



Group Usability Testing: Evolution in Usability Techniques

Laura L. Downey

University of New Mexico

Department of Biology, MSC03 2020

Albuquerque, NM 87131-0001

ldowney@lternet.edu

Abstract

Usability testing has a long history. In its early form, it was conducted with many individual participants much like traditional research experiments. With the advent of discount usability engineering techniques, fewer participants were required (5-7 versus 30-50) and protocols were simplified. The evolution from "many to few" in usability testing has become the standard in formative testing. What is the next tool in our toolbox?

This paper uses a case study to introduce a formative method called "group usability testing." It involves several to many participants individually, but simultaneously, performing tasks, with one to several testers observing and interacting with participants. The idea for group usability testing arose as an answer to limited time resources and the availability of many users gathered together in one place. Data characteristics, benefits, and drawbacks of group usability testing are discussed. Additionally, this method is compared/contrasted with individual usability testing, co-discovery, task-based focus groups, and cooperative usability testing.

Keywords

Group usability testing, collaborative design and evaluation technique, usability testing, formative usability method

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Copyright 2006, UPA.

Introduction

There has been an evolution in the way people use technology. Initially, most used individual applications for specific tasks. Later, individuals used a variety of applications to both perform tasks and gain knowledge. Most recently people have been collectively using a variety of applications and technology to collaboratively solve problems, find new information, or enable new ways of working together.

Usability engineering techniques have developed within this evolution of technology usage. As people began collaboratively using technology, collaborative design techniques, such as participatory design also emerged. Cooperative usability testing [Frøkjær and Hornbæk, 2005] is a new technique that emphasizes collaborative problem identification between the tester and the user.

Usability engineering techniques have also evolved in response to practical resource constraints of time, money, and availability of user participation. Testing with a small number of users and remote usability testing are examples of techniques that evolved in response to practical constraints.

This paper describes a *group usability testing* approach. The testing variation was developed to take advantage of the presence of a large group of users at a training workshop. The group usability testing method is described as applied to a scientific modeling and analysis application called Kepler.

Definition of Group Usability Testing

Group usability testing is defined as a group of users individually, but simultaneously, performing a set of tasks with one or more testers observing and interacting with participants. The interaction style is a

minimal version of the active intervention method [Dumas and Redish, 1994], rather than the standard think-aloud method often employed in individual usability testing. Results of two rounds of group usability testing are included. The approach is compared/contrasted with other formative methods and proposed as an additional usability engineering technique. Implications including data characteristics, benefits and drawbacks of group usability testing, and future research are also presented.

The Evolution of Usability Testing

Usability testing can be simply defined as the “process of learning from users about a product’s usability by observing them using the product...” [Barnum, 2002]. It has a long history and originally was conducted with a large number of users (30-50) [Barnum, 2002] and as a research experiment with accompanying statistical analysis. Traditional think-aloud testing was often videotaped and detailed analysis performed [Nielsen, 1993].

With the advent of discount usability engineering, a more simplified version of think-aloud testing came into existence. Modern formative usability testing methods moved toward the model of 5-7 representative users generally finding about 80% of the problems. Research by Nielsen, Virzi and Lewis [Nielsen, 1994, Virzi, 1994 and Lewis, 1990], supported the notion of using a small number of users for testing to find the majority of major usability issues in the design phase. [Barnum, 2003] nicely summarizes the results of this applied research.

Formative usability testing is usually done individually, or in some cases in pairs (co-discovery). Evaluation in both testing methods is done by the usability

professional. In more recent times, one trend in design and testing is to more fully involve participants in problem determination.

Cooperative usability testing (CUT) [Frøkjær and Hornbæk, 2005], is described as a technique that involves a videotaped individual usability testing session, followed closely by a joint retrospective analysis of the video where the participant identification and agreement of usability problems is stressed as a key element in the evaluation phase. Most recently (August, 2006), a message titled “group usability testing” [Hal Shubin, Interaction Design, Inc., 2006, communication on private usability mailing list] was posted to a popular usability mailing list and several responses ensued.

The original message contained a description of an informal study where four users performed tasks and four testers logged issues simultaneously while all were seated at the same dining room table. A focus group-type discussion followed in the living room. Shubin's group variation on usability testing took advantage of a group of users being gathered in one place. Clearly, there is a practical trend at work in the evolution of usability testing, but there is also a broad trend toward a shared experience and more collaborative forms of design and evaluation within usability [Bødker and Buur, 2000, 2002].

The Application

The software application used in the group usability tests is a scientific workflow application called Kepler [Altintas, et al, 2004 and Ludäscher, 2006]. Kepler is an open source modeling and analysis tool for creating, visualizing, executing, and documenting scientific

workflows. A scientific workflow is a collection of data flow and analytical steps that formalizes the research process.

Usability evaluation of Kepler was done as part of the Scientific Environment for Ecological Knowledge (SEEK) project [Michener, et al, in press]. SEEK, a National Science Foundation initiative, is focused on 1) creating cyber-infrastructure and applications for ecological, environmental, and biodiversity research and, 2) educating the ecological community about ecoinformatics.

A SEEK training workshop presented an opportunity to interact with several scientists learning to use the Kepler application. However, there would be limited time and access for usability activities in the workshop. The idea to conduct group usability testing was formulated.

Procedure

Two identical rounds of group usability testing were conducted on the Kepler software (Figure 1). The testing was done as part of a set of usability activities piggy-backed onto two training workshops. During the workshops, attendees received training on several ecoinformatics tools. Thirteen attendees participated in December 2004. Twenty workshop attendees participated in January 2005.

The purpose of conducting the usability activities included the following:

- begin validating some of our assumptions about the target user audience
- identify usability issues that exist in Kepler
- gather information on future needs

The test included the following usability activities (in temporal order):

1. User Profile Survey – 20-30 minutes
 - Technology Expertise
 - Software Inventory
 - Job, Education, and General Demographics
2. Basic Tasks Exercise – 1 hour
 - Run an existing workflow, then add one output/display component.
 - Create a new workflow (simple graph plot of data).
3. Usability Issues Discussion – 1 hour

These activities were embedded the training program. The user profiling activity was done at the beginning of the training workshop. As part of the training, participants were given an introduction to key terms and concepts, and also performed several familiarization exercises with the Kepler software. After this introduction to Kepler, the group usability testing was conducted.

Participants were given two basic task exercises to perform on their own. These tasks were given as a set of written instructions. Participants were told that observers would be watching and recording issues, and that they might ask quick questions. Users performed

the given tasks individually, but simultaneously. The participants performed the following tasks: 1) modify an existing workflow, and 2) create a simple workflow.

Users were seated in one large training room in groups of five, with each group in a somewhat circular pod configuration. They couldn't easily see their neighbor's screen without purposefully leaning over to look.

During testing, multiple observers walked around and interacted with participants, answering questions, and minimally probing users. The tester/observers included a usability professional, a trainer, and a software developer.

The usability engineer recorded issues in both rounds of testing, sometimes conferring real time with another observer, but with the intent not to disturb participants. In the second round of testing, the usability engineer focused mainly on recording new issues, but did note several of the same issues that had emerged in the first round of testing.

No significant changes were made to the software between rounds of testing. When the group usability testing was complete, the usability engineer facilitated a discussion on observed and user-stated usability issues.

The facilitator asked the group to discuss the issues and problems they encountered during task performance, probing for clarification and perceived difficulty. Additionally, observations recorded during testing were introduced as a way to relate observations and perceptions and focus the discussion. Issues were recorded and prioritized from a group perspective. Participants also suggested and prioritized features to consider for future inclusion in the Kepler software.

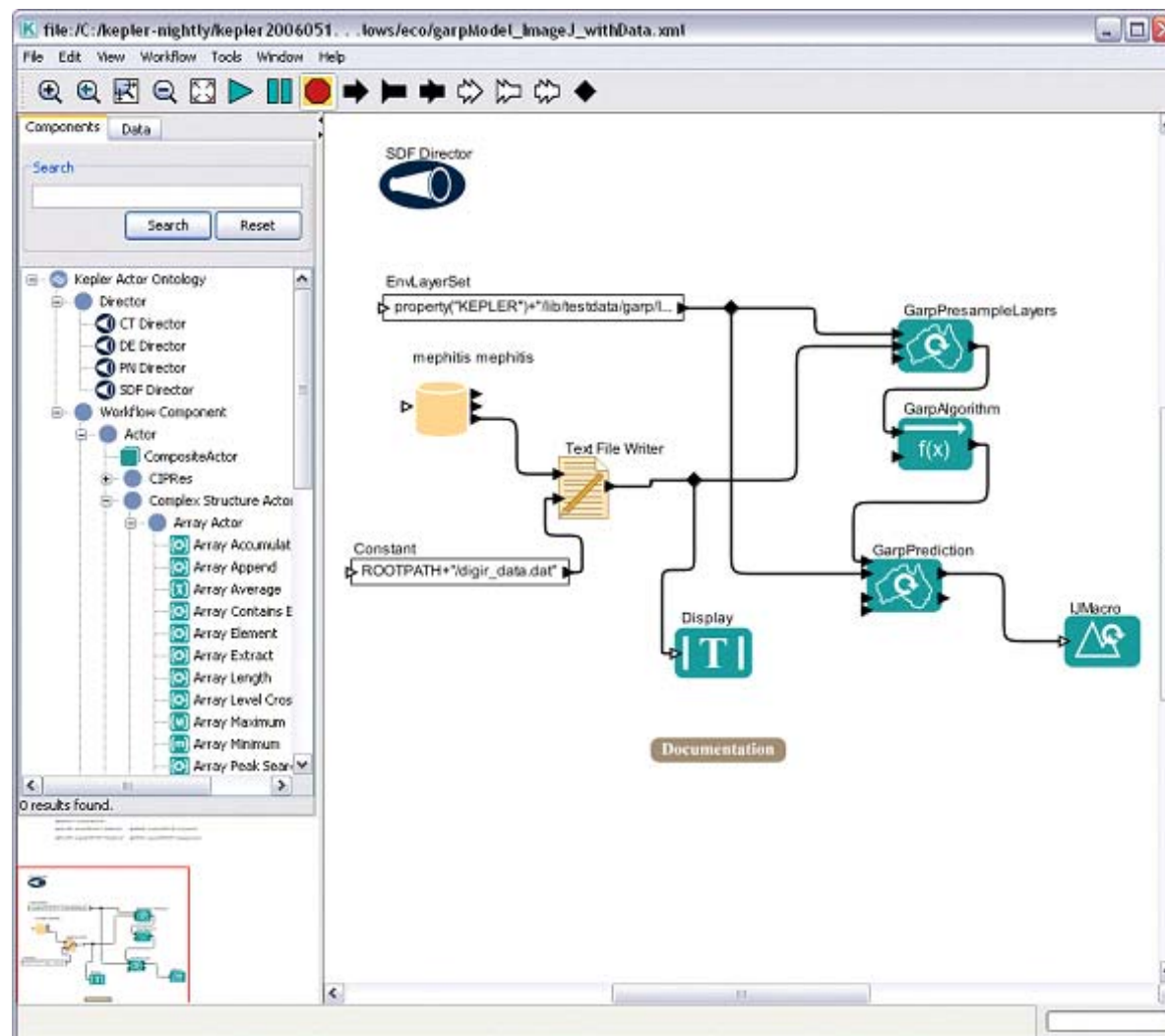


Figure 1 – Kepler application with sample scientific workflow displayed

Results

In both rounds of testing, participants were able to perform both tasks but usability issues were identified. There was an overlap in major usability issues found in both rounds of testing. Two examples were: 1) no feedback of workflow execution, and 2) lack of online documentation. Some of the different issues recorded from round one and round two could be viewed in the same general usability issue category, e.g., confusing terminology. Known bugs were also experienced in both rounds of testing, such as faulty/slow search engine performance.

Participants in round one testing were more technically experienced and more familiar with scientific modeling and analysis overall. As a result, participants in the two rounds of testing could be considered different user groups and some variation in issues was expected. Round one testing produced fifteen usability issues across both tasks. Round two testing included some variation upon previously noted categories of usability issues from round one (e.g., terminology issues), but produced only one new usability issue. That issue was difficulty changing plot display parameters. In round one testing, fourteen design recommendations (eight were classified as high priority) were made by the usability engineer after data analysis. In round two testing, two additional recommendations were made (one high priority). **Table 1** lists the high priority design recommendations resulting from both rounds of group usability testing.

The facilitated discussion of round one resulted in five recommendations of future features to strongly consider. The round two discussion resulted in two different recommendations from the usability engineer.

Table 2 lists the future features to consider resulting from the facilitated discussions. Round one participants, being more technically experienced produced a more detailed and technical list of discussion items. Round two participants experienced significant search problems (software bug) on the second task and not surprisingly were very focused on this issue during the discussion. There may have been more overlap between round one and round two suggestions if the search problems had not occurred.

In summary, two rounds of group usability testing were conducted where 33 users were observed using an application and they participated in a collaborative discussion on usability issues and future features to consider. The group usability tests uncovered usability issues that were translated to 16 design recommendations (with nine having a high priority). The follow-on facilitated discussions produced a list of seven new features for consideration.

Table 1 – High priority design recommendations resulting from data analysis of group usability testing

<i>High Priority Design Recommendations – Round One</i>
<ol style="list-style-type: none"> 1. Incorporate the ability to include functionality from external programs (e.g., R and/or S+) as actors in Kepler. 2. Provide better user documentation, such as a cheat sheet listing all the actors and their functions, along with a cookbook on how to quickly start building scientific workflows and the basics on wrapping actors. 3. Fix known software bugs: <ol style="list-style-type: none"> a. Ports display problem b. Multiple simultaneous user search c. Caching problem reading workflow save tool tips issues d. Left pane re-sizing problem e. Copy/paste overlay problems 4. Provide real-time feedback for workflow progress as a default. 5. Provide a cancel for the search function. 6. Provide a clear/reset for the search function. 7. Provide standard cut, copy, paste, delete on right click menus. 8. Incorporate combo boxes with pick lists in as many places as possible.

High Priority Design Recommendations – Round Two

1. In addition to recommendations from round one, address the terminology issues in terms of computer science-centric labels and term consistency (e.g., consider renaming new graph editor to new workflow and simplify director names so users understand the kind of processing that is occurring. Also consistently differentiate between run, run window, go, and search.)

Table 2 – Future features to strongly consider resulting from data analysis after facilitated group discussion

<i>Future Features to Strongly Consider</i>	
Round One	Round Two
<ol style="list-style-type: none"> 1. Natural language summary of a workflow 2. Summarization of workflow (in a publishable format) 3. Ability to assign checkpoints at various points in the workflow so that the user can check progress and make decisions on whether to modify or continue, etc. 4. Ability to easily visualize data at various places in the workflow 5. Guided analysis (wizard-like functionality for constructing a workflow) 	<ol style="list-style-type: none"> 1. Provide browsing and filtering mechanisms especially for data and data nodes on the ecogrid. 2. Implement the “most recently used” concept for workflows and actors.

Analysis

Data Characteristics of Group Usability Testing

As with other testing methods and design activities, group usability testing produces lots of useful data. Usability issues were observed and recorded as in traditional 1:1 testing. However, because of the group setting, it wasn't feasible to have in-depth discussions, so it is probable that problem identification may not be as detailed. (Although some of this may be addressed retrospectively if a discussion session follows the testing.)

The tasks given were basic tasks, not complex tasks. It is suspected that complex tasks will not lend themselves well to group usability testing because a more in-depth interaction between tester and participant may be required to adequately evaluate issues encountered during performance of complex tasks. It is also suggested that group usability testing is a method better suited to uncovering major usability issues. Low to medium issues may not emerge during basic task performance or limited interaction between participant and tester. On the positive side, being able to see the same problem occur across multiple participants can validate the criticality of an issue in a very short time frame.

Benefits of Group Usability Testing

Group usability testing produces lots of useful data, and allows several to many users to be simultaneously tested in a relative short time (minimizes tester's time). It can quickly validate major usability issues. The group usability testing method also takes advantage of a meeting or gathering of user groups (minimizes user's time). Often, the reason usability professionals do not test lots of users (including different user groups) is

because of limited time and budget. As practitioners, we often wish we had time to test more users and gather more data to inform design decisions. Convincing stakeholders of usability issues may be easier with more users, as there is power in numbers. Group usability testing followed by a focused discussion also supports the trend toward richer user involvement in problem identification and a collaborative design and evaluation experience.

Drawbacks of Group Usability Testing

The most obvious drawback in group usability testing is participants affecting each other, as can happen in co-discovery [Barnum, 2002]. Missing issues/observations in a group setting can also occur because so much is happening simultaneously. This can be partially mediated with multiple observers. However, the need for multiple observers could be a drawback if usability professionals are limited. Again, this can be mediated by involving and asking other members of the design and development team to participate as observers (with some minimal training). Because of the group setting, the think-aloud technique is not practical, so that kind of data would not be gathered. Individual interaction is limited, but some active intervention probing is possible with multiple observers. Reduced interaction may also result in less detailed problem descriptions.

Compared/Contrasted to Other Testing Methods

Group usability testing matches the traditional definition of usability testing. As with other testing methods, empirical data are collected. The major difference with other testing methods is that group usability testing allows for a much larger number of users to be tested and in a much shorter time.

Compared/Contrasted to Task-Based Focus Group and Cooperative Usability Testing

Group usability testing followed by a facilitated discussion is similar to a task-based focus group [Hackos & Redish, 1998]. A task-based focus group is one in which participants are given a set of tasks to perform before the focus group, or at the beginning of the focus group, with a discussion afterward.

The major difference is that in group usability testing empirical data are gathered, whereas in a task-based focus group, participants are not observed doing the tasks, but often perform them independently then self report the results.

The other difference being there is usually only one facilitator in a task-based focus group, not a set of testers/observers as in group usability testing. The major advantage common to both methods is that performing tasks then discussing them provides a richer, more focused discussion than an independent discussion [Hackos & Redish, 1998]. This richer discussion element can also be seen in cooperative usability testing (individual testing followed by joint retrospective analysis of video by user and tester). The commonality between group usability testing followed by a facilitated discussion and cooperative usability testing is the notion of more actively involving the user in problem identification.

Summary

Clearly there are several commonalities and differences in the methods discussed here, as well as advantages and disadvantages. **Table 3** provides a comparison/contrast of five formative usability techniques: Individual usability testing (I), co-discovery (CD), task-based focus groups (TBF), cooperative

usability testing (C), and group usability testing (G) followed by a focused discussion.

Table 3 – Comparison/contrast of five formative usability engineering techniques

<i>Comparison of Formative Usability Techniques</i>					
Characteristic	I	CD	TBF	C	G
Time/cost efficient			X		X
Shared task experience between 2 or more users		X			X
Supports large # of users			X		X
Multiple testers/observers not generally required	X	X	X	X	
Video not required	X	X	X		X
Involves task observation	X	X		X	X
Involves tester interaction with user during tasks	X	X		X	(X)
Takes advantage of user gatherings			X		X
Involves user actively participating in problem identification and analysis				X	X
Provides data on large number of users			X		X
Provides detailed data on individual users	X	X		X	
Think aloud method used	X			X	
Active intervention method used	X	X		X	(X)

<i>Comparison of Formative Usability Techniques</i>					
Characteristic	I	CD	TBFG	C	G
Involves other members of team as active observers/testers					X
Collaborative design and evaluation method			X	X	X

Conclusion

As practitioners, we know deciding which usability activities to use in a project depends on a variety of resource constraints, such as time, cost, staff availability, user access, management support, testing facilities, and prototype/software maturity [Stone et al, 2005]. Group usability testing provides another choice to consider, depending on resource constraints and testing objectives, and is one more method to add to our practitioner's toolbox. It is time efficient for both testers and users and provides a mechanism for testing more users than traditional formative testing methods. Group usability testing also supports the evolution toward more collaborative forms of evaluation.

Practitioner's Take away

- Group usability testing involves several to many participants individually, but simultaneously, performing tasks, with one to several testers observing and interacting with participants. The interaction style between tester/participant is a minimal version of the active intervention method.

- Group usability testing supports testing with several to many users, but in much less time than individual testing.
- If you can take advantage of a gathering of a large number of users, consider group usability testing.
- If you need usability data from more than a small number of users, consider group usability testing.
- If you have limited time and/or access to users, consider group usability testing.
- Use group usability testing if tasks are relatively simple and you are looking for major usability issues, not when users must perform complex tasks or when you are looking for low to medium usability issues.
- Use observations and results from group usability testing to drive facilitated follow-on discussions with users.
- Group usability testing followed by a focused discussion supports the evolution toward cooperative and more collaborative design and evaluation techniques.

Future Research

Future research plans include conducting and comparing/contrasting the same usability test 1) individually with five users, and 2) group usability testing with five users. Another research direction use a groupware application and apply the group usability testing method in a collaborative fashion, followed by a focused discussion group, to examine whether it should be the preferred method for testing groupware applications.

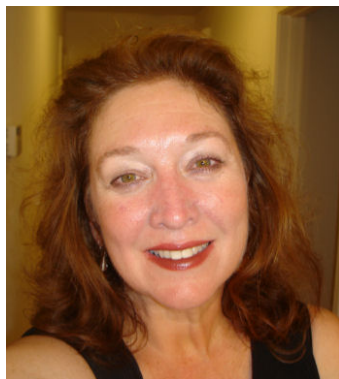
Acknowledgements

The author appreciates the time and feedback of the scientists who participated in the usability activities reported here. Thanks also go to the members of the SEEK project, especially Dr. Deana Pennington for her encouragement and support in this work. SEEK is a National Science Foundation (NSF) initiative that is supported by NSF grant ITR 0225674.

References

- Altintas, I., Berkley, C., Jaeger, E., Jones, M., B. Ludäscher, B., Mock, S. (2004) *Kepler: An Extensible System for Design and Execution of Scientific Workflows*. 16th International Conference on Scientific and Statistical Database Management. IEEE publication number P2146.
- Barnum, C. M. (2002) *Usability Testing and Research*. New York: Longman Publishers, pp 9, 10, 147.
- Barnum, C. M. (2003) Applied Research: What's in a Number?, *Usability Interface*, STC Usability SIG Newsletter, January 2003, Vol 9, No.3.
- Bødker, S. and Buur, J. (2002) The Design Collaboratorium: A Place for Usability Design, *ACM Transactions on Computer Human Interaction (TOCHI)*, Volume 9, Issue 2, pp 152-169, ACM Press.
- Bødker, S. and Buur, J. (2000) From Usability Lab to "Design Collaboratorium": Reframing Usability Practice. *Designing Interactive Systems: Processes, Practices, Methods, and Techniques*. New York: ACM Press. pp 297-307.
- Dumas, J. S. and Redish, J. C. (1994) (second printing), *A Practical Guide to Usability Testing*. Intellect, Ltd. pp 31.
- Frøkjær, E. and Hornbæk, K. (2005), *Cooperative Usability Testing: Complementing Usability Tests with User-Supported Interpretation Sessions*. Conference on Human Factors in Computer Systems, pp 1383-1386. ACM Press.
- Hackos, J. T. and Redish, J. C., (1998), *User and Task Analysis for Interface Design*. Hoboken, NJ: John Wiley & Sons, Inc., pp 147.
- Lewis, J. R. (1994) Sample Sizes for Usability Studies: Additional Considerations, *Human Factors*, 36, 368-78.
- Ludäscher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger-Frank, E., Jones, M., Lee, E., Tao, J., Zhao., Y. (2006) Scientific Workflow Management and the Kepler System, in *Concurrency and Computation: Practice & Experience, Special Issue on Scientific Workflows*.
- Michener, W.K., Beach, J.H., Jones, M., Ludaescher, B., B., Pennington, D.D., Pereira, R.S., Rajasekar, A., and Schildhauer, M., [in press], A knowledge environment for the biodiversity and ecological sciences, *Journal of Intelligent Information Systems*.
- Nielsen, J. (1993) *Usability Engineering*. Burlington, MA: Academic Press, Inc. pp 16-18.
- Nielsen, J., (1994) "Guerrilla HCI: Using Discount Usability Engineering to Penetrate the Intimidation Barrier in Bias", in Randolph, G. and Mayhew, Deborah J. (Eds.), *Cost-Justifying Usability*. Burlington, MA: Academic Press. pp 245-272.
- Stone, D., Jarrett, C., Woodroffe, M., Shailey, M. (2005) *User Interface Design and Evaluation*. San Jose CA: Morgan Kaufmann, pp 446.
- Virzi, Robert A. (1990) "Streamlining the Design Process: Running Fewer Subjects," *Proceedings of the Human Factors Society*, Orlando, FL, pp 291-94.

Author Biography



Laura Downey is a senior research engineer at the University of New Mexico. She is working on an ecoinformatics project where she helps make technology work for scientists. Her background includes experience in government, academia, and industry where she has always been an

evangelist of usability. Laura is formally trained as a computer scientist, but has 14 years of experience in usability and human factors engineering. Her current research interests are in testing and evaluation methods, and user interfaces for semantic annotation.