415-494-4322.

<sup>1</sup>Our relation to the development of Trillium is as researchers interested in the use of technology. Blomberg is an anthropologist who studied the introduction of Trillium into the user interface design community. The study involved participant observation at most Trillium clinics and core meetings, open-ended interviews and informal discussions with members of the Trillium community, and monitoring electronic mail exchanges. Most of the face-to-face interactions were audio taped for later analysis. We draw on these materials in this paper. Henderson is a computer scientist who, while at Xerox PARC, initially conceived of Trillium and was active in its development and use within the user interface design community. Both authors were present at all the meetings discussed.

## PARTICIPATORY DESIGN: DESIGNING WITH THE USER.<sup>2</sup>

There is no single view of what Participatory Design involves, nor is there a unified position on the theoretical underpinnings of the approach. For a detailed discussion of the similarities and differences in the theoretical assumptions, practices, and experiences of those working in this area see Floyd et. al (1989). However, a few basic tenets guide most practitioners of the approach which we outline here.

### The Goal is to Improve the Quality of Work Life.

The design and development of new technology is done in the service of improving the quality of work life for the users of the technology. Both the users and developers involved in the design activity share this objective. The focus of their efforts is not solely or even primarily on the design of the technology. Improving the quality of work life may involve rearranging the furniture, reallocating work tasks, or creating flexible hours. The technology might allow for these other changes or it might require them, but the focus is on the work as a whole and on the technology only as a component of that whole. There is a general commitment to carefully balancing the desire to preserve tradition against the opportunities provided by the technology to transcend old practices in order to attain new heights. As Pelle Ehn (1988: 162) has stated, "*design should be understood as a concerned social and historical activity in which artifacts and their use are anticipated; an activity and form of knowledge that is both planned and creative and that deals with the contradiction between tradition and transcendence.*"

### The Orientation Is Collaborative.

Developers and users work together to design and develop the technology and to integrate it into current work practices. From the start and throughout the project, the focus and goals are actively negotiated and not assumed or imposed by one group. It is recognized that the knowledge and skills of both the developers and users will be required for successful technology design. It is assumed that the technical expertise of developers and the work expertise of the users are equally critical to the success of the project. Establishing mutual respect for, and some understanding of, one another's competencies is understood as crucial to the work of designing new technology, both to support real communication and also to provide rich perspectives on all aspects of the design. It requires that through reflection and experience users become familiar with the technology and developers with the work. To this end, much of the interaction between developers and users takes place in the users' work environment.

<sup>2</sup> Notice the dual sense of "with the user": 1. the design is about the user's work as much as the technology, and 2. users are included in the design team.

## The Process Is Iterative

There is an appreciation for the fact that design should be an iterative process where emerging design ideas are tried out in real work situations. This requires developing ways of presenting nascent design ideas to users in such a way that they begin to get a sense for what it would be like to use such a technology in their work. Mock-ups, scenarios of use, and prototypes are employed to provide the opportunity for the users to "experience" the new technology and for developers to "experience" the new work practice. (Ehn and Kyng n.d., Ehn et al n.d., Kensing n.d.)

## THE TRILLIUM EXPERIENCE: DESIGNING WITH THE USER?

Trillium is a computer-based design tool used to design and implement user interfaces for copiers and printers (see Blomberg 1987, 1988 and Henderson 1983, 1986 for more information about Trillium and its use). When Trillium was first introduced into the user interface design community, it was in an early stage of development and relatively untested. As designers began to use Trillium, there was every expectation that Trillium's initial design and implementation would require some degree of modification as experience with use and utility was gained. Trillium also was designed with the expectation that it would evolve in response to the design requirements of the interfaces being built. As such, it had a mechanism for creating new design abstractions (referred to as "itemtypes") that could be added to a growing collection. These new abstractions might be implemented by the user interface (UI) designers using Trillium or, if programming skills were required, by supporters.<sup>3</sup>

In many respects Trillium evolved in relation to the experiences of its users. In the Trillium case, as UI designers began using the tool and confronting new design challenges, they became aware of ways they would like Trillium to change. The various design teams pushed the technology in different directions due to requirements of the user interfaces they were designing and their overall role in the product development process. On the basis of the needs of designers, "bugs" were fixed, new itemtypes were added, and enhancements to Trillium's basic functionality were made. In general, Trillium developed through the active involvement of its users.

<sup>3</sup> In the literature on participatory design, the two groups in question commonly are referred to as the "developers" and the "users". In the Trillium case, the developers were called "supporters" and the users, who were designers of user interfaces, were called "designers". Despite the danger of confusion in referring to the users as designers, in this paper we use the terms of the Trillium community, not only to honor their practice, but also to remind us that the designers do not think of themselves primarily as those who use Trillium, but rather as those who *design* user interfaces.

With this much user involvement, Trillium's development might be taken for a case of Participatory Design. However, a closer look shows that this was not the case. To help illustrate this, we will make reference to the following excerpts from a transcript of a meeting between designers and supporters (See Figure 1). This meeting, like many between designers and supporters, occurred at one of the periodic Trillium clinics. These clinics, which took place at approximately six-month intervals and generally lasted three or four days, were held to help coordinate some of the activities of the supporters and designers. Clinics provided an opportunity for supporters and designers to meet and talk about desired

changes and additions to Trillium, to develop possible means of implementing such modifications, and to parcel out work assignments. Designers and supporters also met informally outside these clinics when problems were encountered that prevented designers from moving forward with their designs. In addition, electronic mail was used to exchange information about bug fixes, itemtypes, availability of enhancement, and other the activities of the two groups. In the exchange transcribed in Figure 1, designers are reviewing a list of problems they've been having and are making some suggestions about possible ways of improving the situation.

**Designer (Kelly):** One of the problems that ( ) I don't know if I'm supposed to be changing the topic or not ( ) it has to do with changing menus, uh mostly has to do with itemtype classes [ahh] I've discussed it with several people so far, ( ) it seems to be one of the hardest things with the Birthday release ((a version of Trillium)) is the way the itemtypes are classed right now. [OK] The six primitive classes that come up are totally meaningless to designers and ( ) [OK]

...

**Designer (Nicole):** Because sometimes, the thing that's troubling to me is that the two things that I think are related, with basically the same name but maybe one's a little more specific, it's under a different name. And I didn't know what the names meant in the first place and then, when I found one, I thought now I could find the other, and it's not there. [yea] that's really ( )

**Supporter (Michael):** That sounds terrible. Well ( ) ah, we can step all the way out and ask, when it's about to ask for the name of an itemtype, what would you like to see? There's graph itemtypes, which shows you all of them, and what's related to what, and that's probably not what you want to see. And you could have, you could have it, you know, the thing that's there is sort of using the piece menus, which breaks it down into the fact of saying, look ( ) there's this collection of objects and you could have each of them in one or more categories, and then what you do is get to anything roughly two levels, by getting the right category and getting the right thing, which will allow you, like you say, if I'm in the right category, I'll find things that are like you. Which might be helpful. Ahh. Is it that that structure is not right for itemtypes or is it that we just don't have the right set of categories? Namely, we chose the one that had to do with the kinds of items, which was obviously not of interest.

**Designer (Kelly):** Well, we didn't seem to have much of a problem with Halloween. ((an earlier version of Trillium)) Mary and I pretty much came up with a scheme for classifying the itemtypes. [Aha] It seemed to make sense to most of the people around here ( ) [And] And at that point we had the ( ) they were given the opportunity to change the itemtype classes and no one ever changed them [uhhuh] everyone seemed to find them suitable to what they were doing. Now I'm getting requests to make it ( ) to turn it back to the way it was.

**Supporter (Michael):** OK, Now it ( ) it ( ) but it's not ( ) I mean Halloween was different in that it didn't have piece menus too ( ) I mean it had a different kind of menu.

**Designer (Kelly):** One thing could only occur in one menu.

**Supporter (Michael):** No. That wasn't true.

**Supporter (Janet):** You could have more than one. We just didn't do it that way.

**Supporter (Michael):** You could have it in lots of things. But it had a different kind of menu. And I ( ) I changed piece menus because they were somewhat better, I thought. And there may be some disagreement about that.

...

**Designer (Nicole):** Now, the main menu things were easier to understand. Now all these new categories appeared that the names themselves, it wasn't self explanatory what they were, or if I thought they were something, you know, like displayers. Ok, it sounds like it displays something. So it would be where I'd expect to find itemtypes that display things. When I go there sometimes a certain itemtype that I would think was a displayer, displays something, is not there. [yep] So now that is confu ( ) it's obviously an example of my naivete.

**Supporter (Michael):** No, it's not actually. It's a matter of our using the word displayer in a highly technical way which is not the same way that you would, as a designer, use it, and therefore, there's just a mismatch.

**Designer (Nicole):** Oh, OK.

**Figure 1: Transcription of Designer-Supporter Interaction<sup>4</sup>**

<sup>4</sup>No attempt was made to transcribe patterns of intonation. Utterances enclosed by brackets [ ] denote comments that were interjected into the speech of another. Untimed intervals in the stream of talk are indicated by parentheses ( ). To save space, two short sequences in this exchange were omitted from the transcript. These omissions are indicated by ... between the turns of other speakers. Italicized text within double parentheses (( )) represents the authors' comments provided for clarification.

### **The Goal Is to Improve the Quality of Work Life?**

While it might be argued that the supporters and designers agreed philosophically that their goal was to improve the work situation for designers, in fact, this goal was narrowed in two important ways. First, the focus was reduced to developing a tool that designers could use to build better user interfaces. Second, the emphasis was on improving the tool so that the existing mechanisms worked better for the particular activities of its current use without much examination of what that use was. The exchange transcribed above reflects this twofold narrowing of focus. The primary question that concerned both designers and supporters in this exchange was the best way to categorize and display the itemtypes used in the design of an interface. There was little exploration of how itemtype menus were used in practice, but instead concern centered on making the current situation better.

As a result of this narrowed goal, collaboration between designers and supporters focused almost exclusively on the technology and not on how to integrate the technology into designers' work practices. Designers placed the highest priority on securing a design tool that was easy to use and that provided the functionality needed to implement their developing designs. Supporters, on the other hand, were concerned with fulfilling their responsibility, as they understood it, to support the users of Trillium, and if possible to work on interesting programming problems and enhance their professional skills.

Consider the alternative. What would a broader focus have looked like? Note that in this interchange there is little exploration of how designers make use of itemtypes in their design practice. Instead the focus is on fixing the technology. It is likely that even if designers had been asked, it would have been difficult for them to describe their work practices in the abstract. It is significant that this exchange took place at a meeting where, ostensibly, such matters could be discussed. It was widely held among those involved with Trillium that meetings to discuss designers' experiences with the technology were important. However, because such meetings were removed from the situation of use, discussion frequently shifted to talk about the details of particular implementation decisions. The situation might have been improved if the discussion had taken place in the designers' work environment where supporters could have observed use of the itemtype menus in relation to designers' work activities and where designers would have had experiential access to their own work practices. Indeed, when for other reasons, supporters were required to sit with designers and watch their work, wholly new understandings of the problems of Trillium and its use readily emerged.

As for addressing how Trillium could improve the quality of the designers' work life, designers had expectations which were never of much interest to the supporters. Designers were hopeful that Trillium might improve their work situation by giving them more control over the

design and implementation of user interfaces for future products. The hope was that product development teams would be more likely to listen to the designers when it came to decisions about user interface characteristics if designers were able to demonstrate their design concepts on a prototype user interface. Trillium gave them the potential for doing just that. The designers also felt that there would be less room for the applications programmers, who implemented the interface on the machine under development, to misunderstand their design intentions if a "functioning" user interface was available for these programmers to emulate (Blomberg, 1988). Thus, the major qualities of work life that designers hoped Trillium might change were ones of authority, influence, and control. However, these issues were, for the most part, outside the realm of concern for the joint designer-supporter interactions.

### **The Orientation Is Collaborative?**

While on the surface designers and supporters were collaborating to get the job done, there were a number of factors which influenced the character of their collaboration both in concept and in practice. While not all of these factors can or should be changed, they must be recognized and dealt with if a strong collaboration orientation is to be fostered.

*Different criteria for evaluating success.* Designers and supporters did not share the same concerns, in that what might constitute a successful outcome for one group might not be perceived as such by the other. For example, a designer's request for an enhancement might yield a technically elegant solution, clearly a successful outcome for the involved supporter. On the other hand, the designers, unable to appreciate and perhaps uninterested in the technical elegance of the solution, might be focused instead on their difficulty in using the new enhancement. Although they were working together to improve Trillium, what constituted positive steps in that direction were different for the two groups.

*Character of designer-supporter interaction.* Most of the interactions between designers and supporters took place away from the designers' work environment, either at periodic Trillium clinics or at smaller project team meetings. Many of the early clinic meetings were characterized by supporters talking among themselves about the technical details of possible implementation strategies. These discussions rarely dealt with how implementation decisions might effect the way the designers were using the tool. Most attempts to steer the discussion back to the issue of tool use were unsuccessful. The realization that technical issues were dominating these discussions resulted in separate meetings for designers and supporters. Under this regime, designers met to talk about how they were using Trillium, problems they were having, and possible ways of modifying Trillium, while supporters met to talk about the latest fixes and enhancements they'd been working on and to assess the degree to which they had satisfied earlier designer requests. At later joint meetings,

designers would provide supporters with a list of desired changes or additions to Trillium. While the aim of this reorganization was to prevent technical issues from dominating all discussions, the result was to create a barrier between the designers and the supporters. The supporters were not present when designers talked about challenges they were having as they attempted to integrate Trillium into their design practice, and the designers were not as likely to learn about technical constraints that limited the supporters' options.

One result of supporters having little access to the day-to-day realities of Trillium's use was that they relied, not on seeing use, but on talking about it. Success on meeting the real needs therefore became contingent upon the ability of designers to clearly specify what their requirements were. Supporters expected designers to provide them with design specifications for new itemtypes or enhancements. An often-heard refrain from supporters was, "If you just tell us what you want, we'll implement it for you." However, designers were only able to provide very general descriptions of the functionality they wanted, which either left the supporters with little idea about how to satisfy the designer's request or little guidance on which of the possible implementation strategies to choose. The following quote is typical of such requests. "The designers would like a way to find an appropriate itemtype for a specific use. It was generally agreed that many of the itemtypes are badly named and that there is no easy way to find out what an itemtype does or is good for." There are many ways this functionality might have been implemented, and the choice would influence how the tool supported the work practices of designers.

*Organizational relations.* Another factor that effected the character of the collaborations between designers and supporters was the organizational separation between the two groups. At most of the sites using Trillium, the designers and supporters belonged to different organizational units within the corporation and as such reported to different managers. Each group had its own recognized areas of expertise and there was some concern for protecting the authority each had in their own domain. The designers did not want supporters making decisions they considered within their purview and the same was true of the supporters. This resulted in efforts to clearly define areas of responsibility vis-a-vis one other.

*Failure to appreciate and understand one another's competencies.* Because the focus was on the technology, the expertise of the supporters played a more important role in the interactions between supporters and designers. As mentioned earlier and as demonstrated in the exchange quoted above, most joint designer-supporter interactions quickly developed into discussions about fixes to the technology. In this environment where technical issues dominated, many designers were hostile to the idea that they should learn about technical issues effecting design and implementation decisions. From their point of view, their job was to design user interfaces, not build computer

tools to support that endeavor. Even those designers with no such aversion were "protected" by supporters from "technical concerns" because there was little recognition of the value of designers having some input into technical implementation decisions. For example, at one of the joint meetings, a supporter described her work on an enhancement this way: "The database business ( ) just quickly for the supporters. The designers don't need to know this too much. The way it finds things, whenever you store an object out, you specify what file it goes to. Then, after that, you don't have to worry about it anymore ..." This statement also illustrates that the supporters were the ones deciding what technical information was relevant to designers and these assessments were made in the absence of intimate exposure to designers' work practices. While some designers acquired a high degree of technical know-how in the course of using and helping to evolve Trillium, the knowledge they gained was directed at working around technical limitations and the failures of Trillium's mechanisms, and not at possible alternative strategies for implementing modifications or enhancements to Trillium.

When designers complained that Trillium required that they "think" and work in ways that were not traditional for them, the tendency was to see this as a designers' problem. The attitude seemed to be that if you wanted to use a powerful design tool like Trillium, you would have to change the way you thought and worked. While this may have been true to some extent, this belief, coupled with the narrowed focus on the technology, undermined the legitimacy of the designers' complaints and a commitment to developing a shared understanding of the trade-offs inherent in incorporating a computer-based design tool into the designers' work practices. These trade-offs necessarily dealt with larger issues of design practice which were never viewed as central to the designer-supporter collaboration.

In general, supporters had little knowledge of designers' work practices and their traditional practice did not involve engaging in activities that might provide them with such knowledge (e.g. spending time in the users' work environment observing current work practices). From the supporters point of view, the designers were in the best position to know what their needs were and the issue, for supporters, was getting designers to clearly specify those needs. The problem was often characterized as the inability of designers to clearly specify their requirements in terms meaningful to the supporters. To help alleviate this problem, at one of the clinics supporters suggested that designers be taught how to put their requests in terms easily understood by the supporters. The issue of a common language for collaboration also arises in the exchange quoted above where supporters had used technical terms to categorize itemtypes. These terms carried very different meanings for the designers.

The difficulty designers and supporters had communicating with one another directly related to the fact that the designers and supporters generally had different educational

and disciplinary backgrounds; industrial designers and computer programmers, respectively. These disciplines provided different expertise, perspectives, interests and practices. Consequently, in the absence of activities that would have cultivated an appreciation and understanding of one another's competencies, there was limited common ground for collaboration.

### **The Process is Iterative?**

As with the other tenets of Participatory Design, an iterative process seemed to be exactly what was intended when work on Trillium began. While there was iteration in building the first prototypes, Trillium's later development was not characterized by the tight experimentation needed to produce a tool which was truly responsive to the designers' needs. Here again, the influences we now discuss are not ones which make iteration impossible, but rather are ones which tend to make it more difficult.

*No grounding in use.* We have already described how development focused on Trillium's technology rather than on the use of that technology or on its effect on work life. This narrowing of focus made it possible to avoid considering both the richness and the indeterminacies of the full use situation. Instead one had to rely on the belief that the improvements desired in Trillium could be correctly envisioned and described from the outset. As a consequence, there would be little need to iterate on the development of these improvements. Furthermore, because the focus was not on use, occasions for observing use were not part of the pattern of development. This resulted in little opportunity for supporters to observe difficulties which their "solutions" might be creating. Under these circumstances it was unlikely that the need for iteration would be anticipated or discovered.

*Corporate operating style.* In an attempt to reduce miscommunication between cooperating groups, the corporation had espoused the notion that workers should regard these collaborations as customer-supplier relationships. Under this view, the supporters became suppliers of Trillium improvements to the designers, their customers. This view carried with it the idea that the customer-supplier relationship would be best served by clear specifications of the product supplied. While this view explicitly admits an iterative approach to adjusting and refining such specifications, in fact, such refinement could be regarded as more managerially risky. Not only is it hard to plan when there's some uncertainty about what the product ultimately will be, but it is difficult to know when you've satisfied your responsibility vis-a-vis your customer. Exacerbating this situation was the fact that at many Trillium sites, supporters and designers belonged to different organizational units. This meant that the "product" supplied by the supporters was being paid for by the designers, which increased pressure for clarity in contracting for service. The call for specifications, therefore, became a strong pressure for being clear "up front", and for not adjusting the specifications once implementation was in progress. As a result, designers

rarely were consulted until after an enhancement was completely implemented. By then, significant time and personal effort had been invested in the result, creating a difficult circumstance for questioning the quality of the product.

Given this dynamic of intergroup relations, the corporate style was also one which tended to regard any need for change as an indication of failure either of the creation of the specifications or of the implementation of the product. This strongly legislated against iterating once the enhancement was finished. Difficulties tended to become the source of assignment of fault, which did not produce a climate in which iteration was encouraged. Designers often had to simply accept and live with the supporters' solutions to their problems.

*Inability to evaluate.* Because designers lacked a sense of competence in the supporters' domain, they often felt unable to evaluate the work of the supporters. Even though a change or enhancement might be difficult for them to use or might not provide all the functionality they desired, for all they knew, this might well be the best possible solution. This posture is reflected in the above transcript where the designer Nicole, in effect, attributes the problem to her own naivete.

*History of working relationship.* Another pressure working against iteration was the conservative force of history. Once it had become "well known" that considerable effort was required to get changes made, and that when they were made the changes were unlikely to be exactly what was needed, the pressures to get the specifications right from the start increased. Second chances for specifying the desired changes also were curtailed because resources for support were limited, and iterating one solution had to be traded off against addressing other problems. As we have seen, this increased focus on correct specifications was highly counter-productive for the achievement of satisfactory changes.

*Work life situation.* Notably absent from the interactions between designers and supporters was any opportunity to sit down together and "try out" the different implementations. This lack of ongoing input to the implementation decisions being made by the supporters was recognized as a problem by the designers. At one clinic, the designers specifically requested that they be consulted, "Designers would like to be involved in the process of the supporters, defining the problem, and determining the solution to designer requests..." During this meeting the names of designers who would be interested and willing to work with supporters on particular requests were provided. However, little contact resulted; at the next clinic, six months later, only one designer had been consulted.

Operating without feedback, or at least without feedback that they respected, supporters tended to work on problems that were either interesting to them or ones which had

relatively straight-forward solutions. As a result, designers started to prioritize their requests, hoping that changes they believed to be of greatest importance would be implemented first. This strategy met with only limited success. Many of the high priority requests remained uncompleted at future clinics. The reason given was that either the supporters had not understood what was requested or they did not appreciate the importance of the request.

*Techniques for envisioning.* There was little use made of more powerful techniques for communicating between supporters and designers. In particular, techniques for refining the specifications and defining solutions in advance of implementation (e.g. envisioning, mocking up, simulating, prototyping) were untried. Talk and text-based description were relied on almost entirely. Ironically, the central tenet of Trillium itself – that fast prototyping would improve solutions in the design of other machines – was not generally applied to Trillium itself.

## CONCLUSION

We have characterized the Participatory Design approach as advocating three tenets which influence the character of the interaction between developers and users of computer-based systems: 1) the goal should be the improvement of the quality of work life of the users; 2) the orientation should be toward collaborative development; and 3) the process should be iterative. In this paper we examined a process (the development of Trillium, a tool for user interface designers), which on the surface *looked like* an example of Participatory Design, and concluded that Trillium's development did not conform to these tenets. The Trillium experience suggests that the achievement of these tenets has broad entailments which are related to how users are involved in the design of new technology and how the development effort is situated within the larger organizational context.

## ACKNOWLEDGMENTS

We would like to acknowledge the Xerox Corporation for their support of the development of Trillium, the Trillium community, and the research which led to this paper. We also would like to thank the members of the Trillium community who for nearly a decade have been the willing contributors to this effort in technology development, and collaborators in our research. Our description is of necessity a gloss on the realities that they know. We recognize the sincerity of their efforts at working together in developing Trillium, and their openness in sharing their experiences with us.

## REFERENCES

- Bjerknes, G. and Bratteteig, T. Florence in Wonderland: Systems Development with Nurses, In *Computers and Democracy*, G. Bjerknes, P. Ehn, and M. Knyg (Eds.) Avebury, England, 1987, pp. 279-296.
- Blomberg, J. L. Social Interaction and Office Communication: Effects on User's Evaluation of New Technologies, In *Technology and the Transformation of White Collar Work*, R. Kraut (Ed.) Lawrence Erlbaum Associates Pub., Hillsdale, New Jersey, 1987, pp. 195-210.
- Blomberg, J. L. The Variable Impact of Computer Technologies on the Organization of Work Activities, In *Computer-Supported Cooperative Work: A Book of Reading*, I. Greif (Ed.) Morgan Kaufman Pub., San Mateo, California, 1988, pp. 771-782.
- Bodker, S., Ehn, P., Kammergaard, J., Kyng, M. and Sundblad, Y. A UTOPIAN Experience: On Design of Powerful Computer-based Tools for Skilled Graphic Workers, In *Computers and Democracy* G. Bjerknes, P. Ehn, and M. Knyg (Eds.). Avebury Pub. England, 1987, pp. 251-278.
- Ehn, P. and Kyng, M. The Collective Resource Approach to Systems Design, In *Computers and Democracy* G. Bjerknes, P. Ehn, and M. Knyg (Eds.). Avebury Pub. England, 1987, pp. 17-57.
- Ehn, P. *Work-Oriented Design of Computer Artifacts*. Almquist and Wiksell International Pub., Stockholm, Sweden, 1988.
- Ehn, P. and Kyng, M. Design by Doing, In *Design as Action: Experiencing the Future* J. Greenbaum and M. Kyng (Eds.). Lawrence Earlbaum Pub., Hillsdale, New Jersey, (forthcoming).
- Ehn, P., Molleryd, B. and Sjogren, D. Playing in Reality, In *Design as Action: Experiencing the Future* J. Greenbaum and M. Kyng (Eds.). Lawrence Earlbaum Pub., Hillsdale, New Jersey, (forthcoming).
- Floyd, C. Outline of a Paradigm Change in Software Engineering, In *Computers and Democracy* G. Bjerknes, P. Ehn, and M. Kyng (Eds.). Avebury Pub. England, 1987, pp. 191-210.
- Floyd, C., Mehl, W., Reisin, F., Schmidt, G. and Wolf, G. Out of Scandinavia: Alternative Software Design and Development in Scandinavia, 1989, manuscript in possession of the author.
- Floyd, C., Mehl, W., Reisin, F. and Schmidt, G. STEPS to Software Development with Users, In *Proceeding of ESEC 1989: September 11-15, University of Warwick, Coventry, England*.
- Henderson, D. A. Trillium: A Knowledge-based Design Environment for Control/Display Interfaces, (videotape presentation), *CHI '83 Human Factors in Computing Systems*, Boston, Massachusetts, 1983.
- Henderson, D. A. The Trillium User Interface Design Environment, In *Proceedings CHI '86 Human Factors in Computing Systems*, M. Mantei and P. Orbeton (Eds.) April 13-17, 1986, pp. 221-227.
- Kensing, F. Generating Visions: Future Workshops and Metaphorical Design, In *Design as Action: Experiencing the Future*, J. Greenbaum and M. Kyng (Eds.), Lawrence Earlbaum Pub., Hillsdale, New Jersey, 1989.