JUS
JOURNAL OF USABILITY STUDIES

Vol. 3, Issue 1, November 2007, pp. 7-23

A Structured Process for Transforming Usability Data into Usability Information

Jonathan Howarth
Virginia Tech
111 James Jackson Ave
Suite 221
Cary, NC 27513
jhowarth@vt.edu

Terence S. Andre
Aptima, Inc.
3100 Presidential Drive
Fairborn, OH 45324
tandre@aptima.com

Rex Hartson
Virginia Tech
2050 Torgersen Hall (0106)
Blacksburg, VA 24061
hartson@vt.edu

Abstract

Much research has been devoted to developing usability evaluation methods that are used in evaluating interaction designs. More recently, however, research has shifted away from evaluation methods and comparisons of evaluation methods to issues of how to use the raw usability data generated by these methods. Associated with this focus is the assumption that the transformation of the raw usability data into usability information is relatively straightforward. We would argue that this assumption is incorrect, especially for novice usability practitioners. In this article, we present a structured process for transforming raw usability data into usability information that is based on a new way of thinking about usability problem data. The results of a study of this structured process indicate that it helps improve the effectiveness of novice usability practitioners.

Keywords

Usability evaluation, usability problem instances, usability engineering tools, empirical findings, usability evaluation methods

Introduction

Much research has been devoted to developing usability evaluation methods that are used in evaluations of interaction designs. Example usability evaluation methods include cognitive walkthroughs (Polson et al., 1992), heuristic evaluations (Nielsen, 1994), remote usability evaluation methods (Castillo et al., 1998), and lab-based usability testing (Hix & Hartson, 1993). The focus of these evaluation methods is the collection of usability problem (UP) data.

More recently, however, research has shifted away from evaluation methods and comparisons of evaluation methods to issues of how to use the UP data generated by methods. Wixon (2003), for example, discusses issues that are important in actually fixing UPs, such as resource limitations and contextual factors. Hornbæk and Frøkjær (2005) also take a practical perspective and discuss the effectiveness of redesign proposals to accompany UP descriptions; these proposals describe how to improve the interaction design to minimize or eliminate the flaws that result in UPs for users. Additionally, Theofanus (2005) describes which elements to include in a formative usability evaluation report to best meet the needs of a client.

This new focus on fixing UPs is appropriate given the continued maturation of the usability engineering discipline. Associated with this focus, however, is the assumption that transforming the raw usability data generated by evaluation methods into usability information is relatively straightforward. We would argue that this assumption is not practical in interaction design applications today.

One reason is the presence of the evaluator effect. Previous research has confirmed the tendency of usability practitioners to find different types and numbers of UPs during usability evaluations (Jacobsen et al., 1998). There is variation in what usability practitioners identify as being important, especially given vastly different experience levels.

A second reason is that usability data may exist in a variety of forms such as notes, video, audio, and textual critical-incident descriptions and may come from a variety of sources such as self-reports by remote users, usability lab testing data, and inspections performed by usability experts. As such, it may be difficult for a usability practitioner to make meaning consistently out of the data and recognize UPs.

A third reason is that the same UP may be experienced by multiple participants or multiple times by one participant and may be described at different levels of abstraction (Cockton & Lavery, 1999). For example, one UP description may focus on a problem with the labeling of a specific menu item, while another UP description may deal with the same problem with the labeling of all menu items.

A fourth reason is the lack of consensus among usability researchers regarding how to describe a UP. There have been a number of UP definitions and UP description formats described in the literature, which vary to differing degrees (Cockton et al., 2004, Lavery et al., 1997). Additionally, recent work has minimally addressed and sometimes avoided the issue of how to describe UPs. For example, the instructions for usability practitioners in the fourth Comparative Usability Evaluation (Dumas et al., 2004) specified that UPs should have descriptions, but did not specify what to include in the descriptions.

Lund states that usability engineering "needs to grow as a science and engineering discipline based on research and at least as importantly theory" (2006). In line with Lund's article, we argue that there is a need for a more structured approach to transforming raw usability data into usability information. Currently, this transformation is more of an art than a process and is highly dependent on the researcher’s skill and experience.

In this article, we present a structured process for transforming raw usability data into usability information that is based on a new way of thinking about usability problem data. We then describe a study that we conducted to evaluate this structured process. We conclude with a discussion of the implications and limitations of the study.

Structured Process for Transforming Raw Usability Data into Usability Information

Many sources describe the usability evaluation process using a variety of techniques and methods; for examples the reader is referred to Hix and Hartson (1993). All usability evaluation processes whether they use empirical or analytical techniques have three basic stages: usability data collection, UP analysis, and usability evaluation reporting.

The ultimate goal of the usability evaluation process is to transform raw usability data into usability information that can be used to improve an interaction design. Current approaches (Figure 1) rely on the expertise of usability practitioners to extract UPs from the raw usability data in the usability problem analysis stage. The extraction of UPs, however, is not straightforward. Raw usability data are typically very specific and detailed, while usability problems are necessarily general. The gap between raw usability data and usability problems is quite significant and is replete with substantial individual variation. As an example, consider a usability evaluation of a banking website that does not provide adequate feedback concerning the results of transfers and other transactions. The raw usability data might include comments on delays, inappropriate action sequences, or puzzled remarks by participants. The relationship between these comments and the overall problem of inadequate feedback may not be immediately understandable, particularly to novice usability practitioners.

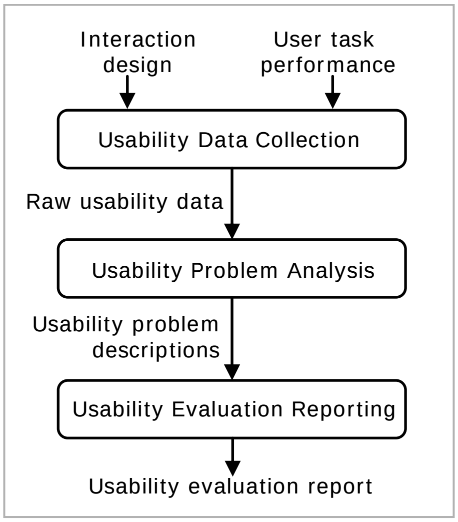

Interaction design
User task performance
Usability Data Collection
Raw usability data
Usability Problem Analysis
Usability problem descriptions
Usability Evaluation Reporting
Usability evaluation report

Figure 1: Current usability evaluation process

We introduce the concept of UP instances to serve as a bridge between raw usability data and usability problems. Each occurrence of a UP as encountered by a participant and observed by the evaluator is a UP instance. The same UP may be experienced by multiple participants or multiple times by one participant. For example, consider a usability evaluation of an image manipulation program in which the handle for adjusting the size of shapes and images is too small. One UP is related to physical actions or the ability of the participants to click on or drag the handle. A new UP instance is used to document each time each participant has difficulty clicking on or dragging the handle. Each UP instance may involve a different context; one participant may encounter the problem while trying to resize a shape, while another may have trouble cropping an image.

A study by Howarth (2007) suggests that working at the UP instance level of abstraction as opposed to working with raw usability data can help facilitate the understanding and relating of usability data. Howarth had two groups of novice usability practitioners watch a video of users performing tasks in a course management application and create records of UP instances. One group of novice usability practitioners used a commercial usability engineering tool that only supported the collection and analysis of raw usability data. The other group used a tool developed for the study that only supported the collection and analysis of UP instances. The group using the tool that required them to work at the UP instance level of abstraction was more reliable in terms of the UP instances identified. Additionally, the descriptions of the UP instances were of higher quality.

UP instances form the basis for our structured usability evaluation process for translating raw usability data into usability information (Figure 2). This process includes the identification of UP instances in the usability data collection stage. The usability practitioner produces brief UP instance records that contain just enough information to describe the UP instance.

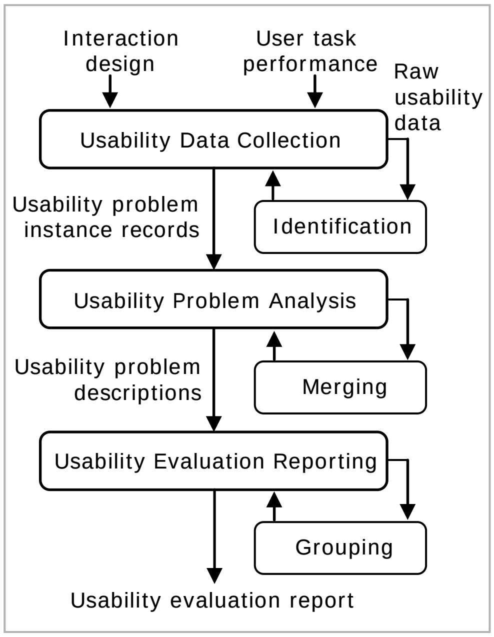

Interaction design
User task performance
Raw usability data
Usability Data Collection
Identification
Usability problem instance records
Usability Problem Analysis
Merging
Usability problem descriptions
Usability Evaluation Reporting
Grouping
Usability evaluation report

Figure 2: Structured usability evaluation process

During the UP analysis stage, the usability practitioner fills in the UP instance records from the usability data collection stage with more details as necessary. The usability practitioner then merges the UP instance records. Merging involves combining UP instances that map to the same UP. The UP description includes what the evaluator predicts as the effect that an interaction design flaw has on the user.

In the usability evaluation reporting stage, the usability practitioner uses the UP descriptions generated during the UP analysis stage to create usability evaluation reports to guide subsequent fixing of the interaction design. Grouping involves associating UPs in a manner that is most appropriate for the target audience of the usability evaluation report. For example, developers may want to know specific areas of an interface that are involved in a UP, while managers may want an executive summary of an interaction design's strengths and weaknesses.

Figure 3 shows a concrete example of using the process to transform raw usability data. The figure shows the transformation of usability data collected during a lab-based evaluation of an online photo album application. A user is expected to first create an album and then upload pictures to store in the album. The user can design pages in the album. Each page has two view modes: organize and edit. The organize mode is used to arrange pictures, while the edit mode is used to edit a photo in place. Comments C1 to C5 are combined into UP instance UPI1, and comments C6 and C7 are combined into UP instance UPI2. Both UPI1 and UPI2 are instances of the same UP, so they are merged to form UP1. Additional UP instances from later in the evaluation are merged to form UP2, which is then grouped for reporting purposes with UP1 to form G1.

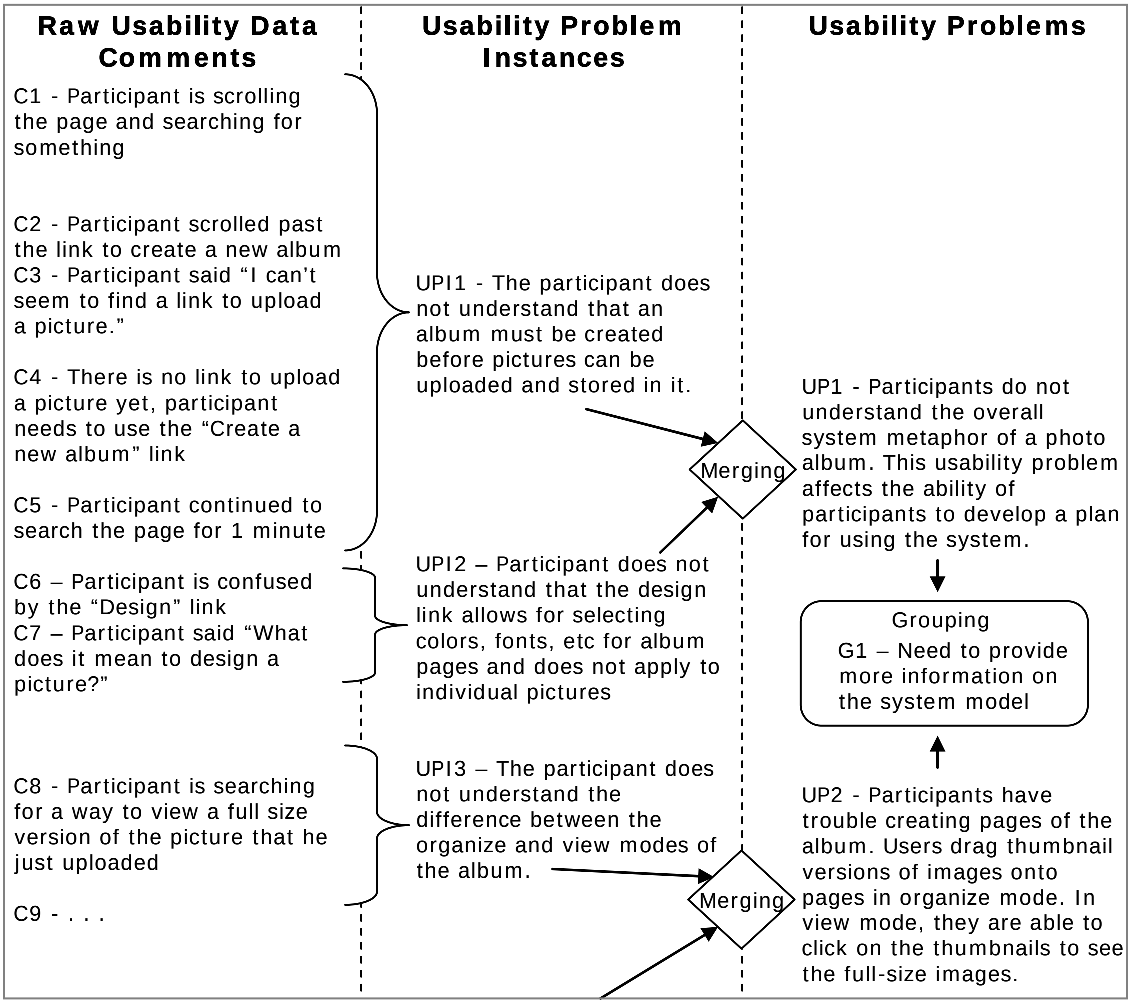

Raw Usability Data Comments
Usability Problem Instances
Usability Problems
C1 - Participant is scrolling the page and searching for something
C2 - Participant scrolled past the link to create a new album
C3 - Participant said "I can't seem to find a link to upload a picture."
C4 - There is no link to upload a picture yet, participant needs to use the "Create a new album" link
C5 - Participant continued to search the page for 1 minute
C6 – Participant is confused by the "Design" link
C7 – Participant said "What does it mean to design a picture?"
C8 - Participant is searching for a way to view a full size version of the picture that he just uploaded
C9 - . . .
UPI1 - The participant does not understand that an album must be created before pictures can be uploaded and stored in it.
UPI2 – Participant does not understand that the design link allows for selecting colors, fonts, etc for album pages and does not apply to individual pictures
UPI3 – The participant does not understand the difference between the organize and view modes of the album.
Merging
Merging
UP1 - Participants do not understand the overall system metaphor of a photo album. This usability problem affects the ability of participants to develop a plan for using the system.
Grouping
G1 – Need to provide more information on the system model
UP2 - Participants have trouble creating pages of the album. Users drag thumbnail versions of images onto pages in organize mode. In view mode, they are able to click on the thumbnails to see the full-size images.

Figure 3: An example of using the structured process to transform usability data from an online photo album application. The vertical axis represents time

Methods

We conducted a study to evaluate how support for a structured process for transforming usability data affects the effectiveness of usability practitioners. We define effectiveness as the accuracy and completeness with which a usability practitioner can produce usability evaluation reports; this definition is based on the ISO definition (1998).

The participants in this study watched videos of representative users performing tasks with Scholar, a course management system. These participants, whom we refer to as evaluators, produced usability evaluation reports using one of two usability engineering tools. We recorded time data while the evaluators created their usability evaluation reports. Individuals with usability experience, whom we refer to as judges, rated the usability evaluation reports from the perspective of a usability practitioner to create measures of quality. The developers of Scholar also rated the usability evaluation reports to create additional measures of quality.

In experimental design terms, this study is a between-subjects design with support for the structured process (no support = freeform, support = structured) as the independent variable. The dependent variables were time measures and measures of usability evaluation report quality as rated by judges and by developers.

Effectiveness was of primary interest for this study. We did, however, assume a fixed-resources environment in an attempt to simulate a real world effort where people and time resources are relatively fixed. We recorded the amount of time that it took the evaluators to perform the evaluations to confirm this operating assumption.

Participants

As mentioned in the overview for this study, the participants are referred to as evaluators. The evaluators in this study were not experienced usability practitioners. We made a conscious decision to focus on novice usability practitioners. The literature suggests that skill plays an important part in usability evaluation. For example, a study by Nielsen (1992) found that usability specialists were better than non-specialists at using heuristic evaluation to evaluate an interface. Also, in a study comparing the iterative development of designs by human factors specialists and programmers, Bailey (1993) concludes that “the training and background of designers can have a large effect on user interface design”. Experienced usability practitioners typically have developed methods and strategies that work for them. Novice usability practitioners, on the other hand, may interpret data incorrectly or fail to recognize important usability data without the guidance and support that can be provided by a structured process. As a result, novices stand to gain more in terms of effectiveness.

Sixteen evaluators participated in this study, eight in each treatment. All the evaluators were Virginia Tech graduate students who had one or more of the following qualifications:

- Had completed or were enrolled in a usability engineering or human-computer interaction course
- Had usability engineering research experience

Additionally, all the evaluators selected for the study had less than one year of job experience related to usability engineering, thereby qualifying them as novices.

Evaluators were recruited from university mailing lists. Fourteen of the participants were students in the Department of Computer Science and two were students in the Department of Electrical and Computer Engineering. Fourteen had experience with systems similar to Scholar, but none had ever used Scholar.

Materials

Videos of Representative User Sessions

Evaluators in this study watched videos of two representative users performing tasks in Scholar, Virginia Tech's adaptation of an open system called Sakai (http://www.sakaiproject.org/). The total length of all three videos combined was approximately 12 minutes. One representative user performed the tasks of adding a student to and removing a student from a course, and the other representative user performed the task of adding a student to a course. The videos were selected from screen action video and audio recordings that we made of five representative users during an earlier usability evaluation of Scholar (Figure 4).

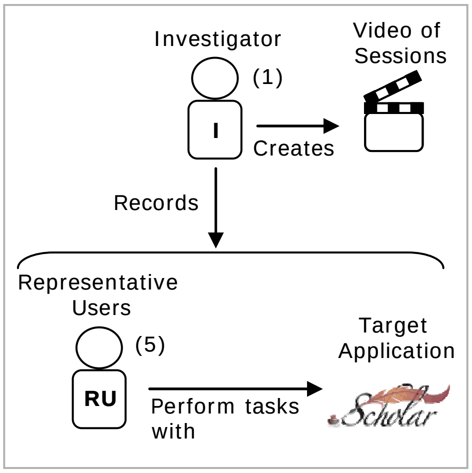
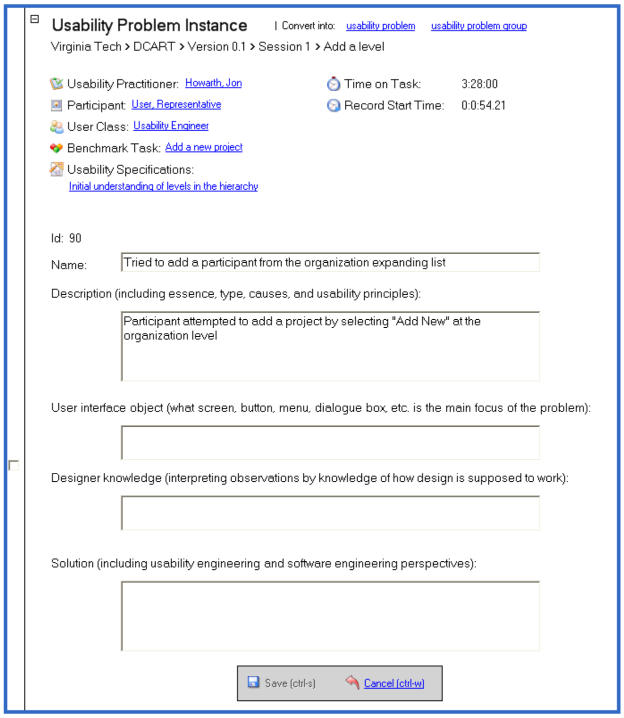


Figure 4: Representative user's role; the numbers in parentheses indicate the number of individuals in each role

Usability Engineering Tools

Evaluators used two different usability engineering tools in this study. Evaluators in the treatment without explicit support for the structured process used Morae (TechSmith), a usability recording tool that allows evaluators to capture screen video, user video, and user audio in an integrated digital file with markers that indicate comments. Evaluators in the treatment with explicit support for the structured process used the Data Collection, Analysis, and Reporting Tool (DCART), which we developed for the study. DCART structures the process of recording and analyzing usability data to create usability information.



Usability Problem Instance | Convert into: usability problem usability problem group

Virginia Tech > DCART > Version 0.1 > Session 1 > Add a level

Usability Practitioner: Howarth, Jon
Participant: User, Representative
User Class: Usability Engineer
Benchmark Task: Add a new project
Usability Specifications:
Initial understanding of levels in the hierarchy

Time on Task: 3:28:00
Record Start Time: 0:0:54.21

Id: 90

Name: Tried to add a participant from the organization expanding list

Description (including essence, type, causes, and usability principles):

Participant attempted to add a project by selecting "Add New" at the organization level

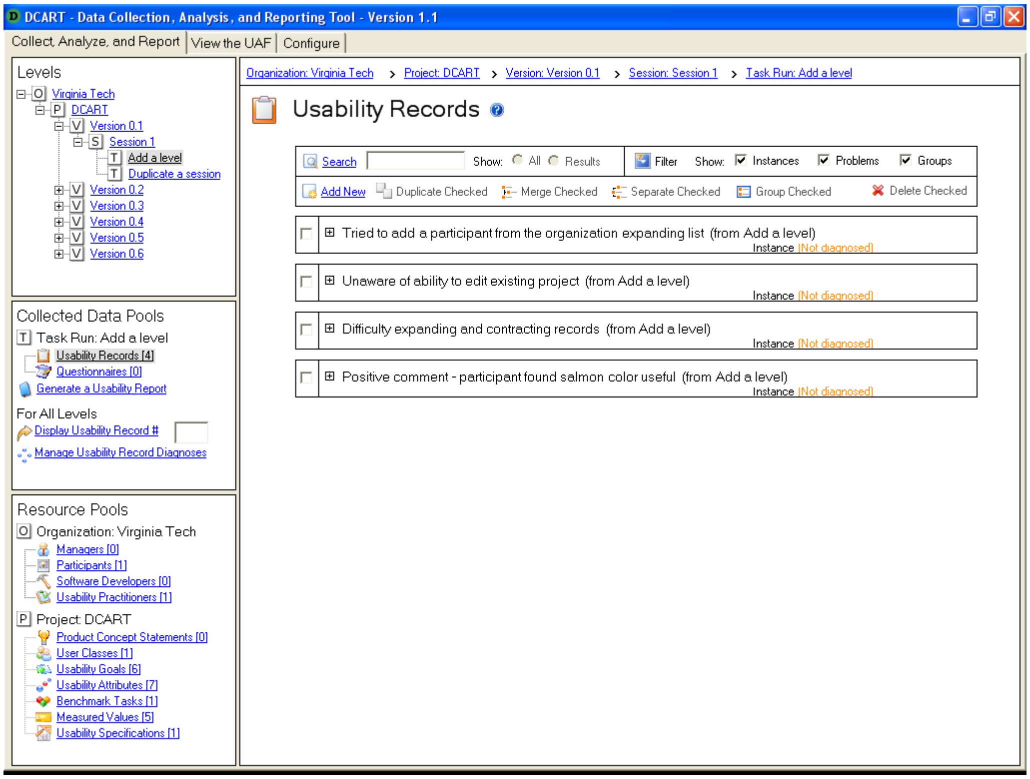
User interface object (what screen, button, menu, dialogue box, etc. is the main focus of the problem):

Designer knowledge (interpreting observations by knowledge of how design is supposed to work):

Solution (including usability engineering and software engineering perspectives):

Figure 5: Usability problem instance record in DCART

DCART is documented in detail in Howarth (2007). For illustrative purposes, however, we include two screenshots of it in this article. Figure 5 shows a sample UP instance record. The section at the top contains context information as well as information about the time at which the UP was encountered in the task run. The remainder of the record contains fields that are filled out by the evaluator. The evaluators merged and grouped these records using the view in Figure 6.



□ ⊞ Tried to add a participant from the organization expanding list (from Add a level) Instance [Not diagnosed]

□ ⊞ Unaware of ability to edit existing project (from Add a level) Instance [Not diagnosed]

□ ⊞ Difficulty expanding and contracting records (from Add a level) Instance [Not diagnosed]

□ ⊞ Positive comment - participant found salmon color useful (from Add a level) Instance [Not diagnosed]

Figure 6: DCART view for merging and grouping usability problem instances

Procedure

We filtered participants via a background survey and performed matching on basic knowledge (usability engineering or human-computer interaction), experience with systems similar to Scholar, and English language skills, so that the participants were as evenly distributed between treatments as possible. In one treatment, evaluators used Morae to conduct a usability evaluation; in the other treatment, evaluators used DCART to conduct a usability evaluation. We notified evaluators who had been selected to participate in the study via email and had them choose a date and time that was convenient for them from a list of available dates and times. Each evaluator participated in one study session that lasted no more than two and a half hours. Evaluators participated individually; each study session consisted of only one evaluator.

Regardless of the tool that they used, the evaluators followed the same basic process. During the first hour, the evaluators performed activities to familiarize themselves with their tool and the steps involved with performing a usability evaluation. During the next one and a half hours, the evaluators performed a usability evaluation of Scholar. The evaluators were told that they were to prepare usability evaluation reports for the developers of Scholar.

The evaluators began the evaluation of Scholar by watching a video that introduced Scholar, a video of a correct way to add a student to a course, and a video of a correct way to remove a student from a course. Next, the evaluators watched a video of a representative user trying to add a student, a video of a second representative user trying to add a student, and a video of the first representative user trying to remove a student. While the evaluators watched the videos of the representative users, they recorded comments (Morae) or created UP instance records (DCART). The evaluators watched the three videos one time through without pausing or stopping to simulate conducting a usability evaluation in real time. Thereafter, they were allowed to rewind, play, fast forward, pause, and stop the videos as much as they needed. The evaluators submitted usability evaluation reports as Microsoft Word documents. Figure 7 shows the tools and objects that the evaluators used and produced.

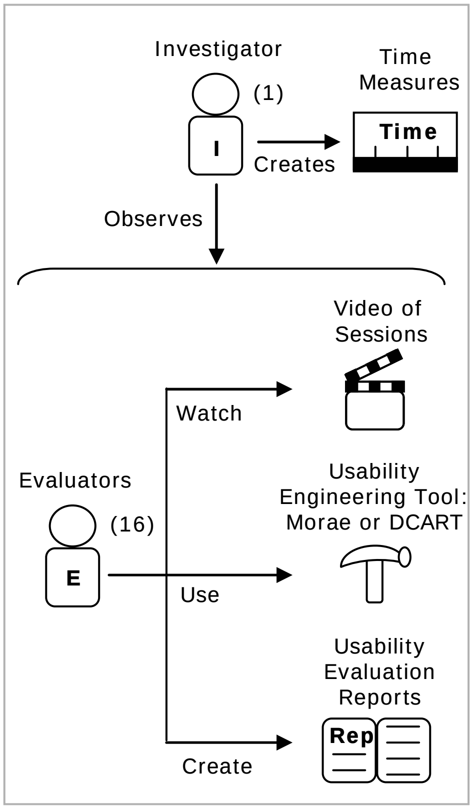


Figure 7: Evaluator's role; the numbers in parentheses indicate the number of individuals in each role

The evaluators who used Morae did not have explicit support for the structured process of translating raw usability data into usability information. These evaluators made time-stamped comments using the observational capture features of Morae Remote Viewer while they watched the videos of representative users. They reviewed their comments, added new comments, and reviewed the video using Morae Manager, an advanced playback tool that allows the evaluator search and review specific comments. They then created usability evaluation reports based on their comments.

The evaluators who used DCART created UP instance records while watching the videos. One UP instance record documented one instance of UP as experienced by a representative user. The evaluators filled out the name and description fields of the UP instance record while they watched the videos of the representative users the first time. They filled out other fields in the UP instance record after they had watched all the videos one time through; these fields were used to document the user interface object or objects associated with the UP instance, designer knowledge about how the design should work, and solution suggestions. The evaluators used built-in functions to merge UP instances that described the same UP and group related UPs. They also used a function built into DCART to generate a usability evaluation report based on the UPs and groups of UPs that they had created. All the evaluators modified the usability evaluation report generated by DCART.

Measures

Time

As described in the overview of this study, evaluator effectiveness was of primary interest for this study. Because we assumed a fixed-resources environment, however, we wanted to remove efficiency as a point of consideration. As a result, we recorded the amount of time that the evaluators spent performing the evaluation and whether evaluators finished.

Usability Problem Instance Quality As Rated By Judges

A number of steps were involved in calculating quality as rated by judges. First, we selected six guidelines developed by Capra (Accepted for publication, 2007a) for UP descriptions. Next, two individuals with usability experience, whom we refer to as judges, rated the usability evaluation reports produced by evaluators based on the guidelines from the perspective of a usability practitioner. Finally, the ratings were used as inputs to calculate a measure of quality. Figure 8 shows the objects that the judges used and produced.

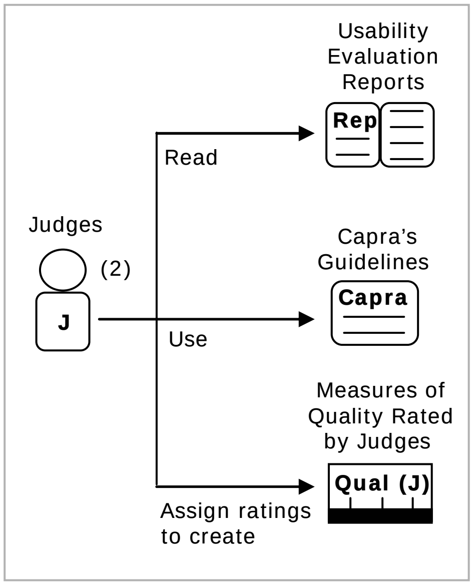


Figure 8: Judge's role; the number in parentheses indicates the number of judges

CAPRA'S GUIDELINES

Capra (Accepted for publication, 2007a) developed 10 guidelines for UP descriptions based on surveys of usability practitioners. Capra (Accepted for publication, 2007b) tested six of these guidelines in a study in which practitioners and graduate students watched the same 10-minute recording, which showed sessions with representative users of a web site, and created usability evaluation reports. Three judges rated each of the usability evaluation reports using the guidelines. The practitioners received higher ratings across all guidelines and specifically for three guidelines. Capra's work suggests that the guidelines can be applied as measures of quality of usability evaluation reports.

For this study, we included the six guidelines used by Capra (Accepted for publication, 2007b). We modified the guidelines in terms of presentation by changing them from paragraphs to bulleted lists. The guidelines are as follows:

- Be clear and precise while avoiding wordiness and jargon
- Describe the impact and severity of the problem
- Support your findings with data
- Describe the cause of the problem

- Describe observed user actions
- Describe a solution to the problem

JUDGES

We asked two professional contacts to serve as judges. One judge is a practicing usability professional, and the other judge is a doctoral computer science student with academic usability engineering and human-computer interaction experience. The judges watched the same videos of Scholar as the evaluators watched during their study sessions. The judges worked independently and viewed the evaluators' usability evaluation reports in different orders; one judge's ordering was the reverse of that of the other to balance any potential familiarization or learning effects. The judges rated each evaluator's usability evaluation report on each guideline using a 6-point Likert-type scale with the following values:

- strongly disagree
- disagree
- somewhat disagree
- somewhat agree
- agree
- strongly agree

JUDGE QUALITY MEASURE

We calculated the mean rating across all guidelines. The mean rating is intended to represent quality per treatment. A higher mean rating would map to more agreement with the guidelines, thereby indicating higher quality.

Usability Problem Instance Quality As Rated By Developers

A number of steps were involved in calculating measures of quality as rated by the developers. First, we created a questionnaire based on the set of Capra's guidelines. Next, three developers from the Scholar development team used the questionnaire to rate the usability evaluation reports produced by evaluators. Finally, the ratings were used as inputs to calculate a measure of quality. Figure 9 shows the objects that the developers used and produced.

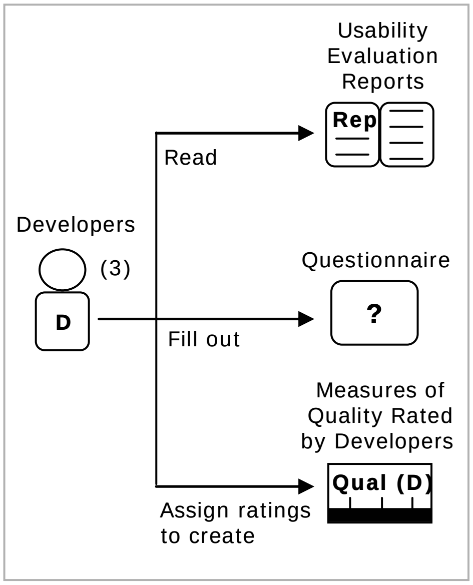


Figure 9: Developer's role; the number in parentheses indicates the number of developers

We included developer input via quality ratings to get the developers' feedback on the usability evaluation reports produced by the evaluators. Previous studies have included developer input. Hoegh et al. (2006) (which also includes previous work by Nielsen et al. (2005)) interviewed developers to obtain feedback on observation of user tests and usability evaluation reports. Hornbæk and Frøkjær (2005) interviewed developers regarding the utility of redesign proposals. Additionally, Law (2006) worked with developers to gather feedback on factors that influenced which usability problems the developers fixed. This study is similar to previous studies in that we are interested in the developers' feedback on the utility of the usability evaluation reports. This study differs from the ones performed by Hoegh et al. and Hornbæk and Frøkjær in that we are comparing different processes for producing usability evaluation reports instead of comparing usability evaluation reports to other forms of feedback. This study differs from the work by Law in that it focuses more on how the usability evaluation reports are produced as opposed to why developers interpret some usability evaluation reports to be better than others.

Developers

Three Scholar developers participated in this study. In exchange for their involvement, we performed a formative usability evaluation of Scholar, produced a usability evaluation report, and presented the results at a Sakai conference. The developers watched the same videos of Scholar as the evaluators watched during their study sessions. The developers worked independently and viewed the evaluators' usability evaluation reports in different orders to balance any potential familiarization or learning effects.

The questionnaire was designed to provide a view of the quality of the usability evaluation reports from the perspective of the developers. Questions 1 through 6 provided information on the quality and mapped to Capra's guidelines; we refer to them as the guideline questions. The guideline questions had a six-point Likert-type scale identical to the one used by the judges. Question 7 was a summary question that was intended to get a measure of a developer's overall opinion of the usefulness of a usability evaluation report. Developers assigned a value from 1 to 10, with 1 indicating that a usability evaluation report was not useful and 10 indicating that it was very useful.

Developer Quality Measures

We calculated the mean rating across the guideline questions, as well as for the summary question. The mean ratings are intended to represent quality per treatment. A higher mean rating for the guideline questions would map to more agreement with the guidelines, thereby indicating higher quality. A higher mean rating on the summary question would indicate a higher level of usefulness as perceived by the developers.

Hypotheses

Time

We hypothesized that support for the structured process would not affect the time that it took novice evaluators to perform evaluations.

Quality as Rated by Judges

We hypothesized that support for the structured process would increase the quality of the usability evaluation reports as rated by judges.

Quality as Rated by Developers

We hypothesized that support for the structured process would increase the quality of the usability evaluation reports as rated by developers.

Results

Time

The mean time for the freeform treatment was 4051 seconds (SD=902), and the mean for structured treatment was 4261 seconds (SD=865). A t-test indicated that there was not a significant difference between the treatment means, t(14)=0.48, p=0.64. There was also no significant difference between treatments in the number of evaluators who finished. In the freeform treatment, all eight evaluators finished; in the structured treatment, seven evaluators finished. The data supported our hypothesis that the structured process would not affect the evaluation in terms of time.

Quality as Rated by Judges

Means are based on individual ratings given by each judge, rather than the sum of the two ratings. Judges rated on a 6-point scale, which has been adjusted to a rating from –2.5 to 2.5. The adjustment was made, so that the value of 0 corresponds to the middle point between the somewhat disagree and somewhat agree points on the scale. The difference in mean rating across all guidelines by treatment was tested as part of a 2x6x2 mixed-factor ANOVA, with treatment as a between-subject factor, guideline and judge as within-subject factors, and evaluator as a repeated measure. There was a guideline main effect, F(5,168)=7.36, p<0.01; a judge main effect, F(1,168)=24.97, p<0.01; and a treatment main effect, F(1,168)=3.95, p<0.05. There were no interaction effects.

The guideline main effect indicated that some guidelines had mean ratings that were significantly different from other guidelines. This result was expected and was not of particular interest for this study.

The judge main effect was explored using a t-test of least square means; the mean rating for judge j1 (M=0.81, SD=1.34) was significantly greater than the mean rating for judge j2 (M=0.08, SD=0.85), t(168)=-5.00, p<0.01. Although judge j1 gave higher ratings than judge j2, the judges' ratings were associated, meaning that they gave higher/lower ratings to the same evaluator. Association was tested using Pearson's product-moment correlation by treatment. Using an alpha level of 0.05, there was a significant correlation between the judges for the freeform treatment, r=0.71, p<0.01, and for the structured treatment, r=0.55, p<0.01.

The treatment main effect indicated that mean rating for the structured treatment, M=0.45, SD=1.17, was significantly greater than for the freeform treatment, M=0.10, SD=1.54 (Figure 10), which supported our hypothesis.

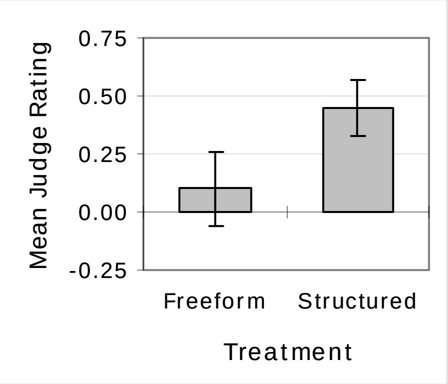


Figure 10: Quality as rated by judges; bars represent mean error

Quality as Rated by Developers

Means for the guideline questions are based on individual ratings given by each developer, rather than the sum of the three ratings. Developers rated on a 6-point scale, which has been adjusted to a rating from –2.5 to 2.5. The difference in mean rating by treatment was tested as part of a 2x6x3 mixed-factor ANOVA, with treatment as a between-subject factor, question and developer as within-subject factors, and evaluator as a repeated measure. There was a treatment main effect, F(1,252)=4.49, p=0.03. There were no other main effects or interaction effects.

The treatment main effect indicated that mean rating for the guideline questions for the structured treatment, M=1.21, SD=0.97, was significantly greater than for the freeform treatment, M=0.39, SD=1.43 (Figure 11), which supported our hypothesis.

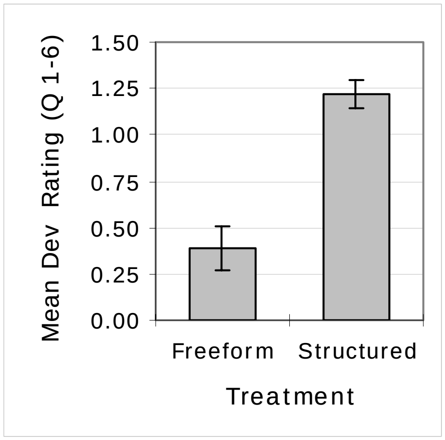


Figure 11: Quality as rated by developers (guideline questions, numbers 1 to 6); bars represent mean error

Both a histogram and a normal quantile plot suggested that the rating data for the summary question was not normally distributed and had a severe negative skew. As a result, a Wilcoxon rank-sum test, a non-parametric test, was performed. The test indicated that there was a significant difference in the medians between treatments, p<0.01; the median of the structured treatment was greater than the median of the freeform treatment (Figure 12), which supported our hypothesis.

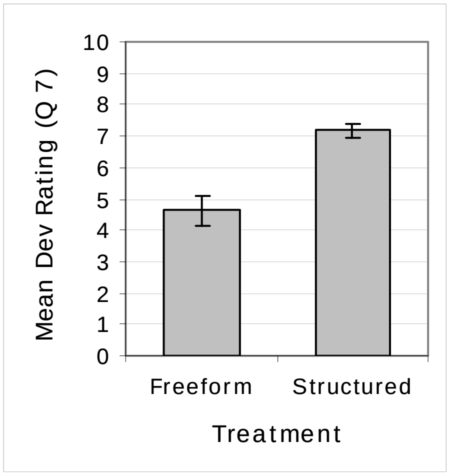


Figure 12: Usefulness as rated by developers (summary question, number 7); bars represent mean error

Discussion

The results supported our hypotheses. The evaluators in this study who used a structured approach to transform usability data produced usability evaluation reports that were rated to be of higher quality by both judges and developers. Additionally, these evaluators did not require any more time than the evaluators who used a freeform approach.

One interpretation of these results is that a structured process helps novice usability practitioners understand and relate usability data. It is necessary to understand the usability data generated during the usability data collection stage to accurately and completely describe UPs. The higher ratings assigned by judges and developers support this interpretation.

The results of this study, however, do not provide any data concerning the use of a structured process to help novice usability practitioners identify important usability data. In fact, we would argue that a structured process helps novice usability practitioners work with usability data, but it does not help them identify important usability data. Identification is related to a usability practitioner's ability to notice critical incidents, which can only be improved through experience or education.

Practical Implication

We embedded a structured process for working with usability data into DCART. The evaluators who used DCART produced usability evaluation reports of higher quality. This result suggests that the effectiveness of novice usability evaluators can be improved through better usability engineering tool support.

Existing usability engineering tool support helps to improve the efficiency of experts, but does little to improve the effectiveness of novices. Experienced usability practitioners typically have developed methods and strategies that work for them; they benefit from usability engineering tool support mostly in terms of efficiency because the tools automate tasks for them and allow them to produce quality usability evaluation reports in less time. Novice practitioners, on the other hand, may fail to understand and relate usability data appropriately without the guidance and support that can be provided by a usability engineering tool. As a result, novices stand to gain more in terms of effectiveness. Accordingly, it is important to include appropriate support in usability engineering tools.

Limitations of this Study

One limitation was the use of only three relatively short video clips (three to six minutes each) of representative users performing tasks with Scholar. In a real lab-based usability evaluation, an evaluator would watch a user perform a number of tasks over a longer period of time (typically one to two hours) and would have more of an opportunity to observe and understand the difficulties experienced by the user. We limited the number and length of video clips because we wanted to simulate a fixed-resources environment, which is novel for this area of research, but which might reflect real-world development constraints. We did, however, provide the evaluators in the study with background information on the context for the tasks and explain that the tasks represented a subset of tasks from an evaluation with five representative users. We also provided the evaluators with videos on Scholar and the correct way to perform the tasks attempted by the representative users.

A second limitation, in terms of generalizing the results, was the focus on novice usability practitioners. The structured process includes a model of the transformation of usability data that can be referenced by usability practitioners of all skill levels. Novice usability practitioners, however, will benefit most in terms of effectiveness from the application of the structured process because they have not developed a framework to help them understand or describe usability problems.

Practitioner’s Take Away

- There is a need for a more structured approach to transforming raw usability data generated by usability evaluation methods into usability information. Currently, this transformation is more of an art than a process and is highly dependent on the skill and experience of the usability practitioner.
- Usability problem instances serve as a bridge between raw usability data and usability problems. Each occurrence of a usability problem as encountered by a participant and observed by the evaluator is a usability problem instance. The same usability problem may be experienced by multiple participants or multiple times by one participant.
- For transforming usability data into usability information, a structured process based on usability problem instances can improve the effectiveness of novice usability practitioners.
- Existing usability engineering tool support helps to improve the efficiency of experts, but does little to improve the effectiveness of novices. To continue to grow the usability

engineering profession, it is important to improve usability engineering tools, so that they better support novices.

Acknowledgements

We would like to thank Dr. Tonya Smith-Jackson for her help with the study design and data analysis and Dr. Miranda Capra for the many conversations that helped us to develop the idea for this study.

Scholar is a course management tool by Blackboard Beyond Initiative. Microsoft Word is a trademark of Microsoft Corp. Morae is a trademark of Techsmith Corporation.

References

Bailey, G. (1993). *Iterative methodology and designer training in human-computer interface design*. Paper presented at CHI.

Capra, M. (Submitted for review, 2007a). Ten guidelines for describing usability problems, *Journal of Usability Studies*.

Capra, M. (2007b). *Comparing Usability Problem Identification and Description by Practitioners and Students*. Paper presented at HFES.

Castillo, J. C., Hartson, H. R., & Hix, D. (1998). *Remote usability evaluation: Can users report their own critical incidents?* Paper presented at CHI.

Cockton, G., & Lavery, D. (1999). *A framework for usability problem extraction*. Paper presented at Interact.

Cockton, G., Woolrych, A., & Hindmarch, M. (2004). *Reconditioned merchandise: extended structured report formats in usability inspection.* Paper presented at the CHI.

Dumas, J. S., Molich, R., & Jeffries, R. (2004). Describing usability problems: Are we sending the right message? *interactions, 11(4),* 24-29.

Hix, D., & Hartson, H. R. (1993). *Developing User Interfaces: Ensuring Usability through Product and Process.* New York, USA: John Wiley & Sons, Inc.

Hoegh, R. T., Nielsen, C., Overgaard, M., Pedersen, M., & Stage, J. (2006). The impact of usability reports and user test observations on developers' understanding of usability data: An exploratory study. *International Journal of Human-Computer Interaction, 21(2),* 173-196.

Hornbæk, K., & Frøkjær, E. (2005). *Comparing usability problems and redesign proposals as input to practical systems development.* Paper presented at CHI.

Howarth, J. (2007). Supporting Novice Usability Practitioners with Usability Engineering Tools. Unpublished dissertation. Blacksburg, VA: Virginia Tech. (http://scholar.lib.vt.edu/theses/available/ etd-04202007-141645/)

ISO. (1998). 9241 - *Ergonomic requirements for office work with visual display terminals (VDTs) Part 11: Guidance on usability*. Geneva, Switzerland: International Standards Organization.

Jacobsen, N. E., Hertzum, M., & John, B. E. (1998). *The evaluator effect in usability tests.* Paper presented at CHI.

Lavery, D., Cockton, G., & Atkinson, M. P. (1997). Comparison of evaluation methods using structured usability problem reports. *Behaviour and Information Technology, 16(4-5),* 246-266.

Law, E. (2006). Evaluating the downstream utility of user tests and examining the developer effect: A case study. *International Journal of Human-Computer Interaction, 21(2),* 147-172.

Lund, A. M. (2006). Post-Modern usability. *Journal of Usability Studies, 2(1),* 1-6.

Nielsen, C., Overgaard, M., Pedersen, M., & Stage, J. (2005). *Feedback from usability evaluation to user interface design: Are usability reports any good?* Paper presented at INTERACT.

Nielsen, J. (1992). *Finding usability problems through heuristic evaluation.* Paper presented at CHI.

Nielsen, J. (1994). Heuristic evaluation. In J. M. Nielsen, R. L. (Ed.), *Usability Inspection Methods*, pp. 25-62. New York: John Wiley and Sons.

Polson, P. G., Lewis, C., Rieman, J., & Wharton, C. (1992). Cognitive walkthroughs: A method for theory-based evaluation of user interfaces. *International Journal of Man-Machine Studies, 36(5),* 741-773.

TechSmith. (2007). Morae: Usability testing solution for web sites and software. Retrieved October 18, 2007, from http://www.techsmith.com/Morae.asp.

Theofanos, M. (2005). Towards the design of effective formative test reports. *Journal of Usability Studies, 1(1),* 27-45.

Wixon, D. (2003). Evaluating usability methods: Why the current literature fails the practitioner. *interactions, 10(4),* 28-34.

About the Authors

Jonathan Howarth
is a Human Factors Specialist at HumanCentric Technologies in Cary, NC where he designs and evaluates a variety of commercial products. Jonathan graduated from Virginia Tech in 2007 with a Ph.D. in computer science

Rex Hartson
is a Professor Emeritus with the Virginia Tech Department of Computer Science. He is a consultant, researcher, educator, and practitioner in human-computer interaction and usability engineering.

Terence S. Andre
is the Dayton Office Director for Aptima, Inc. He also leads the human-computer interaction team and is responsible for interaction design, usability analysis, and new business development. Dr. Andre was recently an associate professor at the Air Force Academy, teaching courses in human factors, system design, and human-computer interaction. Dr. Andre graduated from the Air Force Academy in 1987 and received his Masters in Industrial Engineering from California Polytechnic State University (Cal Poly) and his Ph.D. in Industrial and Systems Engineering from Virginia Tech.