The Impact of Agile on User-centered Design: Two Surveys Tell the Story

David Dayton
Independent Researcher
18 PUBLICATIONS **282** CITATIONS

Carol Barnum
Kennesaw State University
41 PUBLICATIONS **864** CITATIONS

APPLIED RESEARCH

SUMMARY

- Discusses the impact of an agile software development process on usability testing
- Reports opinions about usability testing within a company before and after a change to agile
- Presents strategies to incorporate usability testing into agile product development

The Impact of Agile on User-centered Design: Two Surveys Tell the Story

DAVID DAYTON AND CAROL BARNUM

THE CHALLENGE: INCORPORATING USABILITY WITHIN A DEVELOPMENT PROCESS

Technical communicators often face the challenge of documenting poorly designed products. As a result, technical communicators are frequently the ones who advocate for usability testing. Long recognized—at least among ourselves—as the user's advocates, we are often most acutely aware of the need to see and learn from users so as to improve the user experience. Those of us doing usability testing of the documentation or participating with developers on product teams know the value of a user-centered design (UCD) process in building the user experience into the ongoing development cycle.

The growing popularity of the Usability and User Experience Special Interest Group of STC is one indicator of the interest in and commitment to the important role that technical communicators play in fostering usability and user-centered design within product development. The growing number of sessions on usability and user experience at the STC Annual Summit is another indicator. Nevertheless, surveys to gauge awareness and use of usability methods have consistently shown that, although awareness of UCD and usability testing is not the problem it once was, only a small minority of development efforts incorporate UCD and/or usability testing (Frick and colleagues 2001; Mao and colleagues 2005; Venturi and Troost 2004). After experiencing two decades of continuous growth and maturity as a discipline, usability practitioners still find themselves faced with the need to educate companies about the value of adopting a comprehensive UCD approach to the design and development of their products. Usability testing remains the profession's best available means of persuasion. Surveys of usability professionals have consistently found that usability professionals rate usability testing as the most effective UCD method (Gunther and colleagues 2001; Rosenbaum and colleagues 2000).

The most important question that the usability profession (including technical communicators involved in this work) still faces is this: How can we leverage the persuasive power of usability tests to move companies from a narrow focus on user-interface problems to a thoroughly user-centered mindset, creating a usability culture within the organization? With the rapidly growing interest in agile development methods, how can we make the case for user-centered design practices when the development process has now moved from fast to relentlessly fast-forward?

Compelling benefits of going agile

Since the development of the agile manifesto in 2001 (see http://en.wikipedia.org/wiki/Agile_Software_Manifesto), more and more software development companies have embraced this rapid development process. As defined by *Wikipedia,* agile software development methodologies "generally promote: A project management process that encourages frequent inspection and adaptation; a leadership philosophy that encourages team work, self-organization and accountability; a set of engineering best practices that allow for rapid delivery of high-quality software; and a business approach that aligns development with customer needs and company goals" (*Wikipedia*, s.v. "Agile Software Development").

Software companies switching to agile have found the process to be very appealing because of its lack of documentation, use of a team approach, and rapid product development that can be validated in short cycles. The rapid development process is particularly attractive compared with the traditional "waterfall" software development process, in which development moves through sequential stages, flowing downward like a waterfall, from requirements gathering to design to construction/coding to testing/validation to integration and release.

Manuscript received 10 August 2008; revised 7 January 2009; accepted 24 January 2009.

In the most common agile process, called scrum, a team is headed by a scrum master, whose job is not to lead the team (teams are leaderless) but to remove impediments to the team's ability to achieve its goals. Daily stand-up scrum meetings typically lasting ~15 minutes provide a quick way for the team to assess progress and address issues. A planning cycle, or sprint, can last 15 days to 1 month with typical sprints lasting 3 weeks. Planning begins before the first day of a sprint cycle and moves in cycles throughout development. At the end of each sprint, a retrospective is held to assess how the sprint went and evaluate lessons learned.

Compared with the waterfall process, in which the place for usability testing could be planned ahead and built in to the development timeline, the scrum process provides little time to slip usability testing into a sprint. The team decides on its priorities for each sprint. Priorities are included in a product backlog, which is reviewed at the end of each sprint to see what priorities should be included in the next sprint. Unless a team member argues successfully for making usability testing a priority, it is unlikely to be included because the always urgent, overriding goal is developing working features at the end of each sprint cycle. Agile methodologies have thus changed the development process in ways that provide difficult new challenges for user experience practitioners. (For more background on agile development, see its entry at en.wikipedia.org, or visit www.agilealliance.org.)

Success strategies from a few companies

A few published accounts describe methods used to incorporate user-centered design within agile development processes (McInerney and Maurer 2005; Miller 2006; Patton 2002; Sy 2007). The success model presented by Sy (2007) about experiences at Autodesk (formerly Alias) relies on having internal usability or interaction designers on the agile team or working in parallel with the team. That solution would not be available, of course, for teams—such as those we have usually worked with—that do not include usability or interaction designers. In the other cases cited above, "success" comes with various cautionary notes and conditions, such as:

- Requiring full integration of the customer with access to real users for validation and everyone on the team in place for daily scrums
- "Limited success" in adapting traditional UCD approaches to agile practices because of schedule impact and difficulties making design changes late in the cycle
- Designers' misgivings about decisions being made early in iterations without a view of the entire user interface
- Teams feeling confident they understand what the software should do and why, without actually confirming that the product provides a better user experience

William Hudson (2003) critiqued the marriage of agile and usability in a hypothetical debate between a user experience practitioner and an Extreme Programming (agile) expert. Among the points made by the user-experience practitioner are these:

- Although accommodating change "no matter how advanced development might be" is a plus, this flexibility can cause user interface design issues and ensure usability problems.
- A focus on making things better for the team does not necessarily translate into making the experience better for the users.
- Even if real users become members of the agile team, they quickly lose their ability to see the issues from the perspective of actual users.

Technical communicators get on the agile bandwagon

Technical communicators have joined the discussion about the role we can play as well as the challenges of working in an agile environment, emphasizing the opportunity to be key players in the process and to get closer to the user. Recent evidence of the involvement of technical communicators and managers can be seen in the small but growing number of articles in *Intercom* (O'Connor 2007; Sigman 2007) and presentations at STC annual conferences (Fox and Kramer 2008; MacAller 2008; Mazet 2008; Taliaferro 2006; Wethington 2008). In these articles and presentations, the technical communicators describe how to develop documentation in an agile environment, focusing most often on how to adapt to the challenges. Speaking of their experiences working on or managing the documentation aspects for agile teams at Hewlett-Packard, NetIQ, Microsoft, Troux Technologies, and elsewhere, many of these technical communicators expressed concern, even trepidation, when they first entered into the agile development cycle; all came out squarely in support of the process, after having adapted to it.

The common theme expressed in these articles and presentations is that agile provides the following opportunities for technical communicators:

- Greater accountability and, thus, increased visibility
- Inclusion in daily stand-up meetings (scrums) where information about issues and problems is presented and solutions proposed; more information received earlier
- Shared knowledge about all components of the product
- Credibility for technical communicators as team members/co-developers; improved morale

The challenges and issues associated with the switch to an agile process include these:

- Change is required; some cannot embrace the mindset.
- Planning is required; strong management support is needed.
- Training is required to become agile; debrief sessions are required to adjust estimated hours for completion of work and attendance at daily scrum meetings.

Although some mention the benefit of learning more about the user in the agile methodology, none of the accounts cited discusses ways to build usability testing into the agile process to accomplish this goal.

CASE STUDY: BEFORE AND AFTER AGILE

In January 2004, the Usability Center at Southern Polytechnic State University began working as the usability vendor for a software company that develops and sells software solutions for the IT industry. The company approached us in late 2003 with a successful, mature product; they wanted a single usability test of the new version to get feedback about the user experience. We convinced the company that earlier and more frequent testing would give them feedback about their users' experience that could be built into the product in development and would educate the developers on usability issues that could be translated into better designs for their other products.

Over a 2-year period, the company committed heavily to usability testing in which key product team members participated with us in designing and running the tests, all of which were also observed from an adjoining conference room by many team members. During usability testing sessions, as many as 30 employees participated as team members or observed test sessions—everyone from the president of the company and the vice president for research and development, to product and development managers, quality assurance and technical support, sales and marketing, and information developers.

During that 2-year period, we also helped the company conduct iterative testing on prototypes, early interaction models, and products in various stages of development. Contextual inquiry was begun to explore ideas for developing a new product. We called this company our "poster child" for its success in understanding and incorporating a user-centered design process. The company received positive reinforcement from product reviews and from the effect on its bottom line, measured by increased conversion rates from try-to-buy and decreased support calls, both in duration and number.

After its decision in January 2005 to begin changing to the agile methodology, the company completed only two formal usability tests that spring and one more over a year later. Although management and product development continued to express strong support for usability testing, the company's prior commitment to testing with real users in a formal process was not maintained, despite the urgings of the information developers.

What replaced an iterative, user-centered design process emphasizing usability testing with evaluators carefully recruited to represent real users? For starters, the product manager assumed responsibility for being the chief customer proxy, and everyone on the team was encouraged to take the customer's perspective at all times. Formal usability tests, conducted in a laboratory with real users, were replaced by "friends and (company) family" in what were called "informal usability tests." These informal tests were usually postponed until late in the development cycle when developers had no new features to add to the product. In one instance when such a test uncovered several usability defects that the information developers thought should be high priority, the product manager and scrum master vetoed fixing them because the development cycle was rapidly approaching the target release date.

Although some information developers argued for formal usability testing to be conducted early in the development cycle, product managers and their teams had come to regard such testing as a luxury, which could not be afforded in the early stages of development and was reserved for the end of a development cycle, by means of informal testing. Such informal testing was used as a way of measuring success and providing starting-point goals for the next cycle. In addition, the informal testing process did not include the participation of company observers, which had been a hallmark of the formal testing conducted at the Usability Center at Southern Polytechnic.

Other methods even further removed from the rigor of formal usability testing became highly valued in the company's agile process. Usability feedback was solicited at the end of each sprint at a meeting in which a team would present a web-delivered demonstration of new and improved features and functionality to a group composed of managers, members of other teams, and some actual users, all of whom were invited to ask questions and make comments. One of the three product teams also conducted a "group of 10" test. Ten employees not connected to product development would download, install, and use the product for at least 15 min. They were asked to note what did not work for them, and 9 of 10 had to be successful before management approved the product release.

The change from formal testing to informal testing without the observation of others related to the product can best be understood by comments made by an information developer in an interview, in which he described the impact on developers of observing formal usability testing at the Usability Center at Southern Polytechnic. "[T]hey were much more enthusiastic after observing the process because they saw feedback from real people." However, the initial enthusiasm rapidly waned: "[B]ut after getting back, it's hard to maintain that momentum." This information developer explained that the informal tests in-house just did not have the same impact on the development team. Because few besides the two running the tests observed the "friends and family" users, the findings were conveyed to the team through reports with screen shots and video clips, but hardly anyone took the time to view the clips, and the validity and severity of the findings were often questioned. By way of contrast, this information developer noted that for the tests at the Usability Center, "The whole group who had witnessed did a much more thorough evaluation of what took place."

In summary, when agile methods changed the product development focus to getting new and improved functionality to users as quickly as possible, some team members—including most of the information developers—began to feel that usability was getting sidelined. These team members were particularly concerned that informal usability tests, although cited as proof of the company's continuing concern for improving the user experience, were not keeping developers focused on user needs the way formal usability tests had done.

In November 2004, shortly before the company began switching from the waterfall development methodology to agile, we administered a survey about usability testing opinions and attitudes to the employees of this company (Barnum and colleagues 2005). With the company's permission and complete cooperation, we conducted a similar follow-up survey in April 2007 to determine whether attitudes toward usability testing had changed since the adoption of agile processes. In the following two sections, we present and compare the results from the two surveys.

Results from the first survey (2004)

After a year of working with this company's product teams on usability testing of various products, we requested and received permission to conduct a survey of employees' attitudes resulting from exposure to usability testing and user-centered design methods. The online survey was administered in November 2004 with special urging for response from the president of the company. Ninety-six employees completed the survey—approximately a 64% response rate.

In a nutshell, the survey showed that employees held a strongly positive view of usability testing. To the extent that they knew about the usability testing conducted at the Usability Center at Southern Polytechnic, the respondents were overwhelmingly positive about it. Table 1 condenses the data from nine Likert-type statements about usability testing; respondents were offered five choices—*agree strongly*, *agree somewhat*, *neutral*, *disagree somewhat*, and *disagree strongly*, plus a *no opinion/don't know* choice. The statements were adapted from a longer set of Likert-type statements developed and tested by Frick and colleagues (2001), which followed recommended procedures for presenting sets of Likert-type statements by phrasing half of them positively and the other half negatively. We phrased four of our nine statements negatively, asserting something negative about usability testing rather than stating something positive; for these four items, the answer scale was reverse-coded, meaning that *disagree strongly* had a value of 5, the same value accorded *agree strongly* for the positively worded statements. We calculated the mean value of all valid responses for each statement; left out of the calculation were any *no opinion/don't know* answers, along with missing responses.

We also calculated the percentage of responses to each statement considered to represent a positive opinion about usability testing. We derived this percentage from the ratio of (a) the number of responses on the side of the scale favorable to usability testing and (b) the total number of responses on the scale—that is, not counting no opinion/don't know and missing responses. As shown in Table 1, the Likert-type statement obtaining the lowest "% Positive" response was the one stating that usability testing created extra work for developers; only one third of the respondents to that statement indicated disagreement (and disagreement, in this case, reflects a positive opinion about usability testing). For all the other statements, the proportion of responses coded as positive toward usability testing was greater than the proportion coded as negative toward usability testing.

The data in Table 1 clearly show that opinions about usability testing and its impacts were overwhelmingly positive on six of the nine Likert-type statements. Statistical tests explained and summarized in Table 2 show that concerns about some perceived tradeoffs of usability testing—additional development effort, time, and costs—significantly decreased the strength of the positive response toward usability testing. The statistical analysis also shows a significant difference in how strongly the survey respondents agreed that usability testing had made development teams more customer-centered compared with how strongly they agreed that testing had made the teams more concerned about product usability. We had expected that the responses to those two statements would not be significantly different because we ourselves think of concern for usability as requiring an unrelenting focus on what customers need, want, and experience in relation to a product. Nevertheless, subgroups of respondents who identify closely with customers because of their frequent interaction with them (sales, marketing, technical support) or advocacy for them (information developers) may have perceived a distinction between the developers' increased concern about usability and whether the developers had truly become more customer-centered.

TABLE 1: DESCRIPTIVE DATA SUMMARIZING RESPONSES TO NINE LIKERT-TYPE STATEMENTS IN 2004 SURVEY

	n valid	Percent positive*	Mean	Standard deviation (SD)
Usability testing . . .				
Should be used more often	82	*88*	4.46	0.77
Has made development teams more concerned about usability	80	*91*	4.36	0.75
Has not been worth the effort**	75	*87*	4.27	0.95
Has improved ease-of-use of products tested	79	*86*	4.19	0.82
Has improved usefulness of products tested	76	*78*	4.07	0.79
Has made development teams more customer-centered	74	*74*	3.82	0.91
Has hurt cost-effectiveness**	61	*57*	3.79	1.07
Has slowed down product development**	69	*39*	3.20	0.98
Has made extra work for developers**	67	*33*	3.03	1.07

Note: Statements listed in descending order of mean score on 1–5 scale, with 5 = most positive toward usability testing

**This percentage was calculated from the ratio of (a) the number of responses on the side of the scale favorable to usability testing and (b) the total number of responses on the scale—that is, not counting no opinion/don't know and missing responses. The ratio a/b was multiplied by 100 to obtain the percentage of responses considered to represent a positive opinion about usability testing.*

***Statement phrased negatively, so responses were reverse coded: a higher number reflects a stronger pro-usability-testing attitude, the same as for positively phrased statements.*

In addition to the Likert-type statements, the 2004 survey collected opinions about usability testing through open-ended questions. Analysis of the comments responding to these questions showed that a few developers were skeptical about the ultimate value of usability testing; however, the overwhelming majority were positive about usability testing. One question asked respondents to sum up their opinions about the effects of usability testing on product development and product quality. The 7 respondents expressing negative opinions were far outnumbered by the 56 respondents who were positive, 9 of whom specifically stated that the company should do more to incorporate a focus on usability into the development cycle. Table 3 provides sample comments that reflect the tenor of the qualitative responses received, ranging from those who were strongly negative about usability testing to those who were strongly positive.

Results from the second survey (2007)

Our direct and frequent involvement with the company as a usability testing vendor slowed in mid-2005, after the company's switch to agile spread to all product development teams, and it stopped after one last test in the summer of 2006. Nonetheless, we maintained contact with key managers, including the Information Development Manager, who helped us launch a follow-up survey using the same nine Likert-type statements that had been presented to all respondents to the first survey. This second survey, fielded in April 2007, focused on three topics: formal usability testing (laboratory testing and related processes), informal usability testing (the internal methods used in their agile development process), and opinions toward the agile development method and its impact on usability testing.

TABLE 2: RESULTS OF POST HOC PAIRWISE COMPARISONS OF MEAN DIFFERENCES IN RESPONSES TO NINE LIKERT-TYPE STATEMENTS IN 2004 SURVEY

	Not worth	Use more	More concern	Easy use	Usefulness	Hurts cost	Customers #1	Slows down	More work
Not worth		1.00	1.00	1.00	1.00	0.015*	0.007*	0.000*	0.000*
Use more			1.00	1.00	1.00	0.022*	0.001*	0.000*	0.000*
More concern				1.00	1.00	0.095	0.007*	0.000*	0.000*
Easy use					0.698	0.104	0.001*	0.000*	0.000*
Usefulness						0.964	0.184	0.000*	0.000*
Hurts cost							0.964	0.038*	0.015*
Customers #1								0.077	0.052
Slows down									1.00
More work									

Note: Data collected from 52 respondents were analyzed in a repeated-measures one-way analysis of variance, with list-wise deletion for missing responses and each Likert-type statement treated as a within-subjects factor. The sphericity assumption was not met so the Greenhouse-Geisser correction was applied. The main within-subjects effect was significant, $F(5.70, 290.83) = 21.46$, $P < 0.0005$, partial eta squared = 0.30. The matrix shows the outcome of post-hoc pairwise comparisons performed using the Bonferroni adjustment for multiple comparisons.

**Significant at $P \leq 0.05$. The exact P value for each pairwise comparison is given.*

The company was as interested as we were in gathering data on its employees' opinions about usability testing and agile. The president encouraged participation, as he had in the first survey, by sending an e-mail to everyone in the company inviting them to take the online survey. As before, the response was excellent, with 100 valid responses returned, 4 more than in the first survey. In this second survey, responses to eight of the nine Likert-type statements about formal usability testing were just as positive as in the 2004 survey. Table 4 shows the means and percentages of positive responses on the nine Likert-type statements that in the 2007 survey were presented as separate sets for formal testing, informal testing, and agile development, respectively. The one Likert-type statement about formal usability testing that produced a significant difference in means from the 2004 survey was the statement that usability testing "has made development teams more concerned about usability issues." The mean of the responses to that statement decreased from 4.36 in 2004 to 4.05 in 2007, which is significant by the *t* test for independent groups ($P = 0.023$). The corresponding proportion of positive responses to that question fell from 91% to 80%.

Looking closely at the data in Table 4, we find that the fall off in respondents answering *agree strongly* on this item in 2007 (13 fewer than in 2004) is equal to the increase in respondents choosing the neutral point on the scale (4) and *no opinion/don't know* (9), with a few more *disagree somewhat* answers (4) underlining the only statistically significant weakening of the pro-usability testing opinions in the 2007 survey compared with results from 2004. This lone significant difference in means between the 2004 and 2007 results may say more about some respondents' attitude toward the company's developers than it does about their regard for formal usability testing. We know from talking to people inside the company and from analyzing responses to open-ended questions in the 2007 survey that several information developers and a few others in management and marketing questioned the depth of the development teams' commitment to usability testing after the switch to an agile process.

TABLE 3: SAMPLE OF 63 OPEN-ENDED ANSWERS ABOUT USABILITY TESTING IN WATERFALL PROCESS (2004 SURVEY)

Strongly negative *n* = 2 (3%)	I think in some cases the usability review went too far. Too many advanced features were hidden, making the product more difficult to use by a knowledgeable person.
Mildly negative *n* = 5 (8%)	The jury is out on this. I will wait for more feedback. I think that there [was] some functionality left out of the product, because of the testing. The functionality missing is more of a troubleshooting, power-user functions that a casual user wouldn't be interested in.
Generally positive *n* = 23 (37%)	I have seen the products' UIs [user interfaces] improve.
Strongly positive *n* = 24 (38%)	Can't believe we survived so long without it! This has been a worthwhile initiative and welcome change to the product development process. I think the key was that both Development (especially product development and information development) and Product Manager embraced and wanted to move forward with user-centered testing.
Positive, wants more *n* = 9 (14%)	While I hold a high opinion on the effects of usability testing, I have concerns that it may be relegated to a post-production process and not an approach in design.

Other significant differences. Another key question we hoped the 2007 survey would answer was how formal usability testing was perceived in comparison to both informal testing and agile development. In presenting the statements about informal testing and agile to respondents of the 2007 survey, we used preliminary definitions and questions to direct those with little or no knowledge of these practices to skip those sections. That is why the range of valid responses shown in Table 4 varies substantially across the three sets of Likert-type statements. To compare the means on the 2007 survey across the three topics requires a repeated measures test on the data from only those respondents who gave answers to all three sets of Likert-type statements.

Table 5 presents the means, number of responses (*n*), and standard deviation (SD) summarizing the data from those 48 respondents. We conducted two-group paired *t* tests on the means for (a) formal usability testing and informal testing, (b) formal usability testing and agile development, and (c) informal testing and agile. The means found to be significant are identified in the table footnotes; we will discuss them next.

The two-group paired *t* tests found significant differences in 5 of 27 pairs of means tested. Responses to the statement about:

- Improving product ease-of-use: significantly more positive for informal testing than for agile development
- Improving product ease-of-use: significantly more positive for formal testing than for agile development
- Improving the usefulness of products: significantly more positive for formal testing than for agile development
- Making development teams more concerned about usability: significantly more positive for formal testing than for agile development
- Being used more often: significantly more positive for formal testing than for informal testing

Of the nine Likert-type statements, four focus on goals of a user-centered design (UCD) process: improving product ease of use and usefulness and making development teams more customer-centered and concerned about usability issues. On three of those explicitly UCD-focused statements, formal usability produced significant differences compared with agile development, whereas informal testing produced a significant difference on only one: ease of use. The only significant difference between means for formal and informal testing came on the statement about which should be used more often; respondents were significantly more positive toward formal testing on that question.

In sum, respondents in the 2007 survey who considered themselves knowledgeable about both types of usability testing and the agile development process expressed opinions that were a bit more positive toward formal usability testing than toward informal testing but

TABLE 4: DESCRIPTIVE DATA SUMMARIZING RESPONSES TO NINE LIKERT-TYPE STATEMENTS ABOUT FORMAL USABILITY TESTING, INFORMAL USABILITY TESTING, AND THE AGILE PROCESS (2007 SURVEY)

	Formal usability testing				Informal usability testing				Agile development process			
	n valid	Percent positive*	Mean	SD	*n* valid	Percent positive*	Mean	SD	*n* valid	Percent positive*	Mean	SD
Usability testing/agile development . . .												
Should be used more often/should continue to be used	77	*92*	4.43	0.64	46	*93*	4.26	0.57	69	*84*	4.35	0.89
Has made development teams more concerned about usability	75	*80*	4.05***	0.91	47	*81*	3.94	0.82	63	*63*	3.52	0.96
Has not been worth the effort**	77	*90*	4.42	0.89	45	*89*	4.29	0.92	68	*82*	4.26	1.05
Has improved ease-of-use of products tested	78	*82*	4.13	0.86	46	*78*	3.98	0.77	66	*55*	3.55	0.95
Has improved usefulness of products tested	78	*78*	3.99	0.80	46	*83*	4.07	0.80	67	*67*	3.78	1.00
Has made development teams more customer-centered	80	*73*	3.89	0.89	47	*68*	3.70	0.92	66	*55*	3.59	0.96
Has hurt cost-effectiveness**	61	*70*	4.03	0.87	36	*86*	4.17	0.74	60	*68*	3.90	1.08
Has slowed down product development**	72	*46*	3.28	1.01	44	*48*	3.25	1.06	67	*63*	3.75	1.05
Has made extra work for developers**	69	*36*	3.12	1.05	41	*32*	3.02	1.06	62	*48*	3.32	1.08

Note: Statements listed in same order as in TABLE 1, with mean based on 1–5 scale; 5 = most positive toward usability testing/agile development

**The percent positive figure for each group of Likert-type statements was calculated from the ratio of (a) the number of responses on the side of the scale favorable to the focus of the statements and (b) the total number of responses on the scale—that is, not counting no opinion/don't know and missing responses. The ratio a/b was multiplied by 100 to obtain the percentage of responses considered to represent a positive opinion about the focus—formal usability testing, informal testing, or agile.*

***Negative statement, so responses were reverse coded: a higher mean reflects more answers on the side of the scale considered to be positive toward the focus of the question.*

****Significantly different from mean for same statement in 2004 survey (see TABLE 1);* P = 0.023, t-test for independent groups.

TABLE 5: DESCRIPTIVE DATA FROM RESPONDENTS ANSWERING ALL THREE SETS OF LIKERT-TYPE STATEMENTS IN 2007 SURVEY

	Formal usability testing			Informal usability testing			Agile development process		
	n	mean	SD	*n*	mean	SD	*n*	mean	SD
Usability testing/agile development . . .									
Should be used more often/should continue to be used*	46	4.48	.55	46	4.26	.58	47	4.32	.96
Has made development teams more concerned about usability**	46	4.13	.93	47	3.94	.82	43	3.58	1.01
Has not been worth the effort***	48	4.42	.96	45	4.29	.92	47	4.30	1.12
Has improved ease-of-use of products tested****	48	4.21	.82	48	3.98	.77	46	3.61	.95
Has improved usefulness of products tested*****	48	4.13	.79	46	4.07	.80	46	3.76	1.02
Has made development teams more customer-centered	48	3.92	.96	47	3.70	.91	45	3.67	.91
Has hurt cost-effectiveness***	38	4.21	.88	36	4.17	.74	41	3.88	.19

TABLE 5: DESCRIPTIVE DATA FROM RESPONDENTS ANSWERING ALL THREE SETS OF LIKERT-TYPE STATEMENTS IN 2007 SURVEY (CONTINUED)

	Formal usability testing			Informal usability testing			Agile development process		
	n	mean	SD	*n*	mean	SD	*n*	mean	SD
Has slowed down product development* **	44	3.27	1.07	44	3.25	1.06	46	3.67	1.12
Has made extra work for developers* **	43	3.12	1.1	41	3.02	1.06	44	3.27	1.17

Note: Statements listed in same order as in TABLE 1, with mean based on 1–5 scale; 5 = most positive toward usability testing. A total of 48 respondents answered all three sets of Likert-type questions; however, the paired-group t tests dropped from the analysis any cases with missing values, which were sometimes numerous because respondents could choose an answer not on the Likert-scale: no opinion/don't know. *Thus, the* n *for each matched pair analyzed varied from 29 to 47, which also resulted in variations of the means and standard deviations (SD) for each pair analyzed. These variations are reported below.*

*Significantly different means for this statement, two groups paired t-test: formal usability testing and informal usability testing (df = 43, P = 0.033).

**Significantly different means for this statement, two groups paired t-test: formal usability testing and agile process (df = 40, P = 0.006).

***Negative statement, so responses were reverse coded: a higher mean reflects more answers on the side of the scale considered to be positive toward the focus of the question.

****Significantly different means for this statement, two groups paired t-test: formal usability testing and agile process (df = 45, P = 0.004), informal usability testing and agile process (df = 43, P = 0.035).

*****Significantly different means for this statement, two groups paired t-test: formal usability testing and agile process (df = 45, P = 0.023)

were still quite positive toward both. We were somewhat surprised that perceptions about the benefits of formal usability testing remained so strong, because at the time of the survey the company had not conducted a formal usability test in almost a year. We were even more surprised that the data showed no significant differences in opinions about the two types of usability testing and agile on the statements about cost, time, and effort.

Responses to open-ended questions. As in the 2004 survey, the qualitative responses in the 2007 survey helped us understand the opinions and attitudes reflected in the quantitative data. Recall that in the first survey a question eliciting opinions about usability testing collected eight times more positive statements than negative. On a similar question about formal usability testing in the second survey, the results were even more overwhelmingly positive, as shown in Table 6 (which compares directly to the data summarized in Table 3). We categorized all but 1 of the 58 responses as being positive toward formal usability testing.

In Table 6, as in Table 3, the two categories listed at the bottom contain the comments we interpreted to be the most strongly positive toward usability testing. For the 2004 survey, 52% of the comments were sorted into those two categories; for the 2007 survey, that proportion increased to 66%. In the 2007 survey, we felt compelled to add a category to those used in analyzing the data from the 2004 survey: "positive, qualified." We sorted eight comments into that category (14% of the total) because, although they were positive toward formal testing, they emphasized some caveat about formal testing, as exampled by the two representative comments in Table 6.

The 2007 survey also presented respondents with a similar open-ended question about informal usability testing, and 38 respondents expressed an opinion. This group of respondents was not nearly as positive about informal testing as the group that commented on formal testing (compare the results in Table 7 with those in Table 6).

Slightly fewer than one half the respondents commenting on informal testing (18, or 48%) offered comments that we put into the two categories most strongly supportive of informal testing: "strongly positive" and "positive, wants more." That is quite a fall-off in enthusiasm compared with the parallel data from the open-ended question about for-

TABLE 6: SAMPLE OF 58 COMMENTS ABOUT FORMAL USABILITY TESTING IN AGILE DEVELOPMENT (2007 SURVEY)

Mildly negative *n* = 1 (2%)	This is a developing company which needs to continue to change with the changes in the industry. This is going to require faster turnaround on products and traditional time to market windows are no longer applicable, meaning [the company] has to streamline all development and testing procedures to react to the faster paced market trends.
Generally positive *n* = 11 (19%)	Seems like the products have improved as a result of usability testing, but I don't know for sure what features can be attributed to this. Usability testing is clearly a benefit. There is no strong evidence what impact it is having at [our company] or for our products. Still, it is a very good thing.
Positive, qualified *n* = 8 (14%)	Formal usability testing should be used when we have a major release which is presenting new user interface, new workflow, or new installation sequence. We also need a way to get user feedback into the early design phase of a project, and I'm not sure formal usability testing is flexible enough to work in this case. In my opinion, formal usability is good; however, the end result is not much different from that of an informal usability test.
Strongly positive *n* = 23 (40%)	Speaking from a tech support perspective, customer feedback has been the main driving force in design. Sitting people in front of our software and having them figure out how to navigate has opened our eyes to many blind spots. The idea of usability testing is an excellent one and should be used more regularly with more focused test areas (more often and smaller). To quote a colleague, "Anyone can write a network app, what distinguishes a decent app from a great app is usability". I think that it has probably paid a large part in the increased user-friendly environments of the products tested. I would like though to see more information regarding the testing made available for the employees to read.
Positive, wants more *n* = 15 (26%)	[The company] doesn't do enough testing. We should be doing UT [usability testing] at regular intervals instead of doing it once and thinking it is enough. We should be getting feedback in specific areas of interest, making changes then going back and getting feedback on the changes. Useful—should do more, with correctly selected set of users. I think that usability testing is very important, but I do not believe that has made a large impact on the developers making things work easier for the user. Functionality seems to still be the major factor for the product. Customer usability is just something that is done if enough people complain about it.

TABLE 7: SAMPLE OF 38 OPEN-ENDED ANSWERS ABOUT INFORMAL USABILITY TESTING IN AGILE DEVELOPMENT (2007 SURVEY)

Negative *n* = 3 (8%)	Only done because we do not have time to do formal usability testing due to poor planning. Nearly worthless. Our sample selection is completely biased. Seem to cost less but at some risk of quality of the result.
Generally positive *n* = 6 (16%)	Costs money and time but saves both in the long run. Quick and effective.
Positive, qualified *n* = 11 (29%)	The “customers” aren’t represented as well, but it is still a good exercise that is helpful. Informal usability testing can be beneficial if done properly. I think we have to be very careful about the people we choose to participate and the scenarios we give them. They often have come from within the company and may already have pre-conceived notions about the application and/or functionality being tested.
Strongly positive *n* = 12 (32%)	There is great value to be obtained from informal usability testing. I view it as a continuum between formal and informal. If we structure informal testing to allow it to be fairly spontaneous, and train people how to conduct it, there is much we can learn that will quickly translate into better products. Perfect fit for agile!
Positive, wants more *n* = 6 (16%)	We need to take more advantage of informal testing. We should be getting more people in the system to use [it] at various intervals of product development. I don’t think we do enough informal usability testing, and I think we’re neglecting vital internal resources (IT staff, for one) who can provide excellent feedback with short response/overhead times. However, we need to ensure that any testing, "informal" as it may be, is planned and executed by an individual or team with strong usability skills.

mal testing (Table 6), which elicited strongly positive comments from two of three who answered. The dip in enthusiasm seems linked to a rise in caveats about informal testing; the proportion of positive comments about informal testing that contained hedging statements citing some condition or qualification was twice as great compared with the comments on formal testing: 29% for informal testing *versus* 14% for formal testing. Finally, the only negative comments about informal testing stemmed from the respondents’ high regard for formal testing, which they saw as having been de-emphasized to some extent because of informal testing.

Like the quantitative data summarized in Table 5, the qualitative data summarized in Tables 6 and 7 suggest that the attitudes inside the company were more supportive of formal usability testing than of informal testing. However, that conclusion is undermined by the comments responding to the final three open-ended questions on the survey: one asking about impacts, if any, on usability testing in the company; one asking for a summary of opinions about agile development as implemented by the company; and one inviting suggestions for how the company should change how the development teams worked.

The comments from the final two questions were relatively easy to analyze for our purposes because usability testing was rarely mentioned. Of the 55 respondents who offered comments summing up their opinion about the company's implementation of agile methods:

- 3 (5%) gave no opinion, stating neutrality or lack of knowledge
- 7 (13%) expressed a negative opinion
- 16 (29%) were strongly positive
- 19 (35%) were generally positive
- 10 (18%) were positive in a qualified way

In all, 4 of 5 were positive about agile development, although some of them noted that the company's implementation needed to be improved; of the 10 respondents citing caveats, however, only 1 mentioned a negative impact on usability.

The final question inviting suggestions for improving the development process was more positive with regard to usability. The question garnered 32 serious suggestions for needed improvements, which were put into general categories as follows:

- 6 (19%) wanted more emphasis on usability testing
- 26 (81%) wanted to improve agile execution in various ways
- 7 (22%) wanted better integration of teams and communication among them
- 6 (19%) wanted miscellaneous improvements
- 4 (13%) wanted more customer input and interaction
- 3 (9%) wanted better adherence to schedules
- 2 (6%) wanted better long-range plans
- 2 (6%) wanted more emphasis on user-interface design
- 2 (6%) wanted better quality assurance testing

To summarize, about one in five suggestions for improving the agile development process called for doing a better job of incorporating usability testing. By far, most respondents to this question focused on various other "nuts and bolts" of agile implementation.

In the 2007 survey, this was the "bottom line" question: *What impact, if any, has agile development at [the company] had on the use of both formal and informal usability testing? Please explain, with specific examples if possible.* Forty-four respondents provided comments, but two simply stated they had no opinion and another six said they did not know. Seven comments suggest that the respondents misread the question because six offered positive statements about agile without mentioning usability testing, and one offered a general positive statement about usability testing without mentioning agile. We categorized the 29 analyzable and relevant comments as follows:

- 4 (14%) said that agile had had no impact on usability testing
- 1 (3%) suggested that maybe agile had led to less usability emphasis
- 1 (3%) said agile's impact had been positive for both forms of testing
- 2 (7%) said agile's impact had been positive specifically for formal testing
- 6 (21%) said agile's impact had been positive for usability testing, not specifying which type
- 2 (7%) stated in a neutral way that agile had led to more informal testing
- 4 (14%) stated negatively that agile had led to more informal testing
- 9 (31%) stated approvingly that agile had led to more informal testing

To sum up more generally, 18 of these 29 respondents (62%) expressed the view that agile had had a positive impact on one or both forms of usability testing; only 5 expressed a negative view of the impact, with 4 believing that informal testing was being used, ineffectively, in place of formal testing, and 1 expressing the tentative opinion that agile had possibly led to less emphasis on usability in general.

We conclude that the responses to the final question about the impact of agile on both forms of testing drew responses from a subset of respondents who, as a group, expressed a higher regard for informal than for formal usability testing. This was just the reverse of what the data from the larger group of respondents tell us—the group answering all three sets of Likert-type questions. The statistical findings summarized in Table 5 and the open-ended comments summarized in Tables 6 and 7 lead us to conclude that the larger subset of respondents viewed formal usability testing more favorably than informal testing. We think that the final open-ended questions, and the last one in particular, elicited comments mainly from respondents wishing to defend the use of agile and informal usability testing from what many of them would have perceived at that point as the survey's implied hypothesis: that adoption of an agile process had led to less emphasis on and concern for formal usability testing.

Conclusions from the 2007 survey

We interpret the data from the Likert-type statements and the open-ended comments on the 2007 survey to mean that the dominant opinion about agile's impact on usability testing at this company can be summed up this way: the company cares deeply about the usability of its products and thus believes that usability testing is vital to its development process. However, informal usability testing and other informal methods of gathering user feedback are perceived by many to fit better with the agile process and to be potentially just as effective as formal testing.

Although a handful of respondents from the development teams felt that knowledge of the user experience was being sacrificed on the altar of agility, the majority believed that informal testing was a good match for the agile process. Although the desire to continue using both types of usability testing was apparently as strong, if not stronger, than the desire to continue using agile, the preponderant patterns in the comments from respondents led us to predict that the trend of the past few years would continue: The company would use formal usability testing much less frequently than it had when they were using a waterfall development process.

In November 2007, one of us (Dayton) made a site visit to one of the company's product development teams. A long conversation with two managers focused on details of the agile process and how the team had been conducting informal testing. The managers did not feel that their product's usability had suffered because of the agile process, which they credited with enabling them to get new features and functionality to market more quickly. They said that sales were better than ever.

Since that visit, we have heard from the managers that the team is giving more time to usability testing, which is still being conducted informally but reportedly with more rigor and regularity—and with more attention to the findings. As of this writing (January 2009), the company has not, to our knowledge, resumed formal usability testing.

RECOMMENDATIONS: MAKING USABILITY METHODS MORE AGILE

What we have learned from our experience with this company and from reading reports and commentary by usability professionals who have been through similar experiences leads us to conclude that making formal usability methods more agile presents a difficult challenge. The biggest stumbling block to incorporating usability testing into an agile process would seem to be planning ahead, which must begin with the effort to locate and recruit a pool of real users well ahead of time so that teams can quickly schedule tests with users who authentically represent the product's customers. One company, Autodesk, describes how they have successfully done this by having two parallel tracks—the interaction designer track and the developer track—work simultaneously on sprints (Sy 2007). With the user experience team as part of the interaction design track, the user-experience members can plan with and ahead of the developers to provide viable feedback on the user experience as the product is being developed. This process works by getting ahead of designers by at least one cycle, as well as gathering requirements before cycle 0. Scheduling for usability testing or site visits also has to be planned ahead. Not only does Autodesk look ahead to do usability testing on the next cycle of a product, but the team also looks even farther ahead to do contextual inquiry for the cycle 0 (the planning cycle) of the next release. Autodesk also takes paper prototypes of future features in early development with them on site visits for contextual inquiry and/or usability testing.

The challenge of planning ahead has earlier been described as one of the major obstacles to technical communicators working in agile environments. Added to this challenge is the need to plan ahead for usability testing, so that technical communicators, as the traditional user advocate, can also take on the role of advocating for user testing in agile development teams. Such has been the role of the technical communicators at the software company we have described. However, their voice was not sufficiently loud (or strong) to be heard.

Change in the workplace must come from within, and it has to be championed by someone with the authority to make it happen. We offer several suggestions to push for this change, if it is not already a part of your agile process:

- ◆ Make the case for knowledge of the user experience first-hand to the manager or to the scrum team.
- ◆ Push for training to understand the agile methodology; request that whoever provides the training must understand the significance of usability testing within or between sprint cycles.
- ◆ Plan ahead: with 2- to 4-wk sprints, it is easy to see why usability testing doesn't get picked as a card on the wall; however, planning out (from cycle 0) to touch points where features will come together provides a process to build usability testing into the development cycle.
- ◆ Use rapid, iterative testing and evaluation processes, such as the RITE method developed by Microsoft (Medlock and colleagues 2002) to become more agile and get very fast feedback from users (using this process provides a means to see a very few number of real users, with changes made as soon as a problem is identified).
- ◆ Schedule usability testing, regularly, and routinely; one company (Kaiser) developed a weekly usability test schedule, in which sprint teams can sign up for testing in any given week, knowing that the users are already scheduled (Kieschnick 2008).

If there is not already a designated user advocate, you can take on this role to educate the development team on how to incorporate UCD processes for creating rich user stories (a critical component of an agile process), persona creation, and iterative usability testing, so that the results are valid and can be believed.

If you do not have the expertise, push to hire a user-experience expert, who can serve as a team member and train your team and others on how to build user experience into product development.

Taking these steps means that there will be more focus on requirements gathering about users and contexts of use, more accurate personas and user stories resulting from requirements gathering, early prototype testing to get feedback on user-interface concepts, improved processes and increased quality of informal usability testing, and a planned place in the sprint cycles for informal and formal usability testing. TC

REFERENCES

Barnum, C., D. Dayton, K. Gillis, and J. O'Connor. 2005. Making connections—teaming up to connect users, developers, and usability experts. G. Hayhoe and H. Grady (Eds.). In *Proceedings of the 2005 IEEE international professional communication conference* (Limerick, Ireland, July 10–13, 2005). Piscataway, NJ: IEEE, pp. 108–114.

Fox, A., and M. Kramer. 2008. Mobile and agile: The floating writer's survival kit. Presentation (slides) at 55th Annual Conference—Technical Communication Summit, Philadelphia, PA. http://www.stc.org/edu/55thConf/index.asp.

Frick, T., E. Boling, K.-J Kim, D. Oswald, W. Sugar, and T. Zazelenchuk. 2001. Software developers' attitudes toward user-centered design, 2000. M. Crawford, M. Simonson, and C. Lamboy (Eds.). In *2001 Annual proceedings of selected research and development [and] practice papers presented at the national convention of the Association for Educational Communications and Technology* (24th, Atlanta, GA, November 8-12, 2001) 1-2:140–146.

Gunther, R., J. Janis, and S. Butler. 2001. The UCD decision matrix: How, when, and where to sell user-centered design into the development cycle. http://www.ovostudios.com/upa2001/.

Hudson, W. 2003. Adopting user-centered design within an agile process: A conversation. http://www.syntagm.co.uk/design/articles/ucd-xp03.pdf.

Kieschnick, T. 2008. User research as a commodity (part 3 of 7). Tim's musings. http://timiti.blogspot.com/2008/02/user-research-as-commodity.html.

MacAller, J. 2008. Scrum: An agile approach to managing content projects. Presentation (slides) at 55th Annual Conference—Technical Communication Summit, Philadelphia, PA. http://www.stc.org/edu/55thConf/index.asp.

Mao, J., K. Vredenburg, P. W. Smith, and T. Carey. 2005. The state of user-centered design practice. *Communications of the ACM* 48:105–109.

Mazet, J-L. 2008. Agile technical documentation. Presentation (slides) at 55th Annual Conference—Technical Communication Summit. Philadelphia, PA. http://www.stc.org/edu/55thConf/index.asp.

McInerney, P., and F. Maurer. 2005. UCD in agile projects: Dream team or odd couple?. *Interactions* 12:19–23.

Medlock, M. C., D. Wixon, M. Terrano, R. Romero, and B. Fulton (2002). Using the RITE Method to improve products: A definition and a case study. Proceedings of the usability professionals' association conference. http://www.microsoft.com/downloads/ (use RITE to search).

Miller, L. 2006. Interaction designers and agile development: A partnership. Paper presented at the Usability Professionals' Association Conference, 12–16 June, in Broomfield, CO.

O'Connor, V. 2007. Agile development: Challenges and opportunities. *Intercom* 54:16–18.

Patton, J. 2002. Hitting the target: adding interaction design to agile software development. M. Ibrahim (Ed.). In *OOPSLA 2002 practitioners reports: Proceedings of the conference on object oriented programming systems languages and applications* (Seattle, WA, November 4-8, 2002). New York: ACM, pp. 1-ff.

Rosenbaum, S., J. A. Rohn, and J. Humburg. 2000. A toolkit for strategic usability: results from workshops, panels, and surveys. T. Turner and G. Szwillus (Eds.). In *Proceedings of the SIGCHI conference on human factors in computing systems* (The Hague, The Netherlands, April 1-6, 2000). New York: ACM, pp. 337-344.

Sigman, C. M. 2007. Adapting to scrum: Challenges and strategies. *Intercom* 54:16–19.

Sy, D. 2007. Adapting usability investigations for agile user-centered design. *Journal of usability studies* 2:112–132.

Taliaferro, T. 2006. Agile and doc. Paper presented at the 53rd annual STC conference, 3 May, in Las Vegas, NV.

Venturi, G. and J. Troost. 2004. Survey on the UCD integration in the industry. R. Raisamo (Ed.). In *Proceedings of the third Nordic conference on human-computer interaction* (Tampere, Finland, October 23-27, 2004). New York: ACM, pp. 449-452.

Wethington, M. 2008. Agile development: Challenges in transforming technical communication departments. Presentation (slides) at 55th Annual Conference—Technical Communication Summit, Philadelphia, PA. http://www.stc.org/edu/55thConf/index.asp.

Wikipedia: The free encyclopedia. s.v. “Agile Software Development.” http://en.wikipedia.org/wiki/Agile_software_development.

CAROL BARNUM is Director and Co-Founder of the Usability Center and Professor of Information Design at Southern Polytechnic State University, where she directs the graduate programs in information design and communication. The author of *Usability testing and research* (2002), she is an STC Fellow and recipient of the STC Gould Award for Excellence in Teaching Technical Communication. Contact: cbarnum@spsu.edu.

DAVID DAYTON was an assistant professor of technical communication and an associate of the Usability Center at Southern Polytechnic State University in suburban Atlanta when this study began. He is now an assistant professor of technical writing and information design in the English Department at Towson University (Baltimore County, MD). He has worked in technical communication since 1990 as a practitioner, consultant, educator, and researcher. In 2007–2008, he co-chaired the STC Technical Communication Body of Knowledge Task Force. Contact: dr.david.dayton@gmail.com.