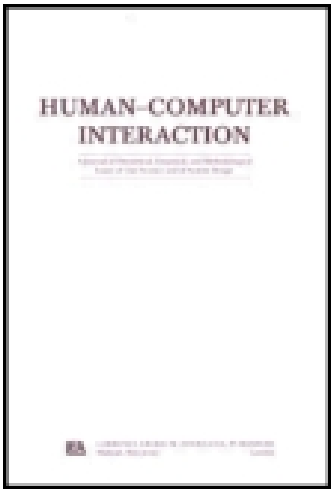


Human-Computer Interaction
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/hhci20

What Designers Talk About When They Talk About Context

Jared S. Bauer[a], Mark W. Newman[b] & Julie A. Kientz[a]
[a] University of Washington
[b] University of Michigan
Accepted author version posted online: 09 May 2014.Published online: 17 Jun 2014.

To cite this article: Jared S. Bauer, Mark W. Newman & Julie A. Kientz (2014) What Designers Talk About When They Talk About Context, Human-Computer Interaction, 29:5-6, 420-450, DOI: 10.1080/07370024.2014.896709

To link to this article: http://dx.doi.org/10.1080/07370024.2014.896709

HUMAN–COMPUTER INTERACTION, 2014, Volume 29, pp. 420–450

ISSN: 0737-0024 print / 1532-7051 online
DOI: 10.1080/07370024.2014.896709

What Designers Talk About When They Talk About Context

Jared S. Bauer,[1] Mark W. Newman,[2] Julie A. Kientz[1]
[1]University of Washington
[2]University of Michigan

Context has long been considered an important component of design, but as technology becomes more capable of inferring the user's behavior and environment, what constitutes context has become an increasingly pressing concern to designers. Although design frameworks and models have been proposed for context-aware computing systems, there has not yet been research that focuses on understanding context empirically from the perspective of the designer. To address this, we present an analysis of 11 in-depth interviews we conducted with designers of a variety of context-aware systems. Our analysis of the artifacts and interviews reveal five concerns designers address in their work. Furthermore, we present a process model that illustrates how context-aware system designers address these concerns. Our findings demonstrate the central role that designers' views of context plays in (a) framing a design space, (b) encoding the relevant features of context, (c) unifying possible solutions within that design space, and (d) evaluating designs. These findings suggest that context is a dynamic concept that evolves over the course of a design project, generally from a more phenomenological perspective toward a positivist interpretation. This, and the process by which it occurs, contributes insight into context-aware design with implications for both academics and practitioners.

Jared S. Bauer (jsbauer@uw.edu) is a researcher with an interest in design for ubiquitous and affective computing; he is a Ph.D. candidate in the Information School at the University of Washington. **Mark W. Newman** (mwnewman@umich.edu) is a researcher with an interest in ubiquitous computing and end-user programming; he is an Associate Professor in the School of Information at the University of Michigan. **Julie A. Kientz** (jkientz@uw.edu) is a researcher in human–computer interaction and ubiquitous computing with an interest in designing and evaluating applications for health and wellbeing; she is an Assistant Professor in the Department of Human Centered Design & Engineering at the University of Washington.

Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/hhci.

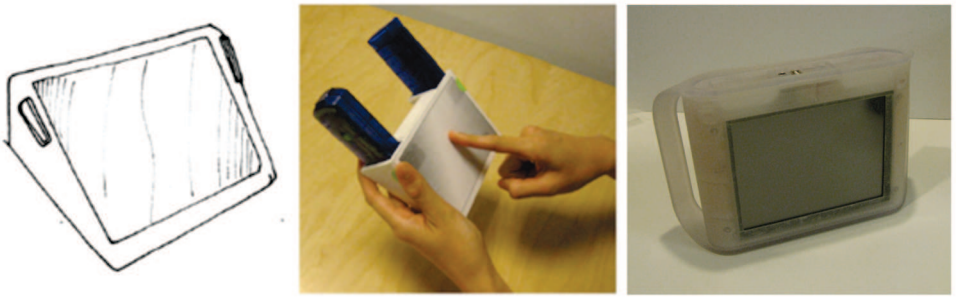
CONTENTS

1. INTRODUCTION

The nature of context and its role in the design and development of technology has been a topic of much debate in the field of human–computer interaction (HCI; Dey & Abowd, 2000; Dourish, 2004; Greenberg, 2001). Now, in an era where context-aware systems are regarded as commonplace (Abowd, 2012), context's role in design is no longer just a topic of debate for researchers—it is a daily concern for practitioners. The perspective of designers is of particular interest in the process of creating context-aware systems because of the role they play in determining how a technology will be situated in the world and therefore what constitutes the system's context. However, how the designers' understanding of context is reflected in their work is a topic of research that is conspicuously absent from the literature.

Although much has been written about the nature of context, much of the literature has focused on a pragmatic approach to what can be detected by computers (Dey, 2001; Schilit, Hilbert, & Trevor, 2002) or on a theoretical exploration of the nature of context (Dourish, 2004). Rather than explore the nature of context, in this article we investigate designers' views on context. Recent work has helped to clarify the value that taking a practitioner's perspective can provide to the larger research domain (Goodman & Wakkary, 2011; Stolterman, 2008). To take the practitioner's perspective means that we learn to understand context not as a neutral, objective phenomenon

FIGURE 1. Artifacts from Participant 4 that show the progression of the design from an initial sketch, to a low-fidelity prototype, and then to a functional prototype.



based on technology or on theoretical models but as a construct that reflects the views of designers. In this article, we aim to determine designers' understanding of context by examining the artifacts they create and methods they utilize in the creation of context-aware systems. Put differently, this article provides an examination of what designers talk about when they talk about context.

To achieve a better understanding of context-aware design practices, we conducted interviews with 11 designers of context-aware systems. During the interviews, designers provided us with the artifacts produced in the design of one context-aware system they had created (see Figure 1). They then walked us through the design process detailing the role of the artifacts and methods in which they engaged. This allowed us to follow the design process from initial concept to a finished product, all from the designers' perspectives.

This article contributes a detailed description of the design processes of context-aware systems from the designers' perspective. Our analysis revealed a set of five concerns designers needed to address in order to produce a coherent system. These concerns are *users*, *context*, *form*, *interaction*, and *implementation*. It further revealed a pattern of activity by which the designers' understanding of context influenced how they *framed* the design space, *filtered* the design space for possible solutions, *encoded* the contextual components of the system into a vocabulary, used this encoded vocabulary to *unify* a solution, and then *evaluated* the solution in terms of the codes. In addition, rather than acting as a static concept through this process, we found that the designers' concepts of context adapted in light of possible solutions and newfound constraints. This forced designers to revisit these stages multiple times to evaluate various solutions.

2. RELATED WORK

2.1. Perspectives on Context

Our work draws on empirical studies of designers and previous work attempting to characterize context. The title of this article is an intentional reference to Paul

Dourish's (2004) work investigating context and its role in design. Dourish's work outlines the value of viewing context from a phenomenological perspective in which context arises from our interaction in the world, drawing on everyday, cultural, common-sense understandings of the nature of the social world. He contrasted this perspective with positivist accounts of context where it is viewed as a set of attributes of the world that can be objectively observed and enumerated. The positivist view of context is represented by Schilit, Adams, and Want (1994), who defined context as "where you are, who you are with, and what resources are nearby" and "lighting, noise level, network connectivity, communication costs, communication bandwidth, and even the social situation" (p. 85). Dey (2001) expanded the idea of context to include "any information that can be used to characterize the situation of an entity" (p. 5). The purpose of the present article is not to discuss the merits of these positions. Each is valuable, but more important, they are formulated as academic positions on the nature of context and do not necessarily represent how context is viewed by practitioners designing context-aware systems. We acknowledge that these positions on the nature context represent the experience of HCI researchers designing and developing context-aware systems; however, this work contributes to prior work by providing empirical evidence to enrich the HCI literature on these positions.

The value of taking the designer's view has been an area of considerable interest as the community of HCI researchers seek to ensure that their work remains relevant to the larger community of practitioners (Goodman & Wakkary, 2011; Stolterman, 2008). Goodman and Wakkary (2011) noted that many HCI frameworks and theories have had limited impact on professional design practice and asserted that this disconnection reflects the inadequate attention paid to the complexity of design practices. Similarly, we hope to extend the research community's understanding of context by providing a detailed account of how context is viewed by practitioners.

2.2. Theoretical Lens

In our investigation of design practice, we draw on Schön's (1992) theory of *design worlds* to inform our understanding of the conceptual space in which design work is conducted. Schön argued that through designers' perceptions of actual or virtual worlds, they create the objects and relationships with which they interact and determine what exists in the design world. These design worlds are abstract spaces in which designers create and evaluate objects and relations as they work to create an optimal design. Rather than investigating particular objects in the designers' world, we explore how the designer creates the relationships among objects. Our contention is that the designer's formulation and representation of these relationships necessarily influences the type of technology that is suited to exist in that design world. This means that when a designer formulates how a new design will respond to context in their design world, it is based on an understanding of context that will guide how that design is able to relate to the world. As Schön noted, design worlds may be unique to the designer or shared across a community of practice. This suggests that explicating

these worlds, and the types of relationships inherent to them, could help to establish a common ground to reason about modes of context-dependent interaction.

To explicate these worlds, we draw on Charles Goodwin's (1994) theory of *professional vision*. Professional vision is the term Goodwin developed to describe the practices used by members of a profession to shape a domain of scrutiny. For example, an anthropologist and a farmer impose different meanings on to the same substance (or "domain"), for example, soil. Their analyses of the domain rely on different assumptions, methods of analysis, and systems of scrutiny. Goodwin argued that these practices create the knowledge that forms the theories, artifacts, and expertise that are distinctive to any professional domain. According to Goodwin, professional vision relies on three practices in any domain:

> (1) Coding, which transforms phenomena observed in a specific setting into the objects of knowledge that animate the discourse of a profession; (2) highlighting, which makes specific phenomena in a complex perceptual field salient by marking them in some fashion; and (3) producing and articulating material representations. (Goodwin, 1994, p. 606)

By analyzing the practices in the domain of design for context-aware systems, we can begin to understand the ways designers understand context as a domain of scrutiny.

2.3. Context Within Design

Our study is situated within a body of literature examining the role of context in design, with special attention paid to the practices of designers. The seminal work on Contextual Design by Beyer and Holtzblatt (1999) has argued for the importance of understanding the context of use for design and outlined methods for understanding context from the perspective of users. Although understanding the context of use is an important aspect of design, our work differs by focusing on systems that are proactively aware of what this context might be. Prior work has explored how Information Architecture (Morville & Rosenfeld, 2008) design methods and artifacts account for context (Bauer, Newman, & Kientz, 2014). Although this work is useful in characterizing how context is manifested in design for Information Architecture, the present article builds on it by providing a comprehensive account of the design process of context-aware systems. Methods such as Experience Prototyping (Buchenau & Suri, 2000) seek to enable designers or other stakeholders to engage with the imagined uses of a system, including understanding the role of context. Similarly, Davidoff, Lee, Dey, and Zimmerman (2007) presented the Speed Dating method, which they argued allows designers to rapidly explore application concepts and their interactions and contextual dimensions. Whereas each of these methods creates compelling ways to explore context in design work, they do not provide insight into designers' views on context.

Previous research has supported the design and development of context-aware applications, including application toolkits (Dey, Abowd, & Salber, 2001) and infrastructure support (Hong & Landay, 2001) to facilitate the rapid development of

context-aware applications. These efforts have produced a number of insights into the technical requirements for supporting context-awareness and the potential for easing the burden of development, but they have been primarily aimed at software developers who have a different set of concerns, practices, and skills than designers.

To support the design of context-aware systems, researchers have taken several approaches. Prior work has sought to use *design patterns* to support design for ubiquitous computing systems (Chung et al., 2004; Landay & Borriello, 2003). Although designers did find such an approach useful, the generated patterns were based on a review of the research literature instead of being informed by observed design practices. Dow, Saponas, Li, and Landay (2006) conducted a series of interviews with designers to investigate design practices for context-aware systems. This work revealed the importance of storytelling for depicting context in design. However, Dow's work largely focused on issues influencing the development of tools to support ubiquitous computing designers rather than the designers' understanding of context.

3. STUDY METHODS

We conducted 11 video-recorded interviews with designers who had worked on recent projects that gave "special consideration to the users' context," as quoted from our recruitment e-mail. We chose a wide framing of context to capture projects that designers themselves described as being context-driven. In addition, we discussed the projects with the designers prior to the interview to ensure they possessed characteristics that meet Schilit et al.'s (1994) definition of "context-aware" (e.g., software that "adapts according to the location of use, the collection of nearby people, hosts, and accessible devices, as well as to changes to such things over time," p. 85). Following prior studies of designers (Dow et al., 2006; Newman & Landay, 2000), we focused each interview on a single project on which the designer(s) worked in the recent past.

The interviews focused on design artifacts and methods, how the artifacts were used, and with whom they were used to communicate in the design process. We define an *artifact* as any tangible or digital document, device, object, or file that was created by the designer to help them in the design process (Blackwell, Whitley, Good, & Petre, 2001; Norman, 1991). This definition includes a variety of items produced by designers, such as wireframes, architecture diagrams, mockups, interactive prototypes, physical models, written documents, paper or dry erase board sketches, videos, and photographs. Artifacts were used as memory triggers and interview foci, but they were also reviewed closely in conjunction with the transcripts for the role they played in each specific design project. This allowed us to understand how the artifacts and the methods were applied to the concerns of each project. Prior to each interview, we asked designers to provide us with an inventory of artifacts produced during the selected project and provide copies, photographs, or scans of the artifacts wherever possible. Because our study was retrospective in nature, our request for the artifact

inventory included a number of prompts and examples of different types of artifacts to help designers recall a greater number of details regarding the project under study.

3.1. Participants and Recruitment Criteria

We recruited via messages posted to several interaction design mailing lists, through personal contacts working in industry, and by searching for publicly shared design projects or products that seemed appropriate for the study and directly contacting the designers. For our study, we defined designers as individuals who engaged in an iterative process to explore a problem space and produce a product or service that addresses the problem they explored. We view this approach as contrasting with scientific, artistic, or engineering approaches. Our view on the distinction between design and science or art draws on Cross's (1982, 2001) definition of "designerly" ways of knowing. Unlike art, Cross argued that design aims to provide a solution to a problem. Cross also argued that science focuses on solving a specific, well-formed problem, whereas designers address ill-defined problems. Similarly, we view engineering as setting out to produce a solution to a known problem rather than working to create a solution to an ill-defined problem. While recruiting, we sought designers that worked in either academic or industry settings.

Because our definition for designers was specific, once we had conducted 10 interviews, we decided to exclude two of the interviews from further analysis. We chose not to analyze these interviews because the process discussed focused too heavily on engineering a solution and therefore were not useful for understanding design practices. Because we chose to exclude these interviews, we returned to recruiting and conducted three additional interviews to ensure that we reached a point of data saturation (Bowen, 2008; Lincoln, 1985). Thus, this article presents findings from 11 interviews, which included a total of 14 designers, as two of the interviews (1 and 8) were conducted with multiple designers (see Figure 2).

3.2. Coding and Analysis

We conducted a two-stage analysis of the interviews. In the first stage, we conducted a content analysis of the interviews by coding each interview using the Text Analysis Markup System (TAMS) Analyzer software (http://tamsys.sourceforge.net/). To identify common concerns and themes in the interviews, we created a code list and then coded the transcripts of the interviews. To generate a list of codes, the first author watched portions of each interview and developed lists of potential codes. These codes were then reviewed to see how consistently they appeared in other interviews as themes. Members of the research team then watched an interview to check for consistency in the application of codes. This led us to refine and remove several codes and then reapply codes to the interviews. The final code set was influenced by Lim, Stolterman, and Tenenberg's (2008) notion of prototypes as filters in design work. According to Lim et al., when viewed as filters, design artifacts are used to

FIGURE 2. Participant experience as designers and the projects discussed in the interviews.

ID	Years as Designer	System Description	Platform
1A	1	Location-aware smartphone application for locating restaurants.	Mobile Phone, iOS
1B	2		
1C	0		
2	12	Wearable location and activity sensing smartphone application for logging physical exercise.	Windows mobile, custom sensor suite
3	10	Small tangible tile capable of recognizing gestures and proximity of other tiles.	Custom-built tangible interactive device
4	3	Suite of sensors and portable educational tool for high school science students.	Custom-built touch screen device with a suite of sensors
5	9	On-body activity and location sensing device and smartphone interface.	Custom hardware and mobile phones
6	5	Tangible device used to promote mindfulness of power consumption.	Custom tangible device implemented with micro-processors
7	4	Interactive TV and ambient interface for socializing through the television.	Custom hardware and commercial televisions
8A	4.5	Location aware smartphone game.	Mobile phone, iOS
8B	6		
9	10	Location-based desktop and smartphone app for healthy lifestyle recommendations.	Web Applications, iOS
10	5	Location-based smartphone application for managing time.	Mobile phone, iOS
11	4	Mobile application for sharing online shopping experiences.	Mobile phone, iOS

investigate, or filter, the qualities in which designers are interested, without distorting the understanding of the system as a whole. We expand the categories outlined by Lim et al. to account for some of the phenomena we felt these designers were "filtering." We added *context* and *implementation* to their initial list of filter categories to describe the phenomena that we felt the designers were trying to explore with their artifacts. This initial stage of coding allowed us to identify the concerns designers addressed in their work.

After identifying the salient concerns addressed by designers, we returned to the artifacts and interviews to explore how the designers' view of context was manifest in their artifacts and practices and how their view on context influenced their work. To explore these questions, we applied professional vision (Goodwin, 1994) as a theoretical lens to the design artifacts and the transcripts of the interviews. We looked for examples where designers attempted to discuss and account for context in the design of the system. Because the focus of our work is to understand context from the perspective of the designers, we attempted to remain neutral about what context could mean. Therefore, rather than analyzing the interviews by looking for examples of what we believed context to be based on a literature review or our personal

intuition, we instead looked for representations or practices that are not accounted for in conventional computing interfaces (Hutchins & Hollan, 1985) or that relied on implicit interaction (Schmidt, 2000). In addition, we looked for instances where the designer specifically discussed context.

4. FINDINGS

In this section we outline the findings from our analysis of the interviews and artifacts in two ways. First, we begin by detailing the concerns that designers addressed across the interviews, where we briefly discuss the concerns we identified. We then provide a description of a design process to demonstrate how designers addressed these concerns. Finally, we turn our attention to the patterns of activities that characterized how designers we interviewed moved through the design process and illustrate these with relevant examples.

4.1. Five Common Design Concerns

Our analysis of the interviews revealed that the designers addressed five common concerns. Although specific design questions were unique to each project, we found the following higher-level concerns to be useful in characterizing what aspects of the system the designers sought to understand. The five concerns we encountered were *users*, *context*, *form*, *interaction*, and *implementation* (see Figure 3). By identifying these concerns we aim to provide insight into the emphases that the designers are exploring through their artifacts and methods. We are not asserting that there is a "correct" amount of emphasis to be placed on any given concern, but it was clear that the extent to which these concerns were visited impacted the outcome of project. In

FIGURE 3. Common concerns that context-aware designers address with their design artifacts and practices.

Concern	Example Questions
Users	• What experience and knowledge do the intended users possess? • What are the intended users' goals, preferences, mental models, and current behaviors?
Context	• In what physical environments will the product be used? • In what concurrent activities will users be engaged at the time of use?
Form	• What physical form should the system take? • How large, heavy, rugged, attractive can/should the devices that make up the system be?
Interaction	• How will the system receive input from the users and the environment? • How will the system communicate with users?
Implementation	• What are the requirements for hardware and software in terms of capability, performance, power consumption, etc.? • What is it possible for the implementation to achieve in terms of these factors?

addition, providing a vocabulary of concerns helps to add clarity to the utility of various methods and practices.

Design Concerns in Practice: Interview 8 Background

To illustrate how the preceding concerns were represented in the practices of context-aware designers, we now detail the design process from Interview 8. This interview is a useful example for two reasons. First, their work resulted in a successful and well-received game, so we can call their process successful in some objective way. Second, this team had several years of prior design experience (see Figure 2) but no experience in context-aware design. This interview is helpful in illustrating the unique challenges that designing to account for context can create.

Interview 8 was conducted with two designers about their work developing a new game for smartphones that used the location of the users as a factor of the game (see Figure 4). The designers were part of a team of four master's students that had received funds to develop the game from a competition financed by a local entrepreneur. The funds paid for them to work on the project over the summer between their 1st and 2nd year as master's students. The team had four members, and although they all were involved in the design and development of the application, our interview was with the two who had focused primarily on its design.

Both of the designers had a background in web design. Designer 8A noted having 3 years of experience working on web design and front-end web development. His undergraduate education was in business and entrepreneurship, but he was eager to have the design and development skills to "start a project and get [it] running." Similarly 8B had a degree in Business Information Technology. Prior to returning to school for his master's degree, he had worked doing web design and development. He commented that his web design experience focused on "evaluation of potential [web] applications. And . . . adding features that might be easy to add via a bit of code."

FIGURE 4. The finished location aware game designed by the participants from Interview 8.

Despite having a background in web design, they felt that mobile, location-aware design was going to be an increasingly important area in design. They saw their work on the project as having value beyond the immediate product as it also gave them the chance to develop their professional skills and build their portfolios in a burgeoning design domain.

Winning the competition was viewed as rather prestigious, and previous winners had turned their ideas into successful startups. However, accepting the grant money was not without risks. Because their work on the project occurred during the summer between their 1st and 2nd years as master's students, it prevented them from having internships. Receiving this grant and forgoing internships meant their work on the project would represent a large portion of their professional portfolio as they entered the job market. The opportunity that the grant for this project afforded then was significant, but because they were working on a new topic with no mentors over a summer break, they knew there was the very real possibility that the project could fail. Also, because the technical domain was reasonably new, they were not certain what would or would not be feasible in the allotted time.

Another factor that impacted their work was the relationship of the team. The two designers interviewed for the study knew each other prior to their work on this project; they were classmates and part of the same social circle at school. However, they had only met the two developers they were working with in order to form a group with the purpose of entering the competition. Thus, the design and development portions of the group did not have a history of working together. This resulted in the members of the group having difficulty selling each other on their ideas. The group's entry into the funding competition proposed that they would build a location aware game for the iPhone, but other than that, they had decided on very few specifics. Because the nature of the game they intended to build was undecided, and they were newly acquainted, the power structure of the group was of particular importance.

It became apparent during our interview that they had no clear group leader and instead relied on achieving consensus when making decisions. Although decision making was diplomatic, both designers mentioned how challenging it was to persuade other group members to take particular directions with the game they were developing. In particular, they stated that they would nervously practice their pitches before group meetings and that meetings would be long and contentious. Although group members seemed to be generally respectful of each other, the lack of familiarity, and thus trust, required the designers to argue their case persuasively. This resulted in the team's work being delayed through much of the summer as they decided on a direction to take the game. At times, it became clear that the challenges imposed by the group dynamics influenced the ordering and creation of the artifacts in their design work.

Design Concerns in Practice: Interview 8 Overview

During their work, they created seven artifacts in all (see Figure 5). As per our interview, we move chronologically through their work using the artifacts as foci.

FIGURE 5. Artifacts and the concerns they addressed in Interview 8.

Concern	Artifact 1: Table Sketches	Artifact 2: User Research Findings	Artifact 3: Whiteboard Sketches	Artifact 4: Lo-Fi Wireframes	Artifact 5: Demo Application (Prototype)	Artifact 6: Whiteboard Sketches	Artifact 7: Final Prototype
User		●			●		●
Context	●	●		●	●		●
Form					●		
Interaction	●	●	●	●	●	●	●
Implementation	●		●	●	●		

This will help to establish the process by which these artifacts were created and what purpose they served. By doing so we can illustrate how the designer's addressed various concerns and how successful their artifacts and methods were in this process.

The designers began their work by sketching ideas on a glass table in their office (Artifact 1). During this stage, they iterated over a variety of ideas for games. Each of these ideas ultimately revolved around a set of location-dependent user interactions. This process was very influential in forming their thinking of how the user's location—or *context*—and *interaction* could be used as components of a game. They also spent much of their time discussing how they could *implement* a system that would respond to these two elements. When discussing this stage in their design process they made comments such as:

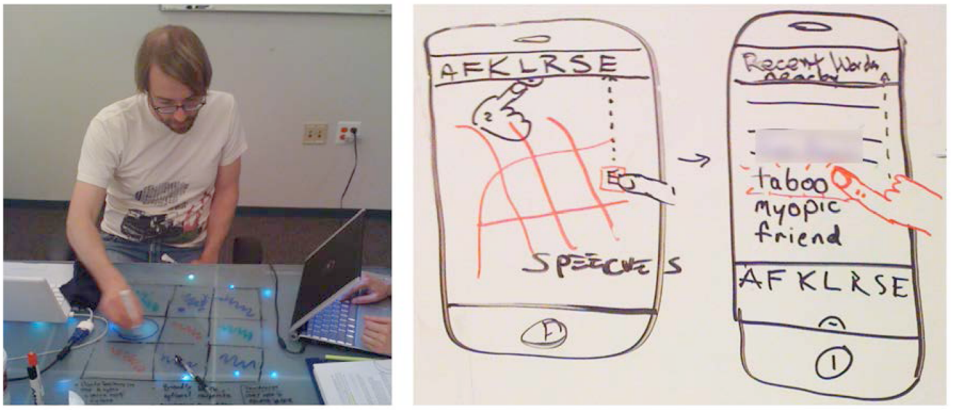
> Participant 8a: "So we knew we wanted to do a game—a game using the GPS capabilities of the iPhone that makes the physical world the game board. That was the kind of assumption that started all this reasoning. We had a few ideas in mind in the very early stages."

Once they had developed a few ideas for games, they conducted informal interviews with six or seven peers to get feedback. The designers kept a handwritten synopsis of the interviews on a white board in their offices throughout the design process and referenced it regularly (Artifact 2). These findings distilled their interviews into a set of high-level themes about potential *users*' attitudes toward *context* aware games and what forms of *interaction* in the game might be compelling. Both designers noted that these notes were regularly referenced during debates about the direction they should take with the game. This was part of the reason why they kept the notes on the dry erase board through the duration of the summer. The visibility of these notes was useful in keeping the direction of the project aligned with their findings. With these findings in mind, they began sketching versions of the interface on a whiteboard (Artifact 3). Externalizing their ideas as sketches gave them an opportunity to discuss how some of these games could be played and how they might implement their ideas, given that they were planning on developing an application for the iPhone. These sketches allowed them to understand how they could *implement* the desired *interaction* of the system.

With their idea for the system externalized in sketches, they developed more detailed wireframes of the interface to determine how the user's *interaction* and location would affect the game (Artifact 4). During this process, they again returned to the sketches on the glass table (Artifact 1) to visualize the role of location in the game (see Figure 6). The locations on the glass table were hypothetical but were used to understand how changes in location, or *context*, would impact the interface and thus the users' *interaction* with system and how they might *implement* the wireframe as a functional application. When commenting on this process one designer stated the following:

> Participant 8b: "[The glass table] is actually a little better than the white boards, 'cause you can kind of hover over it like you would a map; and given that we

FIGURE 6. Artifacts from Interview 8.



Note. **Left side: the glass table used for sketching location. Right side: whiteboard sketches of the interface.**

> were working on a map-based thing . . . at one point I think we even considered putting—like, we have a map that we considered putting underneath it just to kind reinforce the idea that whatever we're doing is based on the geography of the real world."

After they had completed the wireframes, they developed a demo application (Artifact 5) that they gave to a group of peers to try out the game during a street fair. The initial game mechanic was a type of scavenger hunt that the participants were asked to play as they wandered through the street fair. This form of prototyping allowed them to gain *user* feedback about the core *interaction* of the system in *context* running on the iPhone (and thus in the correct *form*). One of the interesting outcomes of this process was the realization that they had focused on the *implementation* too heavily, and this resulted in them having modes of interaction that did not necessarily result in a fun game.

> Participant 8a: "I mean, we have this cool technology, that you can pick up stuff from a map, you can drop stuff in a map based on your geolocation, you can pass stuff between users, there is user-to-user interaction going on, from iPhone to iPhone, cool, I send it to you . . . but what—we don't have . . . I guess the problem is where is the story, the compelling story, where is the narrative of the game, from that part, and we felt that we were not accomplishing anything."

This finding from their prototyping led them to re-evaluate and redesign the narrative of the game. This was complicated by the fact that the summer was coming to its end and they needed to produce a finished product. During a debate about which direction to take the game, Designer 8A introduced the idea of a scavenger

hunt game. The objective of the game was to collect pieces of an image scattered in the physical world and then reassemble the image. They hoped to monetize the game by allowing sponsors to suggest images, which could be assembled and returned for prizes. He explained the game by suggesting that a local bar could give a discount on a beer to people that collected and returned a reassembled image of a beer. As he spelled out the word *Beer* on the dry erase board, the group had an epiphany. They decided to create a word game similar to scrabble but with the letters out in the world waiting to be picked up. This idea was appealing because it was consistent with their initial goal of having a location-aware game but also aligned nicely with the user feedback they had gathered. On a more pragmatic note, this game could be developed without having to create complicated graphics, which had been a sticking point on prior ideas, as neither of the designers came from a graphic design background.

With an idea for the game finally settled, the remainder of the work for the project proceeded rapidly. Having a defined direction was helpful, but the rapidly approaching end of summer gave them additional motivation. They began redesigning the application by sketching ideas for the interface. This allowed them to determine the user's *interaction* with the system (Artifact 6). After several rounds of sketching, they finalized the idea in a higher fidelity prototype and conducted informal testing to evaluate what *users* thought about the *interaction* in *context* (Artifact 7).

Their finished application received primarily positive reviews on Apple's App Store and game review websites; it also had a large and sustained user base, so by many definitions their work was a success. However, to arrive at the final design required significant changes to the application after their first round of prototyping. The issue they encountered with the initial application did not arise from a lack of thought about the game play mechanics; during the course of the summer, they vetted numerous ideas and discussed games at length before deciding to create the scavenger hunt game. So, what caused the issues with their initial concept for the game? By looking at the concerns and the order they addressed them, it becomes clear that from the earliest stages in the design process, they were already concentrating on how they would implement the system. Three of their first four artifacts included some reflection on how they would implement their ideas. As a result, when they finally created a prototype to gain insight on the users' interaction with the system in context, they were dissatisfied with the result. It is hardly a new finding that focusing on implementation early in the design process can preclude possible design directions. However, it was not necessarily their focus on implementation that was the sole issue; we believe a more significant issue was that the designers did not produce artifacts that allowed them to simultaneously triangulate on the interplay between *user*, *context*, and *interaction* concerns. As a result, they focused on implementing the system to enable them to understand these three concerns in conjunction instead of creating artifacts (e.g., prototypes) to explore these concerns directly.

The difficulty in finding ways to explore the user's interaction with the system in context was a common theme across several other interviews. In fact, this concern was mentioned in five of the 11 interviews (Interviews 1, 6, 7, 8, and 10). In regards

to this challenge, the designers from Interview 1 noted the following:

> Participant 1a: We were developing things just to see what it would look like. It was part of the design in that it was just to fool around and prove to ourselves that we could actually make something. See what our limitations were . . .
>
> Participant 1b: "but also see how people could interact with it."

Although this concern was common, it was not uniformly distributed. We noticed that the more senior designers, particularly those with experience doing prior work on context-aware systems (e.g., Designers 2 and 5), produced artifacts that more evenly addressed these concerns. More importantly, they did so without having to implement functional prototypes. Because of this, we believe that designers learn to orient toward these concerns and gravitate toward practices that enable them to understand issues that arise at the intersection of these concerns. In the following section, we discuss in greater depth the processes designers employed to explore these concerns and the role their view of context played in shaping this process.

4.2. Accounting for Context in Design

In the previous section, we discussed the concerns that must be balanced while designing context-aware systems. With these concerns in mind, we turn our attention to the process by which designers investigate these concerns during their work. As we saw earlier in the work of the designers from Interview 8, balancing concerns is fundamental to this process, but how these concerns are balanced, and what role the designer's view on context plays in this process, has yet to be explored. In this section, we outline four practices that we observed designers engaging in during the design of a context-aware system. Specifically, we detail the practices designers followed to *frame* a design space, *encode* the relevant criteria against to create a context-aware system, *unify* possible solutions within that design space, and then *evaluate* possible solutions. Each of these practices is expanded upon next.

Framing the Design Space

Designers' views of context frame the space in which the system they are designing will exist. This process of framing begins when designers articulate what they hope to accomplish in their work and how they might accomplish it. In this sense, the process of framing a design space is largely constructive, but what it constructs is a space bound by the designer's view of context. Gaver's (2011) research on design workbooks suggests that design creates the spaces in which it operates. Our findings are aligned with this, but in addition to creating the space in which the design operates, we saw that designers' notion of context creates a way to frame that space such that certain notions of context are clearly contained, whereas others are not. We can see this process unfold by analyzing what is, and is not, represented in the designer's

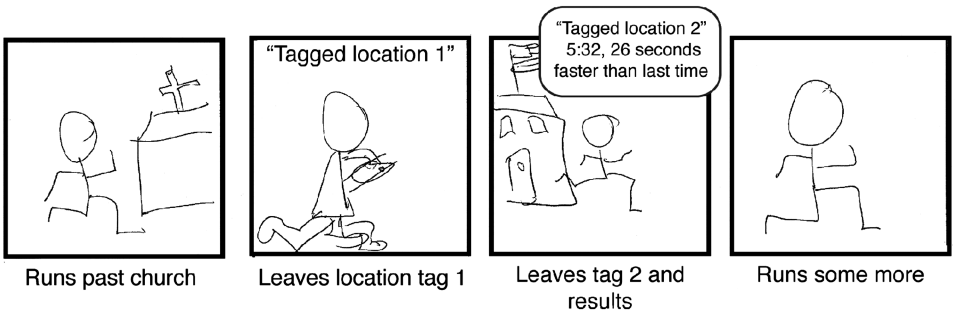
artifacts. A wide range of things could constitute the user's context in a given design space, but by relying on what the designers do represent in their artifacts, we can see how particular notions of context frame the design space. An example of the process can be found be found in the work of Designer 5 while designing an on-body device for runners that adds informational markers to locations as the user runs. Designer 5 is the principal designer for a design firm, and she has a wealth of experience in context-aware and other forms of design. Figure 7 shows a sequence of four of the 24 frames that made up a storyboard she created at the beginning of her work on the system.

What the storyboard includes and omits reveals a great deal about what the designer considered to be the relevant features of context. In this artifact the designer includes the locations (represented by the buildings) and the activity (represented by the illustration of the character running). In the second frame from the left, the designer's annotation shows the text that the system would provide to the runner as they interact with the device to "leave a tag." In the third panel, the designer shows a second location and again uses text to indicate how the system responds to the location and its ability to infer the pace of the runner based on their location. The response of the system also suggests that it is aware of the route the user has taken. Designer 5 explained that the storyboard was useful to help her understand the modes of interaction:

> Designer 5: "This was a very, very rough storyboard of how something might work in context; so, trying to identify what are the points that there needs to be visual or audio feedback, and what are the points there needs to be input—voice input, basically."

Her quote and the storyboard suggest that the designer views the context as arising from the user's activity and the locations where that activity occurs. We can see this because in the third frame, the system responds to her activity at a given location. If either the activity or the location were absent, the caption she included would make no sense. The feedback the system provides in the caption is clearly based on both

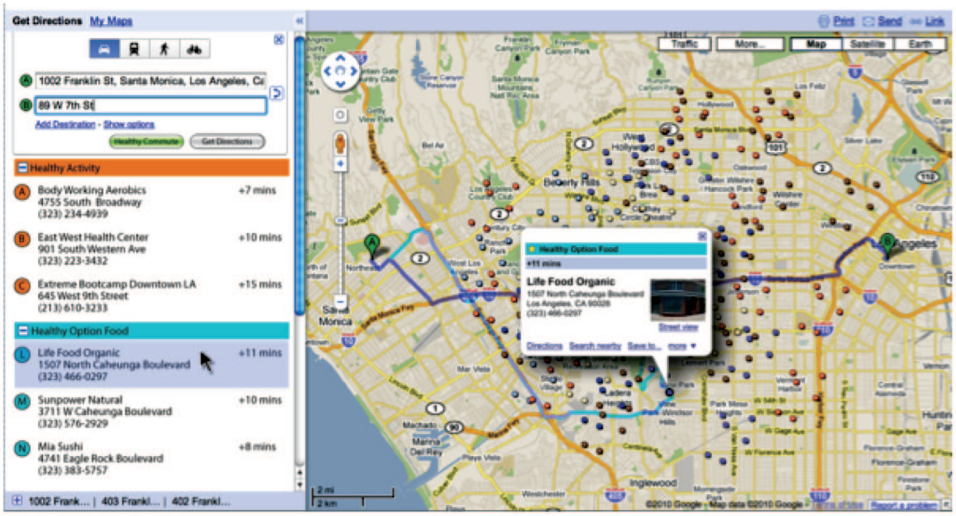
FIGURE 7. Four frames from Designer 5's storyboard representing her view of location in terms of physical locations rather than coordinates.



the runner's pace and the locations where the tags had been placed. Furthermore, we can see that the locations the system responds to are not coordinates but physical locations demarcated by the building in Frames 1 and 3. By depicting the system responding to the user's pace at physical locations by tagging the locations, the designer demonstrates that she views the context as emerging from the user's interaction with the world based on their location and activity. Context, in this example, could have been just the location or just the activity, but by viewing the context as involving the interaction of the two components, the designer frames the design space in a specific way. Furthermore, by focusing on the physical locations and making the system respond to these locations, she further frames a space to explore in her work. The design space she subsequently explores still contains numerous potential solutions, but her concept of context frames that space such that a successful design solution will include physical locations and the user's activity at those locations.

In Interview 9, we can see how an alternate formulation and representation of context imposes a different framing of a design space. Interview 9 was conducted with a senior designer at a major international design firm who also had considerable design experience, including experience with the design of context-aware systems. The project we discussed was an application to encourage healthy lifestyles. He used a "screen storyboard" in his design work on a system to recommend healthy lifestyle options to commuters (see Figure 8). This artifact provides an interesting contrast to how context was represented by Designer 9 in two ways. First, instead of representing a user, it represents only the commuter's route, thereby reframing the idea of context away from an embodied experience to a general trajectory that the user could follow on their commute. Second, his storyboard has the locations of interest to the user represented as "dots." By representing the context as dots corresponding to

FIGURE 8. A "screen storyboard" created by Designer 9 in his work on a system to promote healthy lifestyle choices for commuters.



fixed coordinates, as opposed to buildings or another representation of location, the designer frames the context as being solely derived from the user's location.

The systems discussed in Interviews 5 and 9 each contained a representation of the user's location and a route that the user traversed. However, these examples clearly contain differences in how these components were viewed. Designer 5 viewed the location as physical spaces and the route as something that resulted from the user's activity. Designer 9 viewed locations as a set of fixed coordinates and the route as a path that linked the user's current location to their home or work. In both cases, the way the designer framed the location imposed different constraints on the space their work inhabited and therefore the types of questions they would need to answer in further stages of the design process. Neither of these notions of context is "right," however, these differing views will shape the space they explore to find a satisfactory solution.

Encoding and Evaluating Context

In addition to framing the space in which designers work, designers' view of context plays an important role in establishing which solutions in that space are optimal or even feasible. This process has two steps—first *encoding* their concept of context and then *evaluating* various solutions in terms of the codes. To clarify, the process of encoding is not a deliberate activity by designers; it occurs incidentally when designers articulate concepts as they work. By articulating their concepts relating to context, designers instantiate sets of codes for the context-dependent components of the system. Doing so encodes the notion of context in a specific way, allowing them to communicate the ephemeral aspects of the context to peers and collaborators. It also establishes the basis on which the system will be evaluated.

In the interviews, we found numerous examples of codes helping to reify the designers' concepts for context. By analyzing the appearance of these codes, and how they were manifest in the designer's work, we can see that these concepts evolved through the design process. The location-aware smartphone game from Interview 8 illustrates this point. During the design process, the designers developed the "picking up," "dropping off," and "passing" interaction techniques that were central to their game. Each of the actions relied on the designers' view of how close the users were to the virtual objects or other players. In this example the phrase "passing" establishes a specific relationship between the users, the context (e.g., location and collocation), and their interaction with the application. Such codes further instantiate what they think "near" and "collocated" mean, based on the phone's ability to discern those factors. By using these codes, they transformed a complicated nebulous phenomenon of "picking up" to the distances between the geo-locations of a user and a virtual object.

Designers' concepts of context transform again when the codes begin to be used as criteria against which design objectives are evaluated. The initial articulation of codes, in many cases, draws on a commonsense understanding of the world and our interaction in it. As the work progresses the designers must determine if their

work is leading them to a finished product that satisfies their view for the context-dependent components of the system. Designer 2's work on an activity-aware health and fitness application provides an example of this process. The application she was developing would respond to the user's location or fitness activity, such as running, walking, or using an elliptical trainer. These different fitness activities were detected by a custom-made sensor suite worn on the user's hip and were communicated to the user's smartphone. Most people share a commonsense notion of what running is, but the designer must determine if the system responds to their notion of running in terms of the goals for the system. In this sense "running" becomes a code, and more importantly the designer must determine if the system responds appropriately when it thinks the user is running. While discussing the process of designing around the detected activity of the user, Designer 2 noted the following:

> Designer 2: "And see here, you can see in this one there's a question that we wrote and there's a big [collaborating developer's name] with a question mark so we would bring him over and there's another [collaborating developer's name] with a question mark. So [collaborator] was working with [different collaborator] and so when we would hit some 'Ooh is that even possible?' We would write it on the board and you can see we start working out the details of how things would get triggered."

This process of creating codes to crystalize a concept for context and then evaluate that idea was apparent across all the projects we reviewed; the designers articulated an initial set of codes, which would reify the designer's notion of context in the artifacts. Articulating context with artifacts allowed them to move their idea from the designer's world to the physical world. This enabled the designers to see what possible sets of solutions existed in the design space they had framed. As designers winnowed the design space, their notions for what constitutes context also narrowed. Finally, their concept for context was crystalized as a set of constraints that must be satisfied to demonstrate that the system is context-aware. During this process designers realize the shortcomings in their initial concepts for context, for example that they were too imprecise, limited, or infeasible. This resulted in the designer shifting their concept for context and thereby reframing the design space. The process of framing and coding occurred iteratively as designers searched for optimal solutions to enact their ideas.

Concordant with our observations, Cross has suggested that codes translate abstract requirements into concrete objects and that creating codes is a fundamental component of "designerly ways of knowing" (Cross, 1982, 2001). One important distinction in context-aware design is that the designers are obliged to describe the relationship between their concept for the system and the world. Although some areas of design in HCI—information visualization or information architecture, for example—can happen in the abstract, context-aware design fundamentally requires the designer to articulate and encode the relationship between the system and the world. Because the relationship between the system and the world must be encoded, context-aware designers' practices are more clearly aligned with Goodwin's (1994) concept

of coding. Goodwin argued that coding schemas allow professional practitioners to translate the relevant information in the world into objects of knowledge. He went on to argue that this process of encoding and communicating these codes are fundamental to the discourse of a profession. It is notable, though, that although designers are compelled to develop codes to reason about context on a project-specific basis, few *shared* codes for context currently exist in the HCI community that are formalized in a consistent way.

Context as a Unifying Element

As designers explored various solutions, their concept for context served to unify the various components of the system. Put another way, the way designers encoded context imposed constraints on the system. As we discussed earlier, context-relevant codes also served to establish evaluation criteria against which the success of various solutions could be judged. Moving toward a solution required creating artifacts that enabled the designer to explore multiple concerns simultaneously. This process usually involved the creation of increasingly sophisticated artifacts to evaluate the design. The fact that artifacts increased in sophistication through the design process is unsurprising (Newman & Landay, 2000). What was surprising was the difficulty designers faced in creating artifacts that allowed them to evaluate their ideas. This resulted in a number of trade-offs being made to create artifacts that could implement their ideas. These artifacts generally were constructed toward the end of the design process and provide insight into how the designers' view of context had evolved. Our analysis of these artifacts revealed that context played an important role in aligning the different design concerns. The difficulty inherent in determining and responding to context forced the designers to either winnow the design space to notions of context that were easier to implement or to explore previously unexplored areas of the design space.

One example of how context unified the designer's concept comes from Interview 7. The designer in this interview was a senior researcher in a research and design department of a major international technology firm. In the project we discussed, he was developing a new system that would enable social interaction through television. The context in question they were designing for was the social presence of the user's peers. The system allowed users to watch and comment on shows with their remotely located peers, but communicating others' presence and desire to watch television posed a series of prototyping challenges. Designer 7 explored several techniques to convey the peers' social presence including building a prototype using an Ambient Orb (Ambient, n.d.). However, he and his collaborators were ultimately dissatisfied with these options. In regard to exploring possible options, he commented,

> Designer 7: "So we did things like that, and it was largely a search for . . . the problem was we had these hardware requirements we knew we wanted, like we had this picture of the ideal thing that we could use, and nothing in the market really met that very well."

> Interviewer: "So you developed that prototype to sort of figure out what would be some other options because this system didn't actually satisfy some of the constraints that you had? Was it particularly the Wi-Fi?"
>
> Designer 7: "That was the main thing that we really needed. We have pretty good control over . . . the other thing was the, yeah. The Wi-Fi also, in addition to the reception problems, there was also a factor of the update response. So, you know, the . . . if you just go and buy an ambient orb from Brookstone or something like that, and you plug it in, the updates that you get are going to be within a time window of, I think, something like fifteen minutes or something like that, and we wanted something that was much more responsive than that."

The "picture of the ideal thing" to which he refers clearly entails some encoded notion of context. Essentially, he felt that a change in the social presence of the user's peers could be abrupt. To convey such changes required a way of communicating remote users' presence that would render changes visible in a short period. The Ambient Orb could not communicate this abruptness, thus they had to explore a different solution. Their final design did allow for abrupt changes in user presence, thereby allowing them to align other system aspects with their concept of context. To consider alternatives, they developed a blog of various technologies that could be used to communicate presence. While discussing this approach, Designer 7 stated the following:

> Designer 7: "We had a list of requirements as to how we wanted these things to function, and I think the blog just showed like a number of different ways in which you could have like connectivity, you could have visibility, and how much information you could convey."

From this quote, we can see that the designer was attempting to satisfy a set of concerns and prior prototypes had revealed shortcomings in the initial approach. To achieve a design that unified all of the constraints their concept entailed, they reframed their initial idea in a way that allowed them to explore a new form and method of implementation, thereby creating a solution that unified the elements of their notion of context.

5. DISCUSSION

The goal of this work is to understand the practices of design for context-aware systems from the perspective of the designers creating these systems. Now that we have detailed the concerns designers addressed and the process by which those concerns are resolved, we revisit our analysis to clarify how designers deploy these practices to transform their knowledge and views on context into a functional, context-aware system. As we revisit our findings, we discuss their implications for process, practices, and tool support, as well as their limitations.

5.1. Toward a Context-Aware Design Process

Our findings discussed earlier have helped to outline the process in which the designers we interviewed engaged and the role that context played in this process. Based on these findings, we propose the following process model to characterize the practice of context-aware design (see Figure 9). We see this process as having four phases. However, we do not suggest that these phases are visited only once. Rather, we observed that designers revisited these phases multiple times as their work evolved. Furthermore, we do not suggest a strict ordering to which these phases are visited. However, the order they are presented represented the order they are likely visited at least initially. These phases are as follows:

Framing: The designer(s) articulate and explore a concept of context, which imposes a set of limitations on what exist inside and outside of the design space their work inhabits.

Encoding: As the designer(s) discuss the behavior of the system, they begin to instantiate a vocabulary, or codes, to express this behavior. Encoding allows them to discuss the behavior of the system in a way that corresponds to their initial view as they continue to develop the system.

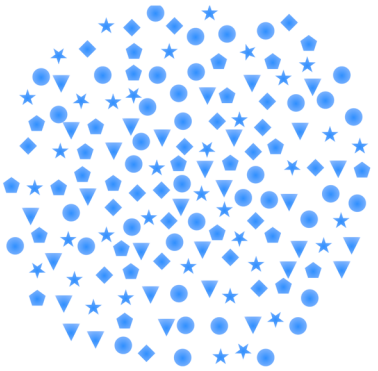
Unifying: As the designer(s) explore the design space, certain possible design solutions are brought to the foreground. These solutions impose additional constraints on the other concerns the designer addresses. As they seek to create a solution that corresponds to their encoded notion of context, ways to satisfy the various concerns will coalesce into a unified solution.

Evaluating: As the process continues, the designer will focus on a solution that satisfies the constraints according to their encoded formulation of context. This allows them to determine when they have arrived at a satisfactory solution.

Perhaps the most important question when looking at the concerns and process model we have outlined is, "What good does this do and for whom?" For practitioners, we believe having a defined language to discuss concerns inherent to the work on context-aware design can lead to a more comprehensive approach. In our interviews, we saw that more experienced designers better balanced these concerns. However, if they were to provide practical suggestions to new designers, they might struggle to communicate which concerns are general, which are specific to a given project, and which arise as a result of focusing on context. By providing a vocabulary of concerns, designers can better communicate how these concerns are addressed. Second, the process model contributes a way to view the role of context in influencing design work. For example, the initial formulation of context may be taken for granted if we fail to identify the role the concept for context plays in framing a design space. Similarly, acknowledging the ways a concept of context unifies a solution serves to draw attention to the formulation of context rather than the solutions that it instantiates. In general, providing a vocabulary to discuss context-aware design, we believe, will help practitioners and academics evaluate the concept that a system embodies.

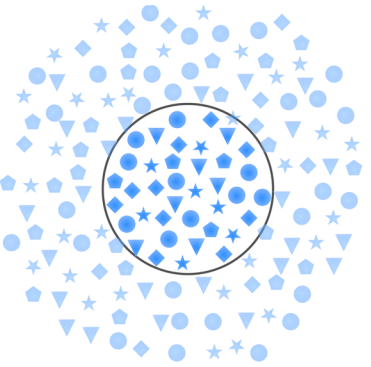
FIGURE 9. Context-aware design process model.

Design Space
Initially, there is an unlimited set of possibilities for the designer to explore.



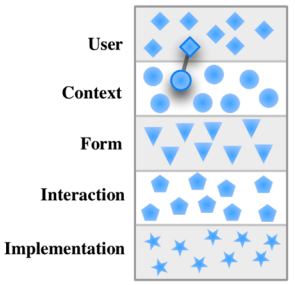
Each design concern is represented here by a symbol (user ◆, context ●, form ▼, interaction ⬟, and implementation ★).

1. Framing
Designers begin by articulating ideas for the behavior of a system and how it will be situated in the world. This frames the space the designer will explore.



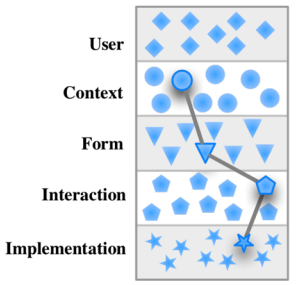
Example: The designer decides to build a fitness application for runners.

2. Encoding
Further elaboration of the concept encodes the behavior of the system as a relationship between the context and the other concerns.



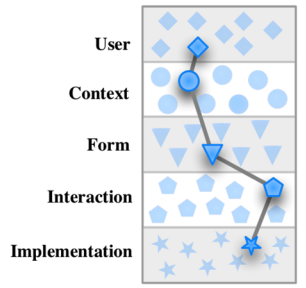
Example: It will detect when the user is running by looking for a change in their location over time.

3. Unifying
As the designer continues to explore the design space, constraints are surfaced that establish specific relationships.



Example: Determining if the user is running (*context*) will require knowing their exact location (*implementation*). An application for runners should be portable (*form*) and largely hands free (*interaction*).

4. Evaluating
The designer will then evaluate various solutions for the system against the codes they have established for the system's behavior.



Example: The designer has found a way to implement the system that satisfies their idea for the user, context, form, interaction, and implementation.

One valuable exercise when viewing this model is considering what is *not* context. As noted in the Study Methods section, we recruited participants using an intentionally wide framing for context. Despite our initial broad framing, we are not of the opinion that everything *is* context. It is our view that context-aware systems enable modes of interaction or experiences that draw on factors external to the system, such as

its context of use, features of its environment, as well as the unfolding interaction with the user. In addition, the designer of a context-aware system should be able to communicate how those factors impact the behavior of the system. Focusing on contextual factors expands upon traditional concerns in HCI such as the user's tasks, goals, and internal cognitive states, which have been the focus of other design models (e.g., GOMS, Card, Newell, & Moran, 1983; TCUID, Lewis & Rieman, 1993). We are not arguing that this model replaces prior design models. Instead, we believe this model is useful for understanding the additional implications to a design when enabling interaction or experiences that draw on these externalities. From this, we can see that if a concern does not draw on external factors to shape the user's interaction or experience with a system, then that *is not* necessarily engaging notions of context. Thus, our model is not a way to view design work in terms of context but rather serves to make explicit assumptions the designer has made about the context of the system.

5.2. Toward Context-Aware Design Practices

A major theme we encountered in our research was designers' difficulty producing artifacts that allowed them to gain an understanding of how context would impact the user's interaction with the system. To counter this, we saw designers adapt "standard" artifacts (e.g., storyboards, flowcharts, wireframes) to distill the complicated notion of context down to individual components of context needed to express the concept in a manner the artifact affords. For example, we saw this process with the designers in Interview 8 when they distilled the context of the system down to simple grid of locations drawn on a table. Certainly location was an important component of context, but as we saw in their work, this method did not provide a rich enough understanding of context to completely evaluate the design concept. In contrast, by using Wizard of Oz techniques and rapid prototyping tools (e.g., Arduino; http://www.arduino.cc/), designers were sometimes able to interactionally instantiate some forms of context, which allows the contextualized behavior of the system to be experienced (e.g., Designer 7's work prototyping a social television experience). By adapting existing artifacts, designers can consider the constraints and opportunities of context with other long-standing concerns.

In addition to affecting the content of artifacts and the details of their use within the design process, a focus on context-awareness foregrounds *new design practices* and their attending artifacts. The most notable example of this is Experience Prototyping (Buchenau & Suri, 2000), which recommends a certain approach to the creation of low-fidelity prototypes and/or simulated environments along with a set of techniques for engaging prospective users in contextually grounded usage experiences. Despite the advantages that Experience Prototyping offers context-aware design, we found that few designers employed this practice in the projects we documented. This finding can be interpreted in several ways. One possibility is that knowledge of how and when to apply such techniques is not yet widespread—even though the academic Design community has known of this technique for more than over a decade. Alternatively,

it may again point to differences in how designers view context. Experience Prototyping allows for an embodied, interactionally instantiated experience of the system. This sharply contrasts other artifacts that designers relied on that use a positivistic representation of context to understand and/or specify the envisioned behavior of the system.

The contrast between which artifacts were chosen and how they were applied sheds light on the tension between phenomenological and positivist views of context. Our findings demonstrate that both perspectives exist in contemporary context-aware design practice. Of interest, we see that both perspectives can be found within individual projects as pursued by individual designers, though admittedly this was less common than seeing one or the other. Although we are reluctant to make claims about consistent temporal patterns for activities and artifacts based on our study, our data suggest a provisional alignment of the phenomenological perspective with earlier design stages and of the positivist perspective with later design stages. Such an alignment makes sense because, as the system progresses, it becomes increasingly necessary that designers communicate the design in a way that can be implemented, which suggests an increasing emphasis on views of context that can be more readily captured, modeled, and acted upon by computing systems with discrete inputs and outputs.

5.3. Prescriptions for Context-Aware Design

Our analysis revealed the importance of generating and communicating codes to the members of the design team and the role that various artifacts play in the design of context-aware systems. This finding draws our attention back to Schön's (1992) discussion of design worlds. Schön argued that design worlds may be unique to a designer or shared across a broader community. This suggests that an analysis of the specific codes could help to establish a common vocabulary for context-aware design. Initiatives to facilitate designers' communication around their context-aware design practices could serve to facilitate the emergence of standard codes or coding practices. This, in turn, could improve the utility in design patterns for context-aware systems.

A question raised by this work is, How *ought* designers consider context within established design process methodologies? For example, we might ask how to reconcile efforts to understand, represent, and account for context within the stages of a standard user-centered design process (Moggridge, 2006; Saffer, 2006). From this perspective, we might seek to prescribe particular practices or representations that are appropriate for considering different types of context in the various stages of needs assessment, design exploration, prototyping, and evaluation. Alternatively, we might look to alternative process models such as Hartson and Hix's (1989) Star lifecycle model that emphasize the separation of concerns and encourages alternating one's focus between specific design issues or activities and evaluation of the integrated whole. Indeed, we were struck by the seeming absence of a consistent temporal pattern of activities and/or artifacts when we compared across projects. Some projects focused on technical concerns first (including the feasibility of detecting particular

contextual states or events); others sought to understand users' needs (especially with regards to various contexts of use); and others began with concerns about interaction, navigation, and content organization (including the expected behavior of the system in the face of contextual events). The nonlinear processes thus observed appeared to align with Moggridge's (2006) suggestion that design should begin by first considering the relevant constraints. He also suggested that the designer ought to move between these concerns in a noncyclical fashion. An open question is whether such flexibility is to be discouraged or embraced and whether standard models of the interaction design process are sufficient when designing context-aware systems.

Our work has implications for the development of tools to support context-aware design as well. As noted earlier, application frameworks and toolkits for building context-aware applications (e.g., Dey et al., 2001; Newman et al., 2010) can speed up the design life cycle significantly. However, our research highlights the fact that these tools, which focus attention on specifying system behavior, are more appropriate for later stages of design when a positivist perspective may be more fruitful. Low-fidelity prototyping tools like Topiary (Li, Hong, & Landay, 2004) or Activity Designer (Li & Landay, 2008) allow designers to take greater advantage of ambiguity, thereby reducing the need to specify all details of the system before exploring its interactional characteristics. However, even these tools emphasize the specification of screen layouts and state transitions, forcing consideration of the details of discrete contextual inputs and states.

Earlier still in the design process, tools that allow designers to explore interactional context, articulate and test assumptions, and experience alternative designs within different contexts of use will play an important role. ChronoViz (Fouse, Weibel, Hutchins, & Hollan, 2011) is an example of a tool that could be used by designers to collect, organize, and visualize sensor traces of user behavior to gain an understanding of the nature and diversity of contexts relevant to a given application. RePlay (Newman et al., 2010) extends this capabilities by allowing captured sensor data to be fed into application prototypes as they are being developed, thus closing the loop between early and late design and supporting the transition from phenomenological and positivist modes of thinking about context. Wizard of Oz (WOz) tools (e.g., Dow et al., 2005; Li, Hong, & Landay, 2007; MacIntyre, Gandy, Dow, & Bolter, 2004) can play an important role in facilitating the consideration of context throughout the design process, though such tools generally require that the designer specify a concrete set of valid inputs that can be simulated during a WOz experiment. Cleverly deployed, however, WOz tools can leave designers with enough flexibility to be able to improvise alternative contextual states and/or system responses, thereby allowing exploration of a system's interactional context in parallel with trying to ascertain its concrete behavior.

5.4. Limitations

One limitation of our study is that we relied on artifacts and the designers' recollections of methods. As we discussed earlier, it is our view that the process

of creating artifacts necessarily influences the way context is represented. Relying on the artifacts may have influenced designers to think about the process in terms of the representations of the process, which may create a bias toward a positivist interpretation of context. Practices such as experience prototyping serve to represent the designer's concept of context, but by relying on their memory of the practice, it undoubtedly loses some of the richness that being there would reveal. Despite this limitation, we do feel that the designers were able to discuss their design practices with sufficient detail for our analysis. However, an ethnographic study of context-aware design practice would be a valuable way to explore this topic in future work.

Finally, we sought to investigate the practices of context-aware designers and, to that end, with whom we were able to conduct interviews limits our view of context-aware design practice. We cannot say how representative the interviews we conducted are, but because of the convergence we began to see by the 11th interview, we suspect that these interviews are broadly representative. Furthermore, our findings aligned with the results of previous work on this area, which again suggests that the practices we observed were representative of the practices at large.

6. CONCLUSION

In this article, we have sought to contribute to the understanding of context-aware design by analyzing the artifacts and practices of designers who have worked on projects where context-awareness was a key component to the system's functionality. In doing so, we hoped to answer the question, "What do designers talk about when they talk about context?" Our findings suggest that designers' view of context is both phenomenological and positivist and that both views play important roles in the design process. However, the choice of methods and artifacts influences how one's view is manifest in the system that is developed. Designers' views evolve as they seek to satisfy five concerns in their work: *users*, *context*, *form*, *interaction*, and *implementation*. By addressing these concerns designers' understanding of context influences how they *frame* a design space, *encode* the contextual components of the system into a vocabulary, use this encoded vocabulary to *unify* a solution, and then *evaluate* the solution in terms of the codes. The process is not a straightforward march but relies on creating multiple representations of the context that are evaluated in conjunction with different concerns and by different stakeholders.

As the field of context-aware system design matures, it will be informative to see how the practices and perspectives we outline in this article transform. Undoubtedly, we expect that the range of contexts that designers address will only increase as technology becomes further embedded in our lives. We present this work in hopes that it will contribute to the grounding of design theory in the practices of actual designers. By doing so, we expect that this work will help to close the gap that exists between theory and practice in interaction design for context-aware systems.

NOTES

Acknowledgments. We are grateful for the inspiration and support of many colleagues, including Lisa Kleinman, Jose Sia, and Jennifer Milam. We also thank Morana Alač for her insight and feedback during our analysis of the interviews.

Editorial Record. First manuscript received on February 16, 2013. Revisions received October 14, 2013 and November 1, 2013. Accepted by Scott Klemmer. Final manuscript received February 6, 2014. — *Editor*

REFERENCES

Abowd, G. (2012). What next, UbiComp?: Celebrating an intellectual disappearing act. *Proceedings of the UbiComp 2012 ACM Conference on Ubiquitous Computing*. New York: ACM.

Ambient. (n.d.). Ambient products. Retrieved from http://www.ambientdevices.com/about/consumer-devices

Bauer, J. S., Newman, M. W., & Kientz, J. A. (2014). Thinking about context: Design practices for information architecture with context-aware systems. *Proceedings of the iConference 2014*. Berlin, Germany: iSchools.

Beyer, H., & Holtzblatt, K. (1999). Contextual design. *interactions, 6*, 32–42.

Blackwell, A. F., Whitley, K. N., Good, J., & Petre, M. (2001). Cognitive factors in programming with diagrams. *Artificial Intelligence Review, 15*, 95–114.

Bowen, G. (2008). Naturalistic inquiry and the saturation concept: A research note. *Qualitative Research, 8*, 137–152.

Buchenau, M., & Suri, J. F. (2000). Experience prototyping. *Proceedings of the DIS 2000 Conference on Designing Interactive Systems*. New York: ACM.

Card, S. K., Newell, A., & Moran, T. P. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ: Erlbaum.

Chung, E. S., Hong, J. I., Lin, J., Prabaker, M. K., Landay, J. A., & Liu, A. L. (2004). Development and evaluation of emerging design patterns for ubiquitous computing. *Proceedings of the DIS 2004 Conference on Designing Interactive Systems*. New York: ACM.

Cross, N. (1982). Designerly ways of knowing. *Design Studies, 3*, 221–227.

Cross, N. (2001). Designerly ways of knowing: Design discipline versus design science. *Design Issues, 17*, 49–55.

Davidoff, S., Lee, M., Dey, A., & Zimmerman, J. (2007). Rapidly exploring application design through speed dating. *Proceedings of the UbiComp 2007 ACM Conference on Ubiquitous Computing*. New York: ACM.

Dey, A. K. (2001). Understanding and using context. *Personal and Ubiquitous Computing, 5*, 4–7.

Dey, A., & Abowd, G. D. (2000). Towards a better understanding of context and context-awareness. *Proceedings of the CHI 2000 Workshop on the What, Who, Where, When, and How of Context-Awareness*. New York: ACM.

Dey, A. K., Abowd, G. D., & Salber, D. (2001). A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human–Computer Interaction, 16*, 97–166.

Dourish, P. (2004). What we talk about when we talk about context. *Personal and Ubiquitous Computing, 8*, 19–30.

Dow, S., MacIntyre, B., Lee, J., Oezbek, C., Bolter, J. D., & Gandy, M. (2005). Wizard of Oz support throughout an iterative design process. *IEEE Pervasive Computing, 4*, 18–26.

Dow, S., Saponas, T. S. T., Li, Y., & Landay, J. A. (2006). External representations in ubiquitous computing design and the implications for design tools. *Proceedings of the DIS 2006 Conference on Designing Interactive Systems*. New York: ACM.

Fouse, A., Weibel, N., Hutchins, E., & Hollan, J. D. (2011). ChronoViz: A system for supporting navigation of time-coded data. *Proceedings of the CHI 2011 Conference on Human Factors in Computer Systems*. New York: ACM.

Gaver, W. (2011). Making spaces: How design workbooks work. *Proceedings of the CHI 2011 Conference on Human Factors in Computer Systems*. New York: ACM.

Goodman, E., & Wakkary, R. (2011). Understanding interaction design practices. *Proceedings of the CHI 2011 Conference on Human Factors in Computer Systems*. New York: ACM.

Goodwin, C. (1994). Professional vision. *American Anthropologist, 96*, 606–633.

Greenberg, S. (2001). Context as a dynamic construct. *Human–Computer Interaction, 16*, 257–268.

Hartson, H. R., & Hix, D. (1989). Toward empirically derived methodologies and tools for human-computer interface development. *International Journal of Man–Machine Studies, 31*, 477–494.

Hong, J., & Landay, J. (2001). An infrastructure approach to context-aware computing. *Human–Computer Interaction, 16*, 287–303.

Hutchins, E., & Hollan, J. (1985). Direct manipulation interfaces. *Human–Computer Interaction, 1*, 311–338.

Landay, J. A., & Borriello, G. (2003). Design patterns for ubiquitous computing. *Computer, 36*, 93–95.

Lewis, C., & Rieman, J. (1993). Task-centered user interface design. Retrieved from ftp.cs.colorado.edu/pub/cs/distribs/clewis/HCI-Design-Book

Li, Y., Hong, J. I., & Landay, J. A. (2004). Topiary: A tool for prototyping location-enhanced applications. *Proceedings of the UIST 2004 Symposium on User Interface Software and Technology*. New York, NY: ACM.

Li, Y., Hong, J., & Landay, J. (2007). Design challenges and principles for Wizard of Oz testing of location-enhanced applications. *IEEE Pervasive Computing, 6*, 70–75.

Li, Y., & Landay, J. A. (2008). Activity-based prototyping of ubicomp applications for long-lived, everyday human activities. *Proceedings of the CHI 2008 Conference on Human Factors in Computer Systems*. New York: ACM.

Lim, Y. K, Stolterman, E., & Tenenberg, J. J. (2008). The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas. *ACM Transactions on Computer-Human Interaction, 15*, 1–27.

Lincoln, Y. (1985). *Naturalistic inquiry* (Vol. 75). New York, NY: Sage.

MacIntyre, B., Gandy, M., Dow, S., & Bolter, J. D. (2004). DART: A toolkit for rapid design exploration of augmented reality experiences. *Proceedings of the UIST 2004 Symposium on User Interface Software and Technology*. New York: ACM.

Moggridge, B. (2006). *Designing interactions*. Cambridge, MA: MIT Press.

Morville, P. & Rosenfeld, L. (2008). *Information architecture for the World Wide Web: Designing large-scale web sites*. Sebastopol, CA: O'Reilly Media.

Newman, M. W., Ackerman, M. S., Kim, J., Prakash, A., Hong, Z., Mandel, J., & Dong, T. (2010). Bringing the field into the lab. *Proceedings of the UIST 2010 Symposium on User Interface Software and Technology*. New York: ACM.

Newman, M., & Landay, J. (2000). Sitemaps, storyboards, and specifications: A sketch of Web site design practice. *Proceedings of the DIS 2000 Conference on Designing Interactive Systems*. New York: ACM.

Norman, D. A. (1991). Cognitive artifacts. In J. M. Carroll (Ed.), *Designing interaction: Psychology at the human-computer interface* (pp. 17–38). New York, NY: Cambridge University Press.

Saffer, D. (2006). *Designing for interaction: Creating smart applications and clever devices*. San Francisco, CA: Peachpit Press.

Schilit, B., Adams, N., & Want, R. (1994). Context-aware computing applications. Mobile Computing Systems and Applications, 1994. *Proceedings of the WMCSA 1994 Workshop on Mobile Computing Systems and Applications*. Washington, DC: IEEE.

Schilit, B. N., Hilbert, D. M., & Trevor, J. (2002). Context-aware communication. *IEEE Wireless Communications, 9*, 46–54.

Schmidt, A. (2000). Implicit human–computer interaction through context. *Personal and Ubiquitous Computing, 4*, 191–199.

Schön, D. A. (1992). Designing as reflective conversation with the materials of a design situation. *Knowledge-Based Systems, 5*, 3–14.

Stolterman, E. (2008). The nature of design practice and implications for interaction design research. *International Journal of Design, 2*, 55–65.