# Technical Authorship: Refiguring the Designer-User Conflict and the Visioning of Collective Technical Futures

Marisa Leavitt Cohn
Department of Informatics
University of California, Irvine
Irvine, CA 92697-3440
+1 949 824 4047
mlcohn@ics.uci.edu

Susan Elliott Sim
Department of Informatics
University of California, Irvine
Irvine, CA 92697-3440
+1 949 823 2373
ses@ics.uci.edu

Kavita Philip
Department of Women's Studies
University of California, Irvine
Irvine, CA 92697-3440
+1 949 824 7092
kphilip@uci.edu

## ABSTRACT

This paper argues for the application of authorship analysis to technology design. It extends techniques used in Science Studies to investigate scientific authorship in order to define a concept of technical authorship. To illustrate the potential of this approach, authorship analysis is applied to particular prescriptive software design methodologies including Participatory Design and Agile Software Development. The results of this analysis are the recognition that: a) design methodologies are rhetorics of authorship, b) the designer-user relationship can be seen as a conflict of contested authorship, and c) this conflict can be seen within the tensions in the figuration of the user and the designer along a subject-object continuum. "Figuration," a technique used by Donna Haraway, brings about a new understanding of the centrality of the user in the design process as well as the role of the designer as the contested author technical futures. The implications suggest that design researchers might use authorship as a new approach to the politics of design by re-figuring the designer and user rather than collapsing the distinction between these roles.

## Topics

Information organization
Information technology and services

## Keywords

Authorship, Science Studies, Design Methodologies, Figuration, Politics of Design.

## 1. INTRODUCING AUTHORSHIP

Authorship is a central subject of study in cultural and literary studies and has received special attention in Science Studies. Michel Foucault first drew attention to the social construction of the concept of authorship, its historical origins and the functions that it serves in his essay "What is an author?" (1977) [9]. He points to the fact that there is both an empirical individual author, and an "author-function" that requires the construction of a figure of an author "who is outside and precedes" the authored text. The author-function and its figure arose as a new form during the 18th century conditioned by a growing culture of private property, [5 (author citing Foucault)].

Donna Haraway and others in Science Studies have developed this approach further to analyze the scientific author, its origins and implications [3, 7, 17, 19]. The scientific author is the authoritative voice of objective reasoning, that produces facts out of the experimental practices of the sciences. Latour [22], Shapin [29], Haraway [15], and others have helped to explain the origins of how this particular authorial voice came to be as well as it how it continues to be taken for granted. Latour [22] especially focuses on how the everyday practices of the laboratory aid in the social construction of facts, thus unveiling the scientific author (empirical individual) of his garb of scientific objectivity.

Haraway's focus shifts away from unmasking the scientist and towards the emergent author-figure who she calls the "modest witness" (borrowing the term from Shapin and Schaffer, *Leviathan and the Air-Pump,* 1985). The "modest witness" is "the witness whose accounts mirror reality" through "self-invisibility." This modest voice is that which "pays off its practitioners in the coin of epistemological and social power." It is "the virtue that guarantees that the [scientist] is the legitimate and authorized ventriloquist for the object world, adding nothing from his mere opinions..." [15].

This scientific authorial voice, which extends to much of academic writing in the social sciences and technical fields as well, is one that puts the objects of study in supposedly plain view while the author as a subject disappears from the text. The "modest witness" is used by the scientific author (the empirical individual) to offset his role as a subject. Authorship in the sciences appears subject-less. It is a world of objects that speak for themselves. This lack of a subject we now take for granted as that which lends authority to the scientific text, but this authority had to be shaped through the creation of the figure of the "modest witness" who steps aside, and out of view.

In this paper we will extend analysis of authorship to the field of technology design relying primarily on the techniques used by Donna Haraway in her book: *Modest_Witness@Second_Millenium.FemaleMan©_Meets_OncoMouse™*. One of Haraway's techniques is "figuration" which she

**Copyright and Disclaimer Information**

The copyright of this document remains with the authors and/or their institutions. By submitting their papers to the *i*Schools Conference 2008 web site, the authors hereby grant a non-exclusive license for the *i*Schools to post and disseminate their papers on its web site and any other electronic media. Contact the authors directly for any use outside of downloading and referencing this paper. Neither the *i*Schools nor any of its associated universities endorse this work. The authors are solely responsible for their paper's content. Our thanks to the Association for Computing Machinery for permission to adapt and use their template for the *i*Schools 2008 Conference.

uses to draw up this character of the "modest witness." Haraway's concept of the figure is one who "collects up the people; a figure embodies shared meanings in stories that inhabit their audiences." Figuration is difficult to explain but we are familiar with figures, in particular the idea of the "Christ-figure" where certain emblematic features of a character come together through the many metaphoric uses of it in different contexts or stories.

The figure in technology design that readily emerges through authorship analysis is that of the "user." The user is central to technology, computational systems, and product design. In sociological studies of technology design, however, the word has been taken up as highly problematic [13]. It is flagged in various debates within technology design as in the words "user-friendly," "user-centered," and "user experience" (aka UX). It is critiqued, ironically, for its lack of utility since it provides no context, characteristics, or sense of expertise. In fact "UX" responds to the last challenge by establishing "experience" as the user's area of expertise. "User" limits agency on the receiving end of technology to a single action: use, and so has been re-conceived as actor, knowledge-maker, or participant. "User" can provide only a generic sense of what is user-friendly or usable. The design researcher's response to that might be to use methods of "contextual inquiry" or "situated actions" [30]. Grudin has pointed out that the word "end-user" is a retronym that tells us more about the history of the man-machine relationship than about the real empirical user.

Yet, there is an apparent difficulty in dispensing with the word entirely. Authorship is a lens that can help to explain this persistence of the user-figure as problematic rather than trying to explain the particular problems it causes. Haraway's techniques for authorship analysis, using figuration, suggest that what is most interesting about the user is the way it is "figured" through technology design.

The other figure that emerges from authorship analysis of technology design is that of the designer. It is not strange that we as designers and design researchers understand this figure far less than we do the user-figure. Unlike the modest witness, the designer-figure is still very much in flux much as the scientific author-figure was in the 18th century. The many competing design methodologies point to an unclear author-figure and even an ambivalence towards the designer as author.

In fact, authorship analysis suggests that the politics of technology design exist in a conflict between the designer and the user that is one of contested authorship. Most often is the designers themselves that contest their own role as author and initiate methods to involve the users as authors in the design process. But there are also cases where software developers or engineers with to empower themselves as authors of the code in cases where they feel they have become instrumentalized by a design methodology.

## 2. MOTIVATIONS

There are a number of motivations for applying authorship analysis to technology design. First, there are many narrative techniques used in various design methodologies such as the use of "personas" [6, 11, 12, 16, 18], "scenarios" [4], "user stories" [1], and "storyboarding" [8, 24] in software systems development. These methods have been shown to have communicative power in the exchange between designers of technology and their client or user group, but these remain isolated methods that have not been theorized about as a whole.

Second, in technology design it is openly acknowledged that approaches to design can be metaphoric. Software development is likened to engineering or architecture and design methodologies such as 'waterfall,' 'agile,' and 'spiral' each make use of a metaphor whether rigorously or no.

Metaphors both reveal and conceal, and are limited in that they cannot provide an exhaustive framework for the evaluation or validation of best practices of design. Still, as young disciplines, technology design and design studies can benefit from a greater diversity of metaphors. Rather than encouraging a winner-takes-all atmosphere among competing methodologies, we should take the time to investigate many metaphors for design, especially since the field is still a contested domain with allegiances that cross disciplinary boundaries of art, science, and social science. As am emergent field seeking growth and foundational knowledge, we must be wary of eschewing practices that do not fit existing paradigms. Authorship as a lens for analysis, not only reveals new information about older design practices, but also defines a new space for understanding emerging practices that have thus far been set aside by academia as counter-productive.

Third, technical authorship is a metaphor that opens the door to collaborative authoring, that is, the creation of co-authored technical futures. With many technology design studies scholars interested in the politics of design, authorship can shed new light on the power struggles within design and suggest new paths to empowerment. Additionally, Haraway's method of figuration causes a different sort of reflection on design practice than other approaches to the politics of design. Figuration allows us to think about how we might want to shape (or "refigure") the role of the user and designer instead of resisting this dichotomy. (Haraway suggests a method of "diffraction" for refiguration, which we will not go into in this paper.)

And finally, there are many parallels between Science Studies and Technology Studies that are only beginning to be explored. In this paper we will be drawing some parallels (that at times are a bit shaky) between technical authorship and scientific authorship with the hope that future work that takes up this approach might lead us to a better understanding of the relationship between science and technology.

## 3. TECHNICAL AUTHORSHIP

The concept of authorship in the sciences, which was introduced above, is that of an author who presents facts about the natural world he investigates. But authorship of technology seems to differ from scientific authorship in that it is less a particular voice found in the texts designers produce than the very material shape of their technologies. In Science Studies, however, much attention has been drawn to the way that texts of all kinds, from reports, to talks, to notebook scribbles [28] and the scientific instruments play a role in authoring 'science' [15]. In design it is often 'use' itself that is authored, where the technological product, the design methodology, and even the users themselves, provide the assemblage that shapes 'use.' So the material nature of authoring technology is not a major issue.

However, in order to extend authorship analysis to technology design the idea of technical authorship still needs to be introduced in order to clarify how authorship applies to design of technology. Technical authorship is a way to look at technology not as only an artifact that is constructed, but also as a vision of the future that is authored. It is a technical future that is authored through the practices of the designer, which can include any number of activities such as requirements engineering, prototyping, iterative development, and especially through its successful implementation and use.

Authorship is just one lens or metaphor through which to examine these design practices, but it is one that is relatively unexplored despite the presence of narrative techniques in various design methodologies. This paper will apply analysis of authorship to some design methodologies, in order to reveal just a sampling of the potential outcomes that this approach can bring. In particular, it is the opinion of this paper's authors, that certain practices in design that are often considered a-theoretical or a-methodological may be better elucidated through this alternate metaphor of technical authorship.

# 4. AUTHORSHIP ANALYSIS OF TECHNOLOGY DESIGN

## 4.1 A Note on Scope

It would be impossible to exhaust the potential for analyzing authorship in technology design in a single paper. Instead we will provide a single attempt to look at software design methodologies through authorship analysis as a way to demonstrate the powers of this approach. In our research lab's ongoing research we contend that design methodologies exist in two forms, prescriptive and empirical, and it is infrequent that a prescriptive method is followed precisely or that empirical design is ever a-methodical even when it does not follow a prescribed method [10].

In this paper we will attempt to investigate the rhetorics of authorship located in prescriptive design methodologies, but we also want to point out that there would be equal if not greater value in applying authorship analysis to investigate empirical design practices where a specific technical future (use/product) is authored. By approaching design methodologies as rhetorical devices we will also touch upon the authorial voice of those design researchers who are writing the design methodologies. You will notice in the quotations provided that it wanders a great deal. At times it is the scientific author, modest, presenting design as a scientific object of study. At other times it is a voice that resembles inspirational or self-help writing, leading the reader/designer through personal development to better design practice.

## 4.2 Design Methodologies as Rhetorics of Authorship

The first thing that becomes clear in looking at prescriptive design methodology literature is that nearly all design methodologies must address the relationship between a user and a designer (though there is some design literature which treats design much more as an artistic and craft approach and disregards the user and 'use' entirely). However, while there are many design approaches that explicitly address the divide between the designer and the user, these do not view the conflict as one of contested authorship. Rather, the divide between the designer's "vision" and the user's "experience" is most frequently addressed because of the urgent and abrupt way that the latter can rupture the former. Massive failures of systems are often cited as evidence that design should "involve the user." Through the lens of authorship it becomes clear that both technical expert (designer) and non-technical non-expert (the user) are envisioning the technology in different ways. The design process then appears as a negotiation or conflict over authorship, which in many cases does not truly resolve.

Many approaches to technical systems design seek to involve the user explicitly in the design process. Methods such as Participatory Design [14, 21, 25] seeks a political aim to de-instrumentalize the user who can often be the subject of power alliances between designers and clients. Participatory Design has revealed design as embedded within a politics of the workplace. Because of this it has focused primarily on the conflict between management and the worker and on combating the ways that technology can instrumentalize the user. User-centered design, on the other hand, has focused on the conflict as a research problem and involves the user for a clearer "understanding of user and task requirements," [31]. This approach may produce technology that does not instrumentalize the user in a Heideggerian sense, but does forge the user into the designer's instrument for understanding use.

In Agile Software Development and Extreme Programming the tension shifts in the other direction. The technical engineer is seen to have been disempowered in the workplace, unable to take pride in the craft of software design, and instrumentalized as the tool through which software is enacted. The "Agile Manifesto" [2] and books like "Extreme Programming Explained: Embrace Change," by Kent Beck [1] introduce a message to empower the programmer or software engineer through a kind of bottom-up, subversive self-management technique.

Through the lens of authorship both Participatory Design and Agile Methods can be seen as attempts at co-authorship, though this analysis will shift the metaphor away from that of democratic participation or involvement used by these design researchers. The impact of this shift will be a focus on the tension between author (subject) and authored (object). This turns out to be extremely useful in understanding emerging software design methodologies that have so far been under-theorized.

## 4.3 The Current State of (Contested) Authorship in Technology Design

A literature review was conducted on prescriptive software design methodologies including Participatory Design [14, 21, 25], contextual inquiry [16], “situated actions” [30] and the use of Personas [6, 11, 12, 16, 18], scenarios [4], and user stories [1]. Authorship analysis, and use of Haraway’s figuration technique, resulted in the recognition that authorship is currently contested in technology design. The results of this analysis are summarized here by illustrating how figuration can fall along a continuum of subject to object.

With the modest witness we see an authoritative author-figure who is at the same time made subject-less. In the rhetorics of technology design there is a tension where the user-figure can either be a subject or an object.

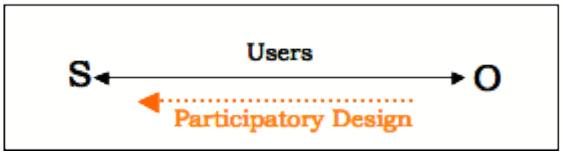

**Figure 1. The Subject-Object Continuum**

Personas are a technique used frequently in large-scale systems design. Companies conduct a research phase during which social science practitioners study the actual users and derive data in the form of archetypal characters. By the end of the research phase, users (empirical individuals) are transformed into objects; they are in a sense authored by the researcher. Participatory Design, on the other hand, explicitly critiques these kinds of methods, which reify the user, and seeks to re-engage with the user as a subject. This kind of movement can be illustrated as one along a continuum of the user as subject or object (Figure 1). The user can never be fully realized as a subject, since the user is a "figure" in the rhetoric of design and so belongs to its narrative.

Technology design rhetorics reveal a corresponding tension within the "designer" figure. Many design methodologies outline techniques by which the designer can arrive at an objective understanding of "use." Contextual inquiry warns that this is not entirely possible and focuses on the agency of the user in defining use. Still, even these techniques are used to help the designer witness and provide an account of a kind of objective "use" [16]. Less worried about the reification of the user, personas and scenarios are said to help the designer step out of and transcend his role in as if on a hero's journey to discover true "use." Even when expressly conscious of the reification of the "user," as in Participatory Design, the designer figure himself presents a significant problem to the rhetoric because he embodies the power-knowledge of the technical language.

Authorship analysis suggests that we add the designer into the Subject-Object Continuum as pictured below (Figure 2).

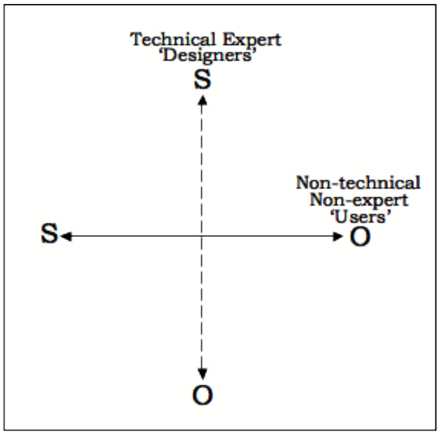

**Figure 2. Figuration of the Designer and the User**

This reveals that the designer can also be reified in the rhetoric of technical authorship. Coders can be seen as the instrument for obtaining technological artifacts as much as users can be seen as the instrument for obtaining objective "use." This new dimensionality provides a space for understanding practices such as Agile Software development, which has otherwise been seen to contribute little to the field.

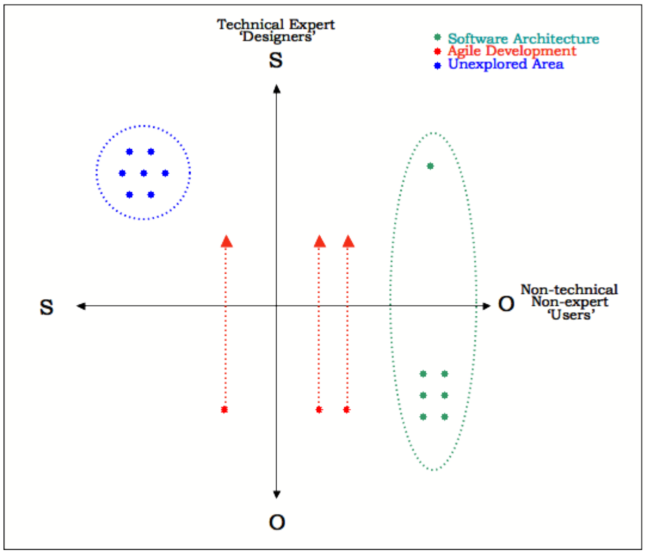

**Figure 3. Figuration in Different Software Methods**

Agile can be seen here (Figure 3) as a political movement to engage the software developer as subject instead of object. The division of those involved in design into the categories of technical expert and non-technical non-expert demonstrates that the user-figure and designer-figure can be located in diverse relationships within this subject-object continuum. Different movements are illustrated by locating methodologies within this continuum. Furthermore we note the relatively unexplored are where both designers and users are situated as subjects.

These tensions between subject and object and the current state of affairs in the rhetorics of technical authorship suggest that the role of the designer as author is contingent on the figuration of the user and likewise that the role of the user as author is contingent on the figuration of the designer. Before we attempt to address these tensions politically through projects to explore the space where both designer and user are subjects, we should first aim to understand how these two are figured in design methods and practices.

## 4.4 Figuration of the user and designer

We will now turn to the rhetorics of authorship found in design methodology literature and begin to shape the figure of the user and the designer that we found through our literature search. When viewed as methodologies, each approach may be seen to align with or critique another approach. But with the technique of figuration, what we look for is the emergent figure of the user, not something generalized from the literature. "A figure collects up the people; a figure embodies shared meanings in stories that inhabit their audiences" [15]. So, our task then is to gather up, from the various narratives within design methodologies, the specifics of the figure.

The user-figure that emerges is unwieldy, difficult to communicate with, and repellent. In the everyday practice of software engineers it is a "common sense" notion that engineers do not want to speak with actual users, that they are incapable of communicating with users, and that users are irksome in their language and demands. Latour [22] refers to the "fierceness" of disputes in the early stages before a fact becomes a fact. There is a similar fierceness in the early stages of design when requirements are not fleshed out, and this antagonism arises between the user and the designer.

The user-figure is a point of access to a naturalized “use.” Designers who advocate for user-centered design, or participatory design, will criticize projects that do not involve the "real user" in the design process saying that the wants and needs of "real users" can never be known without speaking directly to users.

In large-scale projects designers will often be speak to users, without such political motivations, as a pragmatic way to understand the technology’s future use. In this case a researcher will often serve as a proxy to study, interview, and observe users and communicate “use” back to the designer. In this case the user-figure is a way to ward off false use. He is a test for the purity of use.

This purity is set against the impurity of the designer, as a solution to the issue of communication. The trouble with the engineer is that he is "bad at" talking to users, and there is also the issue of scale when the user group and the company hired to design technology may both be too large to interact in a personal way.

The designer-figure must have access to the user-figure because "Getting it right" matters, particularly in large projects where massive failures can arise. There are many invaluable contributions of research in this area highlighting context, cognitive models, and cultures of the workplace. This literature notes that technology must work hand in hand not only with use, but also with context, cognition, culture, training, and management. Two major disciplinary outcomes have been the idea of "co-evolution" [27] and "situated actions" [30]. The first concept tells us that local, contextual practices will shape how a technology is adopted and technology will always change those same practices. Context, cognitive models, and cultures *a priori* will never satisfy these conditions - there must be some support for a back-and-forth either of prototyping or studying use at various stages.

"Situated actions" highlights two aspects - context and action. This refigures the user as an actor who makes use of technology in many ways based on situations in which he acts. It implies the autonomy and agency of the user, but only in a rhetorical way. The situations and the actions are still seen as knowledge, which can be obtained in a modestly objective way through a discipline of studying the context of the technology in search for situations and actions rather than uses. Not only that, it subjugates the designer to the technology as well, by marking the agency and autonomy of the designer as immodest. It flips the hierarchy, placing the actor-users as autonomous and above technology, (the technology itself should be as objective as possible), and the designer-user as a modest witness to the user-technology relationship. It sets up an ideal vision of a design process that develops technology for a single-person, single-situation, in which the voice of the designer is muted and the designer is the instrument for obtaining the technology.

Participatory Design makes explicit that the user is a user-subject of the technology-employer alliance. Technology should align itself instead with the user-worker. This acknowledges the knowledge-power of technology itself if it is authored by management. This is a first step in recognizing the question of who should author a system. But it does not acknowledge the power-knowledge of the developer that still exists in Participatory design, i.e. the possession of the technical languages, including *code*.

The first, and most crucial reflection is that the user must be figured into the design process. Even when the designer responds to a call to empower the user or support the user's actions, design of technology always imagines use and figures a "user." Often the rhetoric of Participatory Design and other design methods that are conscious of the conflict, struggles with this tension. Some generalizable use is maintained and this upsets the idealized relationship between the designer and user. Whether the design process refigures the user as a "participant" or an "actor" the word "user" fails to fall away.

We would like to consider the "user" as a figure because it enables us to see the design process as a narrative in which the user is the primary figure. An alternative response then is not to remove the user, but to refigure or "mutate" the user but continue to see the user as central to the narrative. We can see how Participatory Design and other user-centered projects are attempts to refigure the user. Often it seems that the new processes have been achieved but that the refiguring of the user has in some senses failed.

"A figure collects up the people; a figure embodies shared meanings in stories that inhabit their audiences" [15]. If we understand current approaches to technology design as narratives we can already see that the user is a figure that collects up the people. The user collects up the selection of people who are seen to be the receivers of technology. While designers may select particular users to involve in the design process, the user-figure clearly points to the collection of all people who will be users, which will inevitably include the designers themselves. The user-figure embodies shared meanings of use within the contested space of what users and designers make of the technology being designed.

In technology design, the designer has legitimate authority over the object world of designed objects, but this authority does not follow the same "modesty" of the scientific author. The designer, no matter the efforts to empower users, is one who intervenes through the technology s/he creates. The designer is at times expressive through the medium of technological innovation. At other times, especially in much of design methodology, the focus shifts to the user and the designer as a subject recedes from view. However, in what the designer knows and in how s/he intervenes, s/he is immodest. And in the modes through which the designer tries to achieve objectivity (e.g. through a focus on the user), s/he is immodest.

Using the lens of authorship we can further analyze this immodest role of the designer whose voice remains authorial even if there are moves to limit that voice. The immodesty begins to make sense once we realize that design's authorial voice has not yet stabilized in the way that the scientific author has. And designers as a whole are ambivalent, or take conflicting views, on the designer's role as an author, whether to be expressive or "self-invisible." And attempts to place the designer in the self-invisible mode of the modest witness are ultimately unsuccessful because of the technical skill required to author the technology to which, in most cases, only the designer has access.

Literature on Participatory Design and scenarios both consider the need to get software and design professionals to step out of their role and into a new role. This is one way in which the user is clearly a figure, a role that can be taken on by the design professional who chooses to step out of his role. "Can software professionals recognize and affirm the validity of perspectives other than their own, and value the expertise that comes from experience, not just the knowledge that is attested by academic credentials?" [25]. This acknowledges a kind of common sense notion that software professionals need to be disciplined to seek outside perspectives. In *Five Reasons for Scenario-Based Design*, Carroll suggests that scenarios aid in reflection. "Reflection is not always comfortable; it forces one to consider one's own competence, to open oneself to the possibility of being wrong" [4]. This provides a sense that reflection is possible but that reflection of a different kind does not come easily and must be fashioned through discipline and techniques.

In Carroll's guide to using scenarios, the designer is figured as a kind of hero - "technical professionals are intelligent people performing complex and open-ended tasks. They want to reflect on their activities, and they routinely do reflect on their activities," but they require an extra push to surmount that higher peak, to reflect on experiences of the non-technical non-professional user. The hero-designer must pander to user experience, user expertise about "experience," and must transcend his role. The user is seen as having expertise about his own experience in both Participatory Design and other user-centered design. Expertise of experience is a substitute word for subjectivity and still poses the designer as residing in the world of objectivity but needing to reflect on the subjective world in which his work will ultimately reside.

Techniques to step out of the role of designer or to communicate with "real users" are often narrative in form. There are scenarios, which attempt to provide plain English accounts of use in a chronological ordering like a story, or more accurately like one cohesive scene in a story. Scenarios are most often written by developers after meeting with a customer or user and are then provided back to the user or customer for verification or even as a contract for work. "User stories" in Extreme Programming (another software process) are also used as small units of work that are agreed upon in an iterative weekly meeting and which describe features in the language of the user [1].

The discomfort of reflection in the design process is noted and is moderated in the techniques of scenarios and personas, by a mutation of the user into something that the developer has authored and created. The user as a persona or within the plot of a scenario is generic and though this figure can make the designer uncomfortable he is ultimately a phantom of the designer's own mind and thus the designer remains in control. The designer is described as being brought into an "intimate" engagement with the "concrete elements of the situation," the situation that belongs to the user and is obtained by a reflection on the experience of the user. An intimacy with the person who will use the system appears too dangerous to request of the designer. The subjectivity of the user is again removed. The intimacy that the designer should aim to achieve is with "concrete elements of the situation," i.e. something objective about the subjectivity. The designer requires something objective to take away in order to design.

This is especially seen in the use of "personas" in the design process. Personas are a literary technique first adopted in marketing and then in product design and ultimately in large-scale software design projects.

Grudin [12] says that personas can be used to make up for some shortcomings of user-centered design and scenarios, particularly in large-scale projects. Often these projects require a specialist group (e.g. academic researchers) to spend time with users, to study, interview and observe them. But Grudin critiques that the data often meant to center design around real users fails to engage the designers.

A solution is personas. "Personas are fictional people. They have names, likenesses, clothes, occupations, families, friends, pets, possessions, and so forth. They have age, gender, ethnicity, educational achievement, and socioeconomic status. They have life stories, goals and tasks" [12]. Designers can author personas without any basis on real users, but Grudin does not recommend this. In many cases personas are authored after extensive data collection about real users as a way to represent the data in a more engaging way.

Personas are seen to "take on a life of their own," or "take over" the design process. At the 2007 SIGCHI conference James Nieters of Cisco gave a report entitled "Making Personas Memorable" [26]. Cutout dolls of personas used by Cisco were provided to conference-goers as take-home souvenirs. Nieters said that one of the benefits of personas is that no one asks for the data that back them up.

These very deliberate figurations of the user into "personas" seem to resolve the conflict of broken communication between the user and designer. Personas speak to engineers in a way that real users cannot. But personas are also a way to elide the issue of authorship. The user is kept at a great distance from the design, and is in fact authored into a persona before having a voice in the design process. The designer thus behaves as a ventriloquist for the user as object-world. The persona that the designer authored is inverted in the hierarchy, such that the user-figure is in control of the design but is not a threat to authorship.

Haraway says that the modest witness "is the legitimate and authorized ventriloquist for the object world, adding nothing from his mere opinions, from his biasing embodiment" [15]. The designer fits the role of an immodest witness because he is the "authorized ventriloquist" for the user world, but his opinions and biases are what figure him as the author of the designed technology. This immodesty is a paradoxical dilemma. Unlike the modest witness, the designer is incapable of "self-invisibility" [15]. He must step out of the way for the user to speak the needs of the system, he must ventriloquize this speech modestly, but then he must become visible in the same moment through his authorship of the technology that justifies the design process.

In Participatory Design there is a clear reframing of software design as a political process that involves a conflict of expertise and as a politics of knowledge-creation. It also frames participation in the design process as "not mere 'involvement'" [25]. Muller also acknowledges that participation in the design process is an issue of communication "through language or through artifacts." However, Muller also creates a "Taxonomy of Participatory Design Practice" that poses a dual axis of practice based how early or late the user participates in the design process and how much the user participates in the world of the designer/how much the designer participates in the world of the user.

What is maintained in all these rhetorics is the underlying and under-acknowledged desire of the designer to create "new ways of doing things and new things to do" [4]. In distancing the designer from his desire, design methods create a tension between relinquishing control through in order to understand the real situation of the users and retaining an authorial vision, i.e. the authorship of the design. But this sets up a conflict that cannot be resolved. How can a designer create new ways of doing things and new things to do if they are accountable to the current ways that users do things and the current things that users do.

The role of the designer can be seen as one of a certain expertise in conflict with a different expertise of the user. But there is more to the role of the designer than this kind of expertise. Ultimately designers are good at envisioning use, at imagining and bringing into being new uses and new ways to use. While the framing of design as embedded within economic and political projects allows the designer to consider the obligation to empower the user as a worker-subject, it does not allow us to consider the conflict of authorship. One can involve the user in the design process at any stage and can migrate the designer to the world of the user or the user to the world of the designer but the authorship of the technology remains an unresolved conflict.

## 5. IMPLICATIONS: TRANSLATION OR REFIGURATION

This reframing of technology design as Technical Authorship leads to the breakdown of the designer into the roles of expert, author, and especially translator between the technical and non-technical or popular languages. It lays out a different set of obligations and accountabilities for the designer of technology. Not only can we work to make the "user" and the "designer" subjects in design practice, technical experts could succeed in bridging the gap between designers and users by defining a new obligation to translate technical languages and promote technical literacy.

However, Haraway also warns us that "there is no way to rationality - to actually existing worlds - outside stories, not for our species anyway." That is, we should be wary of de-figuring as a route to co-authorship. Within science studies she suggests that we should not unmask the modest witness. Storytelling is the only way that we know about the natural world and so the figure of the modest witness affords us a place to begin telling stories about the world. If we strip the empirical individual of his figures, how will he continue his practice? All the more so, we should be wary of tossing out the immodest witness or user-figure. If storytelling is the only way we can understand the natural world, as Haraway contends, then it is most certainly the only way that we can come to conceive of future worlds we wish to create.

Re-figuring the immodest witness is important in order for reflective and critical design practices to continue to emerge and grow. And the user-figure should certainly not be dealt with rashly because he is design's primary figure that distinguishes design from science. If the sciences had a figure outside of Nature herself to contend with, perhaps the sciences would be better off. That is not to say that the user-figure does not need re-shaping. But perhaps we should embrace the creative methods found in personas and the "user stories" of agile software development as ways to refigure and reshape design.

By reframing the conflict of the user-designer relationship as one of authorship we find a way that the user and the designer can forge a different relationship. This aim aligns with projects for user-centered design and participatory design that seek a non-instrumentalized user-subject but also departs from these through the consideration of the user-figure as central and indispensable to narratives of technical futures. Through the mutation of the user-figure, the qualities that divide the designer-figure from the user-figure fall into several roles that are commonly conflated: that of the author, expert, and translator.

While the politics of the user has been highlighted in various rhetorics, these have failed to acknowledge the conflict as one of contested authorship. These projects that explicitly express concern for the user, nonetheless maintain authorial control of the technology in the hands of the designer. If the user is the primary figure of the authored narrative, then the user as user-subject stands in the way of authorship, while at the same time being the instrument for discovering an objective "use," and the one whose needs must be satisfied. This is an impossible position for the designer, who is lead into a figurative hero's journey.

By considering the conflict of the user-designer relationship as one of authorship, we are able to highlight the user as the primary figure of the authored work. Current design methodologies can be examined to understand this figure: the "user." This reframing provides the possibility of considering the co-authorship of technology not as the co-design of technology. It leads to the breakdown of the designer into the roles of expert, contested author, and translator between the technical and non-technical languages. It lays out a different set of obligations and accountabilities for the designer of technology.

Co-authorship of technical futures retain the user as the central figure but place the non-designer as a co-author of the narrative the user-figure inhabits including the authorship of a mutated user-figure. The designer plays the role of technical expert, but also translator. The reframing of the problem as one of authorship reminds us that expertise is not the only mode of authorship, that authorship is not the only means for the designer to play a role, and that translation is vital to the political project of building technical futures.

In all cases where the user is figured into the design process or is refigured as an actor, expert, knowledge-maker, meaning-maker, or participant, the attempt to avoid instrumentalization fails to some degree. This is because the user is a figure of technology design, of technology, of the technical and techne. The user is a figure, the focal character in a narrative of imagined future uses. No design process can fail to imagine future uses.

Practices of technology design that do not instrumentalize the user are seen in approaches where authorship is highlighted in the case of artistic or critical technology design where the user is a figure and not intended to inhabit a real person. It is also seen in the abandonment of the user when designers design for designers as is seen in the hackers and opensource movements. The opensource movement, as Kelty [20] discusses in his article on recursive publics, tends towards a mutation technology authorship. Copyleft and other opensource practices are anti-authorial.

Extreme Programming [1], which is an approach to software development, was fashioned as a set or practices to empower software developers in the face of too many demands and requirements from users. It is not often highlighted that this approach also takes a stance against authorship of code. A key principle is that the code belongs to everyone. Additionally stories are used to translate needs of users into test cases and test cases translate easily into code.

Grudin points out that personas are effective because they are "a technique that... can draw upon powerful psychological forces to restore these dimensions" of engagement, complexity, representation, and identity. In this, and his discussion of fiction, Grudin acknowledges what Haraway says that "there is no way to rationality-to actually existing worlds-outside stories, not for our species, anyway" [11]. But the success of personas and other kinds of narrative devices in the design process should not be used as a wedge between the user and the developer. Instead this should form a recognition that the user-figure is central and is always authored with or without data collection.

The user figure should not be put to sleep but should be "mutated" as Haraway suggests with the mutation of the sciences' modest witness. This mutation happens through the focus on technology design as technology authorship. Adrian Johns discusses how authorship in the sciences was crafted over time. The "modern authorial persona" was shaped through "shifts in policing, property, bibliographical classification, and, finally... understandings of the creative process" [19]. In design, the creative process is one of the primary narratives, and so it is not surprising that fictional methods have been deployed in design while they have been extricated from the scientific authorial voice.

Stories are already a part of design, but they are "fictional methods to convey profound truths" [19], when they ought to be fictional methods to convey profound fictions. Those who write about a critical technical practice in technology design such as Phoebe Sengers, draw attention these narratives of future use. But the user-figure is not central to these narratives as much as the technology which instantiates future uses that were previously unknown.

The user-figure is mutated from its original role because it is now distinct from the empirical users, is maintained as the central figure, is more radically subjective, but also intersubjective. The cat's cradle that Haraway discusses requires a back and forth between the designer and non-designer in shaping knots and paths that will make up the narrative of technical futures [15]. The co-authors are the players of the game and should not be tied up in its knots.

The user figure is the use-maker, to draw on the idea of knowledge-maker, he is the character in a story of rain-making, the one who brings about rain by inhabiting a story that imitates the rain and is the rain. The user-figure can be to the modest witness what the white rabbit is to Alice, or the fool to the magus, the one who is followed through the story to tell the story itself.

## 6. REFERENCES

[1] Beck, K. 2000. Extreme Programming Explained: Embrace Change. Addison-Wesley Professional.

[2] Beck, K. et al. 2001. "The Agile Manifesto" http://agilemanifesto.org/

[3] Biagioli, M. and Galison, P.2003. "Introduction." In Scientific Authorship: Credit and Intellectual Property in Science, M. Biagoli and P. Galison, Eds. Routledge, New York and London, 1-9.

[4] Carroll, J.M. 2000. Five reasons for scenario-based design. Interacting with Computers 13, 43-60.

[5] Chartier, R. 2003. Foucault's Chiasmus: Authorship between Science and Literature in the Seventeenth and Eighteenth Centuries. In Scientific Authorship: Credit and Intellectual Property in Science, M. Biagoli and P. Galison, Eds. Routledge, New York and London, (Chap 1) 13-31.

[6] Coney, M.B. and Chatfield, C.S. 1996. Rethinking the author-reader relationship in computer documentation. ACM SIGDOC Asterisk Journal of Computer Documentation 20, 23-29

[7] Elliott, M. and Scacchi, W. Mobilization of Software Developers: The Free Software Movement. Information, Technology and People.

[8] Erickson, T. 1996. Design as storytelling. interactions 3, 4 (Jul. 1996), 30-35. DOI= doi.acm.org—234813.234817

[9] Foucault, M. 1977. "What is an Author?" In Language, Counter-Memory, Practice. D. F. Bouchard and S. Simon, Trans. Ithaca, New York: Cornell University Press, 1977. pp.124-127.

[10] Gallardo-Valencia, R.E. and Sim, S.E. 2007. "Planning and Improvisation in Software Processes," ACM Crossroads, Volume 14. Issue 1.

[11] Grudin, J. 2003. The West Wing: Fiction Can Serve Politics. Scandinavian Journal of Information Systems 15, 73-77.

[12] Grudin, J. and Pruitt, J. 2002. Personas, Participatory Design and Product Development: An Infrastructure for Engagement. Proc.PDC 144-161.

[13] Grudin, J. 1993. Interface: an evolving concept. Commun. ACM 36, 4 (Apr. 1993), 110-119. DOI= doi.acm.org—255950.153585

[14] Gulliksen, J., Göransson, B., Boivie, I., Blomkvist, S., Persson, J. and Cajander, Å. 2003. Key principles for user-centred systems design. Behaviour & Information Technology 22, 397-409.

[15] Haraway, D.J. 1997. Modest-Witness@ second-Millennium. Femaleman-Meets-Oncomouse: Feminism and Technoscience. Routledge Chapman & Hall.

[16] Holtzblatt, K. 2002. Personas and contextual design. Retrieved August 17, 2003.

[17] Jackson, M. 2003. Can Artisans Be Scientific Authors? The Unique Case of Fraunhofer's Artisanal Optics and the German Republic of Letters. In Scientific Authorship: Credit and Intellectual Property in Science, M. Biagoli and P. Galison, Eds. Routledge, New York and London, (Chap 5) 113-131.

[18] Johansson, M. and Messeter, J. 2005. Present-ing the user: constructing the persona. Digital Creativity 16, 231-243.

[19] Johns, A. 2002. The Ambivalence of Authorship in Early Modern Natural Philosophy. In Scientific Authorship: Credit and Intellectual Property in Science, M. Biagoli and P. Galison, Eds. Routledge, UK, (Chap 3) 67-90.

[20] Kelty, C. 2005. Geeks, Social Imaginaries, and Recursive Publics. Cultural Anthropology 20, 185-214.

[21] Kyng, M. 1998. Users and computers: a contextual approach to design of computer artifacts. Scandinavian Journal of Information Systems 10, 7-43.

[22] Latour, B. 1994. Science in action: how to follow scientists and engineers through society. Harvard University Press, Cambridge, MA.

[23] Lessig, L. 1999. Code and Other Laws of Cyberspace (Code v2 Wiki!). Basic Books.

[24] Madsen, K. H. and Aiken, P. H. 1993. Experiences using cooperative interactive storyboard prototyping. Commun.

ACM 36, 6 (Jun. 1993), 57-64. DOI= doi.acm.org—153571.163268

[25] Muller, M. J. and Kuhn, S. 1993. Participatory design. Commun. ACM 36, 6 (Jun. 1993), 24-28. DOI= doi.acm.org—153571.255960

[26] Nieters, J. E., Ivaturi, S., and Ahmed, I. 2007. Making personas memorable. In CHI '07 Extended Abstracts on Human Factors in Computing Systems (San Jose, CA, USA, April 28 - May 03, 2007). CHI '07. ACM, New York, NY, 1817-1824. DOI= http://doi.acm.org/10.1145/1240866.1240905

[27] Rogers, Y. 1994. Exploring obstacles: integrating CSCW in evolving organisations. In Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work (Chapel Hill, North Carolina, United States, October 22 - 26, 1994). CSCW '94. ACM Press, New York, NY, 67-77. DOI= http://doi.acm.org/10.1145/192844.192875

[28] Rheinberger, H. 2003. "Discourses of Circumstance" A Note on the Author in Science. In Scientific Authorship: Credit and Intellectual Property in Science, M. Biagoli and P. Galison, Eds. Routledge, New York and London, (Chap 12) 310-323.

[29] Shapin, S. 1994. A Social History of Truth: Civility and Science in Seventeenth-century England. University of Chicago Press, Chicago.

[30] Suchman, L.A. 1987. Plans and Situated Actions: The Problem of Human-Machine Communication. Cambridge University Press,

[31] Vredenburg, K., Mao, J., Smith, P. W., and Carey, T. 2002. A survey of user-centered design practice. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Changing Our World, Changing Ourselves (Minneapolis, Minnesota, USA, April 20 - 25, 2002). CHI '02. ACM Press, New York, NY, 471-478. DOI= doi.acm.org—503376.503460