INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

UMI[®]

On the Reliability of Usability Testing

by

Martin Kessner

**A thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfillment of
the requirements for the degree of
Master of Arts**

Department of Psychology

Carleton University

Ottawa, Ontario

December 2000

© 2000, Martin Kessner

National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-57704-X

Canada

The undersigned recommend to the Faculty of Graduate Studies and Research
acceptance of the thesis

On the Reliability of Usability Testing

submitted by
Martin Kessner
in partial fulfillment of the requirements for
the degree of Master of Arts

Chair

Thesis Supervisor

Abstract

Six professional usability testing teams conducted a usability test on an early prototype of an open dialogue box for Windows applications. Altogether, they identified 36 unique usability problems. No problems were detected by all six teams, 2 were found by five teams, 4 by four teams, 7 by three teams, 7 by two teams, and 18 problems were identified by one team only. In terms of the mean number of teams detecting a problem, there was more agreement among teams in this study compared to an earlier study and there was more agreement among the teams on severe vs. minor problems. Implications for the cooperation between usability testers and their clients are discussed.

Acknowledgements

The research presented in this thesis has been a collaborative effort. I would like to thank everybody who contributed.

Special thanks go to the people who volunteered to do the usability tests thus providing the data to be analysed: Amy Dillon, Christopher Baum, Coralie Lalonde, Cort Fowler, Farouk Meghji, Kathy Momtahan, Kyla Stewart, Meghan R. Ede, Paul Boyd, Sara Dolansky, Scott Emery, Sheila Thornton, and Steve Macko. They represent companies including Cognos, Integra Solutions, Itemus Solutions, and Nortel Networks (Usability Design Group and Design Interpretive). Thanks to Ed Prinsen and Jennifer Fraser from Corel. Ed provided the prototype and was always available for questions. Jennifer coordinated everything related to the non-disclosure agreements.

Dr. Jo Wood sacrificed countless hours providing a second set of eyes for the content analysis. Thank you for sharing your enthusiasm for this project with me. Dr. Richard F. Dillon was always available for advice and went through numerous revisions of this document. Thank you for being a steward of clarity and for all your support. Thank you, Dr. Gitte Lindgaard and Dr. Robert L. West for your input.

Finally, I want to express my deep gratitude to my parents and my sister Bettina who were there for me selflessly when I really needed them.

Table of Contents

Abstract	iii
Acknowledgements	iv
Table of Contents	v
List of Tables	vii
Introduction	1
Shortcomings of Previous Studies	4
Purpose of the Present Study: Improvements on Previous Studies	7
Method	9
Participants	9
Prototype	9
Procedure	10
Results	12
Creating the Problem Master List	12
Problem Grouping	14
Overlap in the Teams' Findings	15
Is There More Overlap in the Data of This Study Compared to Molich et al. (1999)?	16
Severity Ratings of the Problems	18
Is There More Overlap Among More Severe Problems?	19
Agreement of the Teams' Answers to the Designers' Questions	20
What Kind of Data Did the Teams Base Their Answers on?	22
Resources the Teams Dedicated to the Usability Test	23
External Validity Analysis	23
Discussion	25
Low Overlap in This Study but Higher Than in Earlier Studies	25
Emerging Pattern	26
More Overlap in Severe Problems	26
Agreement Among the Severity Raters	27
Potential Reasons for the Low Agreement of the Usability Testing Teams	27
External Validity	29

Construct Validity	30
Implications for the Cooperation Between Usability Testers and Their Clients	31
Implications for the Value of Doing Usability Testing.....	31
References	33
Appendix A: Instructions.....	35
Appendix B: Client Concerns	36
Appendix C: Post Test Questionnaire.....	37
Appendix D: Excluded Problems	38
Appendix E: Included Problems	40
Appendix F: Severity Ratings	48
Appendix G: Post Test Questionnaire, Resources Dedicated	49
Appendix H: Teams' Email - Questions for the Designers.....	50
Appendix I: Post Test Questionnaire, Teams' Deviations From Their Usual Procedure	51
Appendix J: Post Test Questionnaire, How realistic was this test?.....	53

List of Tables

Table 1. Number of Usability Problems Broken Down by Team.....	13
Table 2. Number of Problems Detected by a Given Number of Teams in this Study Including only Usability Problems vs. all Problems, and in Molich (1999)...	16
Table 3. Number of Minor vs. Severe Problems Detected by a Given Number of Teams	20
Table 4. Answers of the Teams to the Designers' Questions.....	21
Table 5. Information Provided by Each Team	22

On the Reliability of Usability Testing

In the Human Computer Interaction community, usability testing is often viewed as the queen of usability evaluation methods. In many studies that compare usability evaluation methods, usability test results have served as benchmarks to compare the effectiveness and/or efficiency of alternative methods (e.g., John & Marks, 1997; for a critical review see Gray & Salzman, 1998). In the corporate world, many companies rely on usability testing as an integral part of the user-centred design process. While it has often been assumed that usability testing can be invaluable during the design process, there is little research about its reliability.

Nielsen (1993) viewed reliability in terms of how many test participants are needed in a usability test to achieve accurate and stable results. Naturally, the more users who take part in a usability test, the larger the number of problems identified. In formative usability testing, that is usability testing that plays a role in early stages of the product development, accuracy and stability of measures may not be the ultimate goal. Often, the main objective is to find problems users encounter while performing tasks with a proposed design. According to Nielsen and Landauer (1993), there is a diminishing return on investment of having more users participate in a usability test with regards to the proportion of detected problems. In the real world context, budget constraints and the requirement for short turn-around times limit the number of test participants. It makes more sense to invest in several smaller usability tests in different iterations than to invest in one usability test with many users (e.g., Nielsen, 1993).

While Nielsen and Landauer (1993) examined the outcome of usability testing as a function of the number of test participants, Jacobsen, Hertzum, and John (1998) analysed the effect of the number of evaluators analysing identical video tapes of a usability test and found a substantial evaluator effect to be a threat to the reliability of usability testing. Only 20% of the usability problems were found by all four evaluators, 13% were found by three evaluators, 20% by two evaluators, and 46% by a single evaluator. In a real world setting, the effect of the usability evaluation personnel could be even higher because they do not evaluate identical video tapes as in this highly controlled lab study. They also create the test plan, interact with the test participants during the testing sessions, interpret the results, and compile the report of the findings, thereby introducing further possible sources for outcome variability.

In two studies, Molich et al. (1998, 1999) chose a more “naturalistic” approach than that used by Jacobsen et al. (1998) to examine usability testing. Several existing usability teams performed a usability test on the same product. In both studies, the overlap of detected problems was very small. In the first study (Molich et al., 1998), four teams reported 141 usability problems altogether. Only 1 of the problems was found by all four teams, 1 problem was detected by three teams, and 11 problems were reported by two teams, while the other 128 problems were discovered by only one team. The second study (Molich, 1999; Molich & Kindlund, 2000; Molich et al., 1999) was no less devastating: Of the total of 300 problems the nine teams detected, none was detected by all nine teams or even eight teams. One problem was detected by seven teams, 1 by six, 4 by five, 4 by four, 15 by three, and 49 by two teams. The majority of the problems (228, i.e. 75%) was detected by a single team.

It does not come as a surprise that these results caused concern in the usability community. While the Molich results do not suggest that usability testing is not worth doing – many usability problems were identified in both studies – it would have been more comforting if the usability teams had identified the same problems. Reproducible results are a necessary property of a mature methodology.

Molich's studies were not designed to assess the reliability of usability testing. The main goals were to “provide a survey of the state-of-the art within professional usability testing”, to “set a benchmark against which other usability labs can measure their usability testing skills”, to “show participating usability labs their strengths and weaknesses” (Molich et al., 1999, p. 83), and “to show different approaches to the usability testing task” (Molich et al., 1999, p. 84). Molich et al. (1998, 1999) were more interested in comparing what the testing teams did and how they reported it than in testing results and their overlap. The design of the studies was not optimized for the assessment of the reliability of usability testing. Therefore, the shortcomings addressed below are shortcomings only with regards to the reliability issue.

For the present study, a reliability concept similar to the one in Molich et al. (1998, 1999) was adopted. Multiple usability testers performed a usability test on the same product. This concept differs from the reliability concepts used in psychometrics where the reliability of an instrument is assessed by testing the same set of participants with the same instrument multiple times (test-retest reliability) or the same set of participants is tested with multiple equivalent versions of the instrument (parallel test reliability). While it would be possible to have multiple teams do a usability test using the same test tasks, metrics, and usability goals for the sake of a study, the information

gathered would not represent real practices accurately. In practice, the test team tailors a set of test tasks, metrics, and usability goals individually to the product to be tested; and establishes the objectives of the test. The test team also decides on the background and number of participants, recruits them, carries out the testing, and analyses and reports the data. The testing itself is often more similar to a clinical interview, in the sense that there can be extensive interaction between the tester and the test participant, than to filling out a psychometric test questionnaire. All these decisions and processes potentially introduce additional variance, which must be taken into account if the reliability of real world usability testing is to be assessed.

Shortcomings of Previous Studies

Molich et al. (1999) identified three major shortcomings of the earlier study (Molich et al., 1998). Firstly, the product to be tested was a four-year-old Windows program that was well developed. Second, the testing teams couldn't communicate with the development team at all. This is unusual when a real usability test is performed. Third, "the teams ... were not given a high level task set to focus on. Consequently, they all tested different parts of the user interface." (Molich et al., 1999, p. 84).

Although efforts were made to address these shortcomings in the second study, there remain potential problems with the second study that might have caused the lack of overlap in the findings of the testing teams. The tested product in the second study, a web-based email service, was in the market for years before it was tested. Potentially, the more obvious, major usability flaws had been fixed, and only minor and less obvious flaws were left. With usability testing, major usability flaws are much more

likely to be found than less severe flaws (e.g., Virzi, 1992). Cosmetic problems are harder to find than glaring usability catastrophes and there will be less agreement on them. Following user centred design principles, usability testing is done in early stages of the product development cycle (e.g., Nielsen, 1993; Rubin, 1994). The Molich et al. (1999) data can't really be used to draw conclusions about formative usability testing but rather about summative usability testing where the purpose is providing data about the usability of a finished product. Summative usability testing, rather than formative usability testing, calls for a modified method (Bevan, 1998).

In Molich et al. (1999), the list of requests for information from the development team included specific issues, but there were 29 of them and the product had many features. Therefore, as in the first study, not every team tested the same features due to lack of time for testing everything requested. For example, only five of nine usability teams tested sending an email with attachment. If different parts of the product are tested, naturally the findings will overlap less. Moreover, some of the issues listed in the scenario were not usability issues but usage or marketing issues. These usually wouldn't be addressed by usability testing but with other methods, for example, a user survey. Thus, the list of problems was inflated by issues that were not usability related, for example, trust issues like "users are suspicious because the service is owned by Microsoft." Some usability teams might have felt it would be inappropriate to include this kind of non-usability information in their reports.

The user profile specified in the scenario ("anyone who would like the convenience of managing their email through a web browser") was very broad. While it is legitimate to have a broad target user group, the fact that some usability testing teams

tested novice users, some expert users, and some both novice and expert users might be another reason for the lack of overlap in the problems identified.

For establishing the list of all the different usability problems the teams found, Molich et al. (1999) used a debatable criterion to decide whether two reported problems were the same or not. If two problems required the same fix, then they were treated as the same problem. This is problematic for two reasons. Fixes don't necessarily and unambiguously come with the detection of a problem. In most cases, several solutions are imaginable. Second, several problems could be "fixed" by simply eliminating a whole feature, although these problems could be of very different nature and of different severity. Furthermore, the classification of problems was done by only one rater so there was no safeguard against bias and no indication of the reliability of the classification in terms of inter rater agreement. The lack of overlap could be due to conservative classification (biased towards uniqueness) of the detected usability problems. The list of the 300 "unique" problems was inflated by listing very similar problems as unique. For example, one problem was described as "Layout – Buttons: Clarify what should be a button, what a link and what something else, and change appearance accordingly" and another as "Layout – Buttons: Looks unprofessional that the application contains links in places where buttons should be used." This would inflate the number of problems and tend to decrease the overlap. On the other hand there might be a tendency to under-report problems. It seems to be common practice among usability testers not to list every minor problem that was encountered. Only one of the nine teams in Molich et al. (1999) reported every problem they found. This practice may have led to decreased commonality among the reports.

Furthermore, there were two student teams among the nine teams of Molich's second study. It is likely that usability testing benefits from experience, so it wouldn't be surprising if the student teams found fewer problems than the professionals. Finally, one of the professional teams didn't use usability testing methods to evaluate the product but a survey method, leading to decreased construct validity. Their data, therefore, cannot be used to assess the reliability of usability testing. Previous research indicates that different problems are found with different methods, suggesting a relative strength of usability testing when it comes to severe problems (e.g., Jeffries, Miller, Wharton, & Uyeda, 1991; Karat, Campbell, & Fiegel, 1992).

Purpose of the Present Study: Improvements on Previous Studies

The purpose of the present study was to address the issue of the reliability of professional formative usability testing by determining the agreement of the testing teams both on the list of usability problems found and on the questions raised by the design team. The agreement of the teams on the list of usability problems found was determined separately for severe and minor usability problems. The approach taken was the same as in the Molich et al. (1998, 1999) studies. Several teams tested the same product. To avoid factors that could bias the results towards non-overlap in the findings of the testing teams, thus accounting for the emphasis on reliability, the following modifications were made.

To provide data about formative as opposed to summative usability testing, the product to be tested was a prototype early in the development cycle. The design team had a number of concerns with their design that had not undergone any prior usability

evaluation and hadn't benefited from customer feedback because it had not been in the market.

To increase the probability that all teams test the same parts of the prototype, two steps were taken. First, the request of the design team was specific as to which features they needed data about and was not as extensive as the list in Molich et al. (1999). Second, the chosen prototype was a small part of an application as opposed to a whole application.

The participants were all professional teams with experience in usability testing. Although differences in experience cannot be ruled out, all teams did have substantial experience.

Because the results of the testing needed to be compared, the testing teams were explicitly asked to perform usability testing providing quantified, objective performance measures, subjective user ratings, user comments, and a severity rating from the testers for each problem detected.

In analysing the results, more than one individual was used at each step so that interrater reliability could be examined. This is described in the results section of this study.

Method

Participants

The design team designed and built the prototype and provided the request as to what should be tested. The testing teams designed the usability tests and did the usability testing, thus providing the data to be analyzed in this research. The author and a university professor teaching usability-related courses extracted the usability problems from the reports and grouped them. Finally, three independent usability experts, also university professors teaching usability courses, rated each problem for severity. To qualify for participation, a testing team had to have experience in usability testing in a professional setting. The testing teams were recruited from the local high tech industry and among alumni of the CURE Lab. Initially, seven testing teams agreed to participate. Six teams handed in their reports before the deadline. There was no compensation for the participating teams. They covered all expenses themselves and were aware of the Molich et al. (1999) study.

Prototype

The prototype tested by the testing teams was a novel open files dialogue box for applications running under Microsoft Windows, and was provided by a large software company. The dialogue box contained various new widgets and had the framework for a powerful search feature. The design was in a very early stage of development, existing only as a first iteration paper prototype at the time, and had not undergone any usability evaluation. The software prototype was built specifically for

this study on the basis of the previously existing paper prototype. It was functional in the relevant aspects. For example, because navigation was to be tested, the prototype displayed the actual files on the user's computer as opposed to a fixed, predetermined list. On the other hand, the advanced search feature was not to be tested, so it was not fully functional. The prototype contained known bugs that did not effect the testing. In iterative design processes, it is not unusual, and is even recommended, to test incomplete prototypes even if they contain known bugs so that the testing can be done early in the design process (e.g., Lindgaard, 1994).

Procedure

The testing teams downloaded the instructions (see Appendix A) and the prototype with written documentation from a restricted-access website. In the instructions, the design team explicitly asked for quantitative measures of performance, users' subjective ease of use ratings, user comments, and the testing teams' severity ratings on a four-point scale (1 = cosmetic problem, 2 = minor usability problem, 3 = major usability problem, 4 = usability catastrophe; suggested by, e.g., Nielsen, 1993, p. 103; Dumas & Redish, 1999, p. 324; Rubin, 1994, p. 278). Other than that, the testing teams were asked to do the usability testing as closely as possible to their usual procedure and to use their usual format for the written test report with the exception that the identity of the team shouldn't be apparent. The instructions also included a list of six research questions (shown in Appendix B) that the design team wanted to have addressed. Five of the questions asked whether users could find out how to use a certain feature. The remaining question asked about performance impediments due to a

non-conventional arrangement of widgets. Participating teams were explicitly told that the advanced search feature was not to be tested because it was considered a power user feature. The testing teams were asked to watch out for problems with the design that were not listed.

The design team was available for clarifying questions the testing teams had about the prototype and their task at any time between delivery of the prototype and handing in the report. However, the testing teams did not directly communicate with the design team. The testing teams were asked to send their questions by email to the author of the study, who passed them on to the design team, and forwarded answers to the testing team. This was done to log the questions and replies and to eliminate the need to contact the design team multiple times in case of duplicate questions. The questions and answers were not provided to the other teams. The teams were asked not to communicate with other usability professionals about the study to ensure that independent data were collected.

In accordance with previous studies, the testing teams were asked to keep track of the resources they dedicated to the test (number of people involved and their roles, person hours, equipment), and the deviations from their usual procedure to assess the external validity of the study (see post test questionnaire, Appendix C).

The people involved in the usability testing signed a non-disclosure agreement that inevitably revealed the name of the company that provided the prototype.

Results

Creating the Problem Master List

On the basis of the written reports of the usability testing teams, a master list of unique problems was created. In accordance with John and Mashyna (1997), the problem tokens were not reduced to problem types or categories. This seems reasonable because “developers need to know the specific problem (e.g., a problem with an item in a particular menu) and not the general one (e.g., “there are menu problems” ...)” (Gray & Salzman, 1998, p. 242).

All teams included summary tables of the problems they found. However, most teams mentioned additional usability problems in other places in their reports. Collectively, the teams listed a total of 102 problems in summary tables. In a first step of the data analysis, two evaluators independently decided for each of those problems whether it is a usability problem or not. Issues that were not usability problems included prototype limitations or bugs, user needs analysis problems, feature requests, and problems concerning the advanced search feature that was not to be tested. The evaluators both excluded the same 21 problems. There was an initial disagreement on 11 problems (10.8%). The evaluators discussed the 11 problems and came to an agreement on them. Altogether, 31 of the 102 problems were excluded, resulting in a total of 71 remaining problems. Of the excluded problems, 14 were bugs or prototype limitations, 11 were feature requests or user needs analysis problems, and 6 were other miscellaneous non-usability problems. For a list of excluded problems see Appendix D. Most of the excluded problems came from the reports of Team C and Team D. None of

the excluded problems was detected by more than one team. For a breakdown of the number of problems listed and the number of excluded problems by teams see the first two rows of Table 1 where columns are teams and the first two rows show the number of problems included in the summary table and the number of these problems excluded. Because this step was not taken by Molich et al. (1999), a comparison to their results was made with and without these non-usability problems.

Table 1

Number of Usability Problems Broken Down by Team

	Team					
	A	B	C	D	E	F
Problems in Summary Table	8	14	39	12	19	10
Problems Excluded	2	0	21	6	1	1
Problems Added	0	3	0	12	0	0
Final Number of Problems	6	17	18	18	18	9

In a second pass, both evaluators independently searched the reports for problems mentioned in places other than the summary list of problems. A breakdown of the number of added problems and total number of problems per team is presented in the last two rows of Table 1. Both evaluators independently suggested including the same 10 additional problems. After discussion, they agreed to include another 5 of 8 suggested problems. Altogether, 15 problems that were not mentioned in the team

report summary tables were added, most of them for Team D, resulting in a new total number of problems of 86. Most of the problems that were added for Team D were described in the executive summary instead of the summary table.

Problem Grouping

The original wordings of the problems in the list of the 86 problems were maintained. Therefore, this list included some problems that were essentially the same but had been put in different words by the individual teams. For example, “There are four controls that look like they could represent Search functionality. ...” (Team B), “Search icon was repeated and closely copied in three places on the prototype.” (Team D), and “Simple Search: There was confusion with the selections on screen that are search related (Search button, saved searches icon, and advanced search icon).” (Team F). Moreover, some teams mentioned the same problem multiple times using different words.

Both evaluators independently grouped the problems into categories of problems that are essentially the same. The two evaluators integrated their categorization and resolved initial disagreement in consultation. There was high overall agreement among the evaluators: 69 of 86 problems (80.2%) were categorized the same way by both evaluators independently. Seventeen problems were discussed and resolved. The resulting list of unique problem tokens included 36 unique usability problems (see Appendix E). For example, the above-mentioned search related problems were categorized as “Too many controls looked like they could initiate search.” Some teams

mentioned essentially the same problems multiple times using different words. In nine cases, this led to multiple entries for the same usability team in the same category.

Overlap in the Teams' Findings

In Table 2, the numbers of usability problems that were detected by a given number of teams are summarised. Rows show the number of teams that found problems. The second and third columns are the number and percentage of problems that were identified as usability problems in this study. None of the 36 problems was detected by all six teams. Two problems were found by five teams, 4 problems by four teams, 7 problems by three teams, 7 by two, and 16 problems (44.4%) were detected by only one team. The fourth and fifth columns of Table 2 show the frequency and percentage of *all* problems, including the 31 non-usability problems, detected by six, five, four, three, two teams or one team. Because all 31 non-usability problems were mentioned by only one team, the proportion of problems detected by a single team went up from 44.4% to 70.1% (see the second row from the bottom) when the non-usability problems were included.

Table 2

Number of Problems Detected by a Given Number of Teams in this Study Including only Usability Problems vs. all Problems, and in Molich (1999)

No. of Teams	This Study				Molich (1999)	
	Usability Problems		All Problems		All Problems	
	Count	%	Count	%	Count	%
6	0	0.0	0	0.0	0	0.0
5	2	5.6	2	3.0	4	1.4
4	4	11.1	4	6.0	2	0.7
3	7	19.4	7	10.4	13	4.7
2	7	19.4	7	10.4	41	14.8
1	16	44.4	47	70.1	217	78.3
Total	36	100.0	67	100.0	277	100.0

Note. To make the results from Molich (1999) comparable, the data from the two student teams and the team that used a survey method were excluded, leaving six teams in the data set.

Is There More Overlap in the Data of This Study Compared to Molich et al. (1999)?

To make the data from Molich's et al. (1999) study comparable to this study, the data from the two student teams and from the team that used a survey method for their evaluation were ignored, leaving six teams in the data set. For this reanalysis, the raw data provided by Molich (1999) were used. Molich revealed which teams were student teams (R. Molich, personal communication, July 28, 1999). The number and

percentage of problems detected by a given number of teams for the reanalysed Molich (1999) data are shown in the rightmost columns of Table 2.

These reanalysed data were compared to the data of this study. One comparison was made using the 36 usability problems. A second comparison was made using all problems including the 31 non-usability problems.

The percentage of the 36 *usability* problems detected by a single team (44.4%) was much lower in this study than in the reanalysed Molich (1999) data (78.3%). Compared to the reanalysis of the Molich (1999) data, there was a shift away from a small number of teams detecting problems toward a higher number of teams detecting problems. The mean number of teams detecting a problem for the reanalysed Molich (1999) data was $M = 1.32$ ($SD = .73$). This was lower than the mean number of teams detecting the problems in this study $M = 2.14$ ($SD = 1.27$). The mean difference was found significant in a *t*-test comparing the mean number of teams who detected the 36 problems in this study with the mean number of teams who detected the 277 problems in the reanalysed Molich (1999) data, $t(38.06) = 3.79$, $p = .001$. The *t*-test used was corrected for heterogeneity of variances of the two samples, hence the non-integer degrees of freedom.

To determine whether the higher agreement among the teams in the current study was solely a function of the initial removal of non-usability issues, the reanalysed Molich (1999) data were also compared with the data of this study taking into account the 31 non-usability problems that were excluded from the other analyses. The proportion of problems detected by a single team in this study (70.1%) approached that of the reanalysed Molich (1999) data (78.3%). However, the proportions of problems

detected by three or more teams in this study remain higher than in the reanalysed Molich (1999) data. None of the 31 non-usability problems was detected by more than one team. Therefore, the mean number of teams detecting a problem in the current study was lower with than without those problems, $M = 1.61$ ($SD = 1.09$). However, the mean remains significantly higher than in the reanalysed Molich (1999) data, $t(80.90) = 2.08, p = .041$.

Severity Ratings of the Problems

One of the objectives of this study was to look at the agreement among teams on severe and not severe problems. To address that issue, it was necessary to get an estimate of severity for each problem. Three university professors teaching HCI courses served as judges to assess the severity of the 36 usability problems independently using the same four-point scale the teams were asked to use (cosmetic problem, minor usability problem, major usability problem, usability catastrophe). Each judge was provided with a problem description with screenshot and had access to the prototype and the documentation but not to the usability test reports.

Kendall's coefficient of concordance of the raters was $W(35) = .54, p = .01$. Kendall's W is "an index of the divergence of the actual agreement shown in the data from the maximum possible (perfect) agreement." (Siegel, 1955, p. 230) It ranges from 0 (absolute disagreement) to 1 (perfect agreement). Thus, there was a moderate agreement on problem severity among the raters. For the individual ratings see Appendix F.

Following Nielsen's (1993, 1994) recommendations, the severity ratings were aggregated. Because of the ordinal nature of the severity scale, the median of the ratings of the three judges for each problem was used to aggregate the ratings. Of the 36 problems, 18 problems had a median rating of "minor usability problem," 14 had a median rating of "major usability problem", and 4 problems had a median rating of "usability catastrophe." No problem had a median rating of "cosmetic problem."

Is There More Overlap Among More Severe Problems?

The median severity ratings were used to determine whether there is a difference in the overlap between severe and not so severe usability problems. Because of the low number of usability catastrophes (4), the 14 major usability problems were pooled with the usability catastrophes resulting in a group of 18 severe problems. Table 3 shows the data from the first two columns of Table 2, categorized into severe and not severe problems. In Table 3, the absolute frequencies and percentages of the usability problems that were detected by a given number of teams are shown separately for minor problems in columns 2 and 3 and severe problems in columns 4 and 5. The percentage of problems detected by one or two teams is lower, and the proportion of problems detected by three or four teams is higher in the severe problems compared to the minor problems.

Table 3

Number of Minor vs. Severe Problems Detected by a Given Number of Teams

Number of Teams	Minor Problems		Severe Problems	
	Count	%	Count	%
6	0	0.0	0	0.0
5	1	5.6	1	5.6
4	0	0.0	4	22.2
3	2	11.1	5	27.8
2	5	27.8	2	11.1
1	10	55.6	6	33.3
Total	18	100.0	18	100.0

The mean number of teams detecting the 18 severe problems $M = 2.56$ ($SD = 1.34$) was significantly higher than the mean number of teams detecting the 18 minor usability problems $M = 1.72$ ($SD = 1.07$), $t(34) = 2.06$, $p = .047$.

Agreement of the Teams' Answers to the Designers' Questions

Spencer et al. (2000) found that there is more agreement among usability engineers' reports on research questions identified in advance than on after-the-fact lists of usability problems. The answers of the teams to the designers' questions (Appendix B) were extracted from the reports and are shown in Table 4. Question 4 had two parts. It was not possible to determine whether the yes/no answers referred to the first part,

second part, or both. Furthermore, the question was formulated in a leading way. It was, therefore, omitted from this analysis.

Three teams gave explicit answers to all five questions. Team E didn't give an explicit answer to any of the questions. Team C provided the answers on a per participant basis leading to inconclusive information for question 5 because the answer was "yes" for three participants and "no" for the other two. Team F did address question 6 with a test task and reported data but didn't report whether the prototype passed or failed the task. There was moderate agreement on questions 3 and 6, and low agreement on questions 1, 2, and 5.

Table 4

Answers of the Teams to the Designers' Questions

Question	Team					
	A	B	C	D	E	F
1	Yes	No	Yes ^a	No	? ^a	No
2	Yes	No	Yes	Yes	? ^a	No ^a
3	Yes	No ^a	Yes ^a	Yes	? ^a	Yes
5	Yes	No	?	Yes	? ^a	No
6	Yes	Yes ^a	Yes ^a	Yes	? ^a	?

Note. Question marks indicate inconclusive information. ^a There was no test task directly addressing the question.

What Kind of Data Did the Teams Base Their Answers on?

Table 5 gives insight on the basis of the individual teams' answers. Table 5 shows whether each team provided performance data, ease of use data, and user comments on a per task basis, whether there were explicit usability goals, and whether there were tasks to address all the issues directly. Only three teams provided performance data on a per task basis, only three teams provided subjective ease of use ratings on the individual tasks. One team chose not to include user comments. Two teams used explicit usability goals for the decision whether the prototype passed or failed a task. Two teams had tasks in their test protocols that were suitable to directly address all questions. The answers from teams that are not based on a task that directly addressed an issue are marked in Table 4.

Table 5

Information Provided by Each Team

Did the team provide...	Team					
	A	B	C	D	E	F
Performance Data	Yes	No	No	Yes	No	Yes
Ease of Use Data	Yes	Yes	No	No ^a	No ^a	Yes
User Comments	Yes	Yes	No	Yes	Yes	Yes
Usability Goals	Yes	No	No	No	No	Yes
Tasks for all Issues	Yes	No	No	No	Yes	No

Note. ^aThese teams provided overall ease of use data as opposed to ease of use ratings for each task.

Resources the Teams Dedicated to the Usability Test

In the post test questionnaire, the teams provided data about the resources they dedicated to the usability test. The number of members of the usability teams ranged from one to five (Median = 3). The number of test participants ranged from 5 to 12 (Median = 6). The time the teams dedicated to the project varied greatly, ranging from 24 to 146 person hours (Median = 40.75 hrs). Four teams chose a usability testing approach with no video and audio recording. The two teams that did use audio/video recording used the most person hours. One of the teams used a usability lab with one-way mirrors. For the full post test questionnaire data on the resources used by each team see Appendix G.

External Validity Analysis

External validity concerns generalizing from the settings of the study to target settings, in this case usability testing in a corporate context. The external validity of this study can be assessed on the basis of the participants' answers to the questions in the post test questionnaire regarding the deviations from their usual procedure and their assessment on how realistic the study was with regards to size and nature of the project, the prototype, and the task.

Four teams would normally use a three-point severity scale instead of the four-point scale they were asked to use for this study. Four teams said they usually would have more interaction with the design team. However, only one of these four teams made use of the offer to contact the development team by email as did one other team. The majority of the questions the two teams asked were clarification questions, for

example, how a certain feature was supposed to work or whether certain aspects of the prototype were design intent or a bug. The list of all questions asked is shown in Appendix H. For the full list of deviations from the teams' usual procedure see Appendix I.

When asked how realistic the project was in terms of size, nature, and the request of the design team, three teams explicitly said that it was realistic compared to their normal usability testing. One team said it would be among their smaller projects. Another team said that most of their projects were smaller than this project. Two teams said they usually test web sites as opposed to Windows based software and three teams said they usually test projects of internal clients. For the full data see Appendix J.

Discussion

Low Overlap in This Study but Higher Than in Earlier Studies

To better address the reliability issues brought up by Molich et al. (1998, 1999), the modifications in this replication study included using an early prototype, providing a specific and focused client request, having multiple evaluators extract and group the problems, and excluding non-usability issues from the problem list.

As in the Molich et al. (1998, 1999) studies, the agreement among the testing teams on the list of usability problems was low. Not a single problem was reported by all six teams, and only two problems were detected by five teams. A large proportion (44.4%) of the 36 usability problems was detected by only one team. When the full list of issues, including non-usability problems, is considered, the proportion of problems detected by only one team in this study (70.1%) approaches that of Molich et al. (1999) (78.3%). Thus, the concerns about the reliability of usability testing that arose as a consequence of Molich et al. (1998, 1999) cannot be dismissed.

However, in combination, the modifications to this study did result in a higher overlap of findings between teams than in the earlier studies (Molich et al., 1998, 1999). This is not entirely due to the fact that the 31 non-usability problems were excluded because there was still a significant improvement even when those 31 problems that were not usability issues were included.

Emerging Pattern

Integrating the findings of a series of four studies (Molich et al, 1998, 1999; this study; and Spencer et al., 2000), a pattern emerges. More specific and focused requests by the client lead to more overlap in the findings of the usability testing team.

While there was very little overlap in Molich et al. (1998, 1999), more overlap could be demonstrated in this study. When given a set of fully operationalized research questions including definitions of success and time limits for the task completion, usability teams reached perfect agreement on a yes/no scale (Spencer et al., 2000). Spencer et al. (2000) found that there was less agreement on the questions that did not include definitions of success and time limits. As a note of caution, the fact that all five teams in Spencer et al. (2000) answered the two highly operationalized question with “no” (i.e., users could not perform the task within the time limit) could indicate that the time limits might have been unreasonably strict.

More Overlap in Severe Problems

Literature comparing the effectiveness of various usability evaluation methods suggests that one of the relative strengths of usability testing is its aptness to find severe usability problems (e.g., Jeffries et al., 1991, Karat et al., 1992). The results of the present study show that there is more agreement on severe usability problems than on minor usability problems. On average, the severe problems were detected by more teams than the minor problems, indicating more agreement among teams on more severe problems. Nevertheless, there was less than perfect agreement on severe problems.

Agreement Among the Severity Raters

An interesting side issue is the moderate inter-rater agreement of the three severity raters. Nielsen (1994) reports data indicating that judgement based severity ratings have low reliability, leading to his advice to aggregate ratings (Nielsen, 1993, 1994). Kendall's coefficient of concordance between the eleven evaluators serving in a Heuristic Evaluation was $W(39) = .31, p < .01$ (Nielsen, 1994). There was more concordance among the three raters in this study, $W(35) = .54, p < .01$. There are many possible explanations for a less than perfect agreement, including raters' bias towards giving higher or lower ratings, or different strategies when assigning the ratings. For example, one rater made a conscious effort to use all four categories, while the others did not. Some raters might have thought more in terms of the numeric values while others might have thought more in terms of the verbal scale labels.

Instructions might improve agreement on severity of ratings. The raters were not given any instructions about how to rate severity. For example, they were not instructed to consider specific aspects that contribute to problem severity like frequency of occurrence of the problem, impact on use of the dialog box, and persistence of the problem with use when assigning their ratings. One of the raters pointed out that an instruction like this or explicit criteria for the assignment of the different ratings might have led to more agreement.

Potential Reasons for the Low Agreement of the Usability Testing Teams

No negotiation of usability goals. Setting usability goals is recommended as part of the process of usability testing (e.g., Dumas & Redish, 1999; Rubin, 1994). As

Spencer et al. (2000) suggest, there is more agreement among teams when there are clearly defined success criteria. The moderate to low agreement on the questions the designers wanted to have addressed (see Table 4) in the present study could be due in part to the lack of success criteria for the individual questions. Only two teams set their own usability goals. It remains speculative whether the other teams usually negotiate usability goals with their clients, but the lack of direct access to developers made setting usability goals difficult in this study. The study could have obtained more agreement if usability goals had been included as part of the design team's request, but usability goals should be negotiated between clients and usability testers.

Differences in features tested. One of the goals of this study was to ensure that all teams test the same aspects of the prototype by providing a specific and manageable list of research questions. This goal was not achieved. As in Molich et al. (1999), not all teams had tasks that directly addressed all the issues. Teams that didn't test a certain feature couldn't possibly detect problems associated with that particular feature. It is difficult to understand why a team would not address an issue raised by developers, but these "misses" likely depressed the agreement on the list of usability problems.

Differences in task design and order of testing. Another source of disagreement was differences in task design and task order. For example, one problem related to the search feature (i.e., the simple search, not the advanced search which was not to be tested) could only be detected if the participants had to do certain other tasks after the completion of the search task. The three teams who did not have their users perform another task after the search task could not, and did not, find this problem. The other

three teams did. Careful test design based on understanding of issues is essential to success.

Differences in users tested. The instructions were, in hindsight, not specific enough about the target user group. Familiarity with Microsoft Windows and standard office applications was required, but the degree of this familiarity was not specified. Although even power users suffer due to low usability of a design, it is plausible that power users might have less difficulties with a poor design than intermediate users because they might pick up clues that are too subtle or cryptic for less experienced users. All eight participants of one team described themselves as power users of Microsoft Word. The design passed all the performance related usability goals that team set. Another team described their users as “comfortable in a Windows 2000 environment.” The design passed only two out of five performance related usability goals that this team set. Both teams had very similar tasks. While we don’t know whether the users of the second team would describe themselves as power users, it cannot be ruled out that there was a difference in the experience between the two samples of participants that might have contributed to the difference in the findings.

External Validity

In their post test questionnaire answers, four teams said they usually have more interaction with the client. Interaction of the testing teams with the development team might have led to a better understanding of the design and the potential problems with it. Other than that, the teams’ answers regarding their deviations from their usual procedure did not indicate any threats to the external validity of this study. Three teams

said explicitly that the project was realistic in terms of nature, size, and the requests of the design team.

Construct Validity

One possible interpretation for the failure of several teams to include important, standard information in their reports and to address all issues could be that they didn't take this project as seriously as their real work, thus threatening the validity of the construct "*professional* usability testing." There were aspects of some teams' testing and reporting that appear less than professional, but there is no way to know if it was typical. When professional usability testers voluntarily take part in a time consuming study on top of their already high regular workload one might not expect the same quality of work they usually deliver. However, all teams dedicated a substantial number of person hours to this project. Why would a team spend a lot of time on a project and still deliver work that leaves much to be desired?

Answers to the post test questionnaire and informal communications indicate that two teams used this project as an opportunity to train junior staff under the direction of senior staff. While this qualifies as *professional* usability testing, it can't be ruled out that this had an effect on the quality of their evaluation.

Another threat to the construct validity is posed by at least one team mixing evaluation methods. If different evaluation methods were mixed, it is hard to draw conclusions about the construct under scrutiny – usability testing. The data of one team indicate that only a portion of the findings was based on usability testing. This team had only three tasks. One task was of a rather exploratory nature. Yet this team addressed

all the design team's questions and mentioned problems that couldn't possibly come up while performing the tasks used. Their evaluation method might be better characterized as an expert review. Also, without making explicit where the data come from, there is a danger of passing off tester opinions as observed user behaviour.

Implications for the Cooperation Between Usability Testers and Their Clients

The notion that specific and focused requests lead to more reliable results has very clear implications for the cooperation between usability testers and their clients. The client should work closely with the usability tester when identifying and understanding issues to test, designing test tasks, choosing behavioural measures, and deciding on usability goals. In the negotiation process on the test plan and on the usability goals, the usability testing teams provide their expertise and experience and can take the role of an advocate for the users. It is critical for the success of a usability test that the testing team has access to detailed information, especially on the potential issues with a design. Often, only the development team can provide this information. Usability testers shouldn't let the client get away with just dropping off a prototype with the request to find as many flaws as possible. A sign-off of the test plan may help to ensure that the client's questions are addressed.

Implications for the Value of Doing Usability Testing

The finding that there was low agreement on most of the 36 usability problems raises the question whether there should be any resources dedicated to usability testing in the user centred design process. Although there is much to be desired in terms of the

agreement of the teams on the list of usability problems, the data presented in this study do not allow for the conclusion that usability testing is worthless. Every team detected at least six usability problems. Every team detected at least one usability catastrophe and at least three major problems. All teams provided valuable information that would help to improve the design of the open dialogue. Gray and Salzman (1998) point out that “in the cause of usability, doing something is almost always better than doing nothing.” (p. 207) Nevertheless, the results of this study indicate that there is considerable need for improvement in usability testing.

References

- Bevan, N. (1999). Common industry format usability tests. Proceedings of UPA99 (Usability Professionals Association 1999 Conference), 25-30.
- Dumas, J.S. & Redish, J.C. (1999). *A practical guide to usability testing* (Rev. ed.). Exeter, UK: Intellect.
- Gray, W. D. & Salzman, M. C. (1998). Damaged merchandise? A review of experiments that compare usability methods. *Human-Computer Interaction*, 13, 203-261.
- Jacobsen, N. E., Hertzum, M., & John, B. E. (1998). The evaluator effect in usability tests. *Human factors in computing systems CHI 98 Extended Abstracts*, 255-256.
- Jeffries, R., Miller, J. R., Wharton, C., & Uyeda, K. M. (1991). User interface evaluation in the real world: A comparison of four techniques. *Proceedings of the ACM CHI '91 Conference on Human Factors in Computing Systems*, 119-124.
- John, B. E. & Marks, S. J. (1997). Tracking the effectiveness of usability evaluation methods. *Behaviour & Information Technology*, 16, 188-202.
- John, B. E. & Mashyna, M. M. (1997). Evaluating a multimedia authoring tool. *Journal of the American Society for Information Science*, 48, 1004-1022.
- Karat, C.-M., Campbell, R., & Fiegel, T. (1992). Comparison of empirical testing and walkthrough methods in user interface evaluation. *Proceedings of the ACM CHI '92 Conference on Human Factors in Computing Systems*, 387-404.
- Lindgaard, G. (1994). *Usability testing and system evaluation. A guide for designing useful computer systems*. London: Chapman & Hall.
- Molich, R. (1999). Comparative Usability Evaluation – CUE. Retrieved September 20, 1999, from the World Wide Web: <http://www.dialogdesign.dk/cue.html>
- Molich, R., Bevan, N., Curson, I., Butler, S., Kindlund, E., Miller, D., & Kirakowski, J., (1998). Comparative evaluation of usability tests. *Proceedings of UPA98 (Usability Professionals Association 1998 Conference)*, 189-200.

- Molich, R. & Kindlund, E. (2000). *Improving your skills in usability testing*. Notes for a tutorial given at the ACM CHI 2000 Conference on Human Factors in Computing Systems, The Hague, Netherlands.
- Molich, R., Thomsen, A. D., Karyukina, B., Schmid, L., Ede, M., van Oel, W., & Arcuri, M. (1999). Comparative evaluation of usability tests. *Human factors in computing systems CHI 99 Extended Abstracts*, 83-84.
- Nielsen, J. (1993). *Usability engineering*. San Francisco, LA: Morgan Kaufmann.
- Nielsen, J. (1994). Heuristic evaluation. In J. Nielsen & R. L. Mack (Eds.), *Usability inspection methods* (pp. 25 –62). New York: Wiley.
- Nielsen, J. & Landauer, T. K. (1993). A mathematical model of finding usability problems. *Proceedings of the ACM Interchi '93 Conference*, 206-213.
- Rubin, J. (1994). *Handbook of usability testing. How to plan, design, and conduct effective tests*. New York: Wiley.
- Siegel, S. (1956). *Nonparametric statistics for the behavioral sciences*. New York: McGraw-Hill.
- Spencer, R., Miner, T., Gilman, E., Fraser, K., Story, L., & Palmen, H. (2000). Observed sources of variability in usability reports. Manuscript submitted for publication.
- Virzi, R. A. (1992). Refining the test phase of usability evaluation: How many subjects are enough? *Human Factors*, 34, 457-468.

Appendix A: Instructions

Instructions

Thank you very much for contributing to the research for my M.A. thesis in the CURE program at Carleton University.

As you know from the request for participation, we are doing a replication of a study Rolf Molich presented at CHI'99. He had the same product, hotmail.com, tested by nine different usability teams and compared the results. The overlap of detected problems was very small.

For our replication study, we ask you and several other highly experienced professional usability teams to perform a usability test on an early-stage prototype of an open/save dialogue. On the basis of the test reports we will receive, we will determine the overlap of detected problems and compare it with Molich's results.

Here is some more information:

- The amount of resources you dedicate to the project depends entirely on you.
- On the next page, you'll find the request of the client. Please address their concerns and provide data as requested.
- For the sake of external validity, it would be beneficial if you could do the testing as you would do it on a real project.
- The deliverable is a written report. Please use your usual reporting format but don't include the name of your team anywhere in the report.
- Please do not talk to other usability professionals not in your team about this study until we have all the results. They might be participating, too, and we want to gather independent data. I'll let you know when I have all the results.
- The information included in this package should give you a good idea about the product to be tested, the target user group, and the request of the design team. If you have further questions for the design team, please route them through me by email (mkessner@chat.carleton.ca). I'll forward them to the design team and you'll receive the replies through me. This is to spare the design team having to answer duplicate questions multiple times.
- Please keep track of the number of person-hours you dedicate to the usability test (for test preparation & recruiting, the actual testing, data analysis and report generation separately). This is important information. There is a post test questionnaire. Please fill it out after you've finished everything else and send it back to us with the report.
- We are shooting for an end of April completion date so I can complete my thesis and get a job.

Appendix B: Client Concerns

Client Concerns

Your client is working on the design of a new open/save dialogue for possible use in future releases of Windows applications.

- The target user group are people who are familiar with Windows and standard office applications.
- The client hopes that people can use the majority of features of the dialogue without any training or help.

In particular, the designers are interested in data on the following questions:

- Do users find out how to turn on/off the explorer tree?
- Do users find out how to make use of the shortcuts toolbar on the left?
- Do users find out how to edit properties of files?
- Are users thrown off by the unusual location/arrangement of the file name and file type controls when filtering the view for a certain file type?
[usually, both controls are located in the lower part of the box and beneath each other]
- Can users perform a simple search?
- Do users find how to switch between the normal open mode and the advanced search mode?
[Designers are not interested in actually testing the advanced search functionality because it is targeted specifically at the power user group.]
- Additionally, the design team wants to know about any other problems you find.

The client asks you to gather data on objective performance, subjective ease of use, and user comments in a usability test. Furthermore, they are interested in your severity ratings on the problems you've found. They ask you to use the following scale for your severity ratings for each problem. Do not consider how easy or practical it is to fix.

- 1 - cosmetic problem
- 2 - minor usability problem
- 3 - major usability problem
- 4 - usability catastrophe

We have included a short "manual" for the dialogue that should give you an idea of the new features of the dialogue, and also some information about how the prototype works and the limitations of the prototype.

Appendix C: Post Test Questionnaire

Post Test Questionnaire

1. How many people were involved in the project and what were their roles (don't count your test participants)?

2. How many person hours were dedicated to
 - test preparation and recruiting?

 - actual testing?

 - data analysis and writing of the report?

3. Did you use any of the following equipment during the test?

☐ audio recording	☐ video recording
☐ logging software	☐ one way mirrors
☐ other <hr/>	
4. In what ways, if any, did you deviate from your usual procedure (e.g. severity rating scale)?

5. Compared to your normal usability testing, how realistic was this test in terms of size and nature of the project, and the request of the design team?

Appendix D: Excluded Problems

The following 31 problems were considered non-usability problems by the two independent raters and were excluded from some analyses.

User Needs Problems/Feature Requests

A7 3 participants did not rate the shortcut bar as 'useful' or 'very useful'. 3 participants commented that they would find the Shortcut bar more useful if they could customize it (i.e. be able to add/delete folders on the bar).

C2 The intent of the project. Why do we need another Windows open / save dialogue? Wouldn't recommend this as a product.

C4 Participants were looking for a trash can. Would suggest having a trash can where the shortcut items are located.

C11 Lack of a file menu. Would provide a 'File' menu.

C13 Don't know how to add anything to 'Favorites'. The interface should support drag and drop.

C16 'Favorites' should include more than URLs. Include applications and files in the 'favorites' or just put a globe behind the heart on the 'favorites' icon.

C19 Only 'All Files' and 'Windows Bitmaps' listed under file types. Should have more than 'All files' and bitmap files as filters or don't have a file type filter at all.

C20 Can't drag and drop into shortcuts folders. Should support drag and drop.

C24 In the 'Look in' pulldown, there are no lines to indicate what level you are at. Provide lines.

C31 Should be able to have preview on or off. Provide a way of setting preferences like preview.

F1 Users expected to find the My Documents icon in the left hand shortcut toolbar.

Prototype Limitations/Bugs

A5 1 participant commented that the Tooltip for the tree pane 'Show/Hide' button always says 'Show Directory Tree'.

C3 When you change the 'Views', the icon does not change to match the view that you've chosen. Suggest fixing.

C5 Open file button and pulldown menu does not work. Is either a prototype limitation or a software bug.

C7 'Create new folder' doesn't work. Is either a prototype limitation or a software bug.

C15 Getting a 'Fileopen' error when initially double-clicking on a file to open. Is a software bug.

C18 No history list under 'File Name'. Is either a prototype limitation or a software bug.

C22 'File type' filter does not work. Is either a prototype limitation or a software bug.

C23 When you choose the shortcut 'Work Folder', it says 'Templates' in the 'Look in' field. Shortcut names should match in all locations.

C28 When you choose the shortcut 'Recent Files', it only says 'Recent' in the 'Look in' field. Shortcut names should match in all locations.

C30 When in 'Details' view, sorting by columns only works in ascending order but not descending order. Should sort in both ascending and descending order.

C32 Sort doesn't work. Is either a prototype limitation or a software bug.

D8 For all Search results, the icon is changed and the file cannot be opened by any means. Locating files using other means (Look in, Shortcut, back) did not result in the same problems; file icons were correct, and the files could be opened with a double click of the mouse.

D9 Back works fine for folders and drives, but does not display previous search results.

D10 The "Look in" field does not register location changes when the "one level up" button is depressed. If the user is looking at the "Look in" to ascertain which area she/he is searching, then the search will be incorrect as the display in the file box no longer matches the 'Look in' or explorer tree.

Other

C25 Having 'Sort' in Views is not expected. Would expect to find in a main menu that would pop up a dialogue box. Move the sort function location.

C29 What appears in 'File Name' is not what is expected unless it is a typed in search. Participants expected when they clicked on a file to have the text from the last search in the 'File Name' field be replaced. Should provide this functionality.

D2 Default view not 'List' as in most Windows applications.

D7 "Look in" has no dropdown.

D11 Starts in My Computer when prototype is launched. Current Windows applications start in the My Documents folder.

E17 Advanced search was not tested but appears complicated to users.

Appendix E: Included Problems

This is a list of all usability problems reported by the six teams. Below the italicized group labels of the 36 unique usability problems, the problems in the original wordings as found in the reports are listed. The identifier of each problem contains the code of the team that found the respective problem.

(1) Undesired association of file type dropdown with properties pane due to proximity. Users relate the two widgets functionally, e.g. expected filtering results appear in the properties pane.

A4 3 participants were not clear what the pane was for under the 'File Type' dropdown (in Properties view). 2 participants thought that the pane had something to do with the 'File Type' dropdown since it was located below it. 1 participant thought that when a file type was filtered for, all of those files would show up in this pane.

D16 The arrangement of the [File Type/File Name] controls did cause confusion regarding the proximity of File Type to the Properties window. File Type's proximity to the Properties window caused participants to relate them functionally. One participant commented during the general feedback portion of the study that he "liked the File Type window" while pointing at the Properties window.

F6 Displaying Specific File Types. 2 of 4 users who successfully completed this task, expected results to show in the properties view pane below. This problem exists because the File Type dropdown list lines up exactly with the Properties view pane.

(2) Functionally related entry fields and command buttons are locationally disconnected (in general). This doesn't support the workflow.

C33 Buttons, widgets and fields are not laid out in a way that supports the tasks. Requires a redesign.

D20 Layout suboptimal ("Recommendations: ... Better cluster like functions ... and related items.", p. 2, exec summary).

(3) File name field (where the search terms are entered) and search button are locationally disconnected. Doesn't support task flow, makes grammar of simple search unintuitive.

C8.2 Search button and file name field too far apart. Locate the simple search button and pulldown menu closer to where the search field is.

D3 A general disconnect exists between similar functioning buttons and the applicable fields. For example, the *Search*, *Open*, and *Cancel* buttons are not locationally related to the *File Name* and *File Type* fields. [Team D has an entry in the "general" category already]

F3.1 Performing a Simple Search. 4 out of 6 users could not complete the task by themselves. Generally, users knew where to input data, but did not know how to start

the search. There seemed to be a too distant spatial relationship between where you input data to be searched and the Search button.

(4) File name and file type fields are locationally disconnected. Doesn't support task flow.

C9 Positioning of File Name / File type does not support a good task flow. Would suggest a redesign, possibly stacking the File Name and Type vertically.

(5) Icons and buttons are all over the place, causing confusion.

C10 Positioning of icons and buttons. Relocate so that everything is not so spread out. 'Open', for instance, could be in a menu and not be a button.

E4 Categorization of buttons with a group on top, a group on the bottom and various groups vertically places on the left of the screen is difficult to understand.

(6) Panes can't be resized, sometimes leading to the obstruction of information.

A1 5 participants commented that they would like to be able to re-size the different panes in the dialog.

C1 Can't resize anything. Makes navigating through the tree very difficult. Support re-sizing.

C6 Unable to change the width of boxes or change or minimize the size of the window. Provide this functionality.

C27 Unable to tell the hierarchy of the file location. Redesign the file properties view or at least allow resizing of the window. Alternatively, eliminate file properties view. File properties doesn't add any value as you already know the file name and you can check 'Look in' field for the location.

D6 The prototype cannot be resized. On a 15-inch monitor using 'normal' resolution (800X600), part of the prototype is off the screen. (Note: This is also a problem in the current Window's version of the Open/Save dialogue.)

E16 Users expect to be able to conduct standard window operations such as minimize window and resize window and resize information boxes (such as tree and file properties).

(7) Icons do not look like active buttons (generally).

D4 Icons do not look like buttons. User must roll-over them with the mouse to recognize that they are "live".

(8) Icons are hard to identify (generally).

E2 Icons are difficult to identify without the use of tool tips.

(9) Shortcut icons are hard to see.

C37 Some of the icons are a bit fuzzy (such as the 'Desktop' one). Suggest fixing.

E6 Shortcut icons are difficult to see and interpret.

(10) Up button/icon is not intuitive.

B2 Up button/icon is not intuitive. An alternative function exists.

(11) Views button/icon is not intuitive.

B4 Views button/icon is not intuitive. There is a problem discovering this function, but once users discover it, they remember what it does. However there is a question as to whether or not the users will discover "Views" at all.

(12) Launch Browser button/icon is not intuitive.

B5 Launch Browser button/icon is not intuitive. An alternate function exists, but it is unclear to users what the button/icon actually does.

(13) Tools button/icon is not intuitive.

B6 Tools button/icon is not intuitive. Once users discover it, it should not be a long-term problem.

(14) Desktop Shortcut Button/Icon is not intuitive.

B7 Desktop button/icon is not intuitive. An alternate function exists.

(15) Work Folder Shortcut Button/Icon is not intuitive.

B8 Work Folder button/icon is not intuitive. Users had no idea as to the function of this icon (see the comments in the table on page 7 - Left side buttons/icons).

D14.4 Familiar icons were easily identified by name and function (Desktop and My Computer), other icons were somewhat less clear (Favorites, Searches), and still others were not useful at all (Recent Files, *Work Folder*).

E7 Work folder shortcut is difficult to understand.

(16) Recent Files Shortcut Button/Icon is not intuitive.

B9 Recent Files button/icon is not intuitive. Once the users discover it, it should not be a long-term problem.

D14.3 Familiar icons were easily identified by name and function (Desktop and My Computer), other icons were somewhat less clear (Favorites, Searches), and still others were not useful at all (*Recent Files*, *Work Folder*).

(17) Favourites Shortcut Button/Icon is not intuitive.

D14.1 Familiar icons were easily identified by name and function (Desktop and My Computer), other icons were somewhat less clear (*Favorites*, *Searches*), and still others were not useful at all (*Recent Files*, *Work Folder*).

(18) Searches Shortcut Button/Icon is not intuitive.

B10 The Searches button/icon in the left shortcuts bar is not intuitive. Once the users discover it, it should not be a long-term problem.

D14.2 Familiar icons were easily identified by name and function (Desktop and My Computer), other icons were somewhat less clear (Favorites, *Searches*), and still others were not useful at all (Recent Files, Work Folder).

(19a) Explorer tree button is difficult to discover. It doesn't look like an active button.

B11.1 Users did not notice the Show Directory Tree button. It doesn't look like a button (possibly due to placement).

C26 None of the participants were able to tell what the function of the tree icon was until they clicked on it and then it became obvious. Make the tree look more like something that is selectable. Reconsider its location.

D13.1 In general, the explorer tree's design and placement do not facilitate its usage, though the functionality is considered useful. Only 2 of 6 participants were able to locate the explorer tree without assistance – one of the successful participants clicked on the icon during another task.

E8.1 Directory tree icon is difficult to *discover* and understand.

F4.1 Turning on the Tree View. 3 of 6 users could not find the way to turn on the tree view without hints. Most users thought the tree image was an icon. All the users started with pressing the icons in near proximity to the "Look in:" dropdown list.

(20) Explorer tree icon is difficult to understand.

B11.2 Show Directory Tree icon is not meaningful.

C17 The directory tree is discoverable but the purpose is not apparent from the icon. Once participant thought it was a company logo. Suggest icon redesign.

D13.2 Several participants commented on the design, stating that the icon "was ugly" or "looked like a logo."

E8.2 Directory tree icon is difficult to discover and *understand*.

F4.2 2 of 6 users had never heard of the term "tree view", therefore, the tree icon would be irrelevant to them.

(21) Participants had difficulties closing the explorer tree.

A8 3 participants had difficulty closing the Explorer tree pane. 2 participants clicked on the 'blue left arrow to try and close the pane. 2 participants commented that they would like to see the Show/Hide Tree button closer to the tree pane when it is open. 2 participants commented that they would like to see the Show/Hide Tree button above the tree pane. 1 participant commented that they would not have been able close the pane without help.

E13 Some users experienced difficulty getting out of tree mode by not realizing that the button is a toggle.

(22) Participants had difficulties discovering how to enter notes in the properties view.

B14 Users did not discover how to edit properties of files.

D15 In Windows, users can see properties but not edit them. Based on the findings in this study, the Notes may not be something that a user would consider “editable” when using the Open/Save Dialogue box in a non-testing situation. 4 of 6 participants completed the task without assistance. One participant commented that it did not occur to him that he might be able to enter information it was difficult to see the Notes field.

E11 Users experienced some difficulty discovering the ability to enter notes about a file in the file properties. The design of the notes field does not suggest that data can be entered.

(23) The label “Properties” is used for two different functions: properties pane and the properties entry in the context menu, bringing up a different dialogue box.

D12 Properties (under Tools) are not the same as in the Properties window of the prototype. Another window is launched – the same as when a desktop icon is right-clicked. Usability may suffer since the labels are the same, but the function is not.

(24) Too many controls look like they could initiate a search.

B13 There are four controls that look like they could represent Search functionality. Having four controls (Search Mode button/icon, Searches folder, Search button - at the bottom, and the Launch Browser - button/icon at the top) could potentially create a constant problem for users.

B16 Users did find performing a search difficult (see comments for Task # 5). Four out of six users clicked on the advanced search mode button/icon to perform a simple search.

D1 Participants commented that the Search Mode icon was misleading, that it indicated “search” but not advanced search and mistook the Searches folder on the shortcut bar for a Search Mode.

D19 (Search) icon ... repeated (and closely copied) in three places on the prototype.

E9 Distinction between search button and advanced search mode is confusing.

F3.2 Simple Search: There was confusion with the selections on screen that are search related (Search button, saved searches icon and advanced search icon).

F7 Same search icon used for 2 different functions. 1 out of 6 users observed this problem. The same icon is used for the Advanced Search mode (upper left hand corner) and the run a search in the Search button dropdown list. Though only 1 user noticed this, it was felt this problem could of added to the other users confusion and lack to complete the Simple Search task.

(25) *The (conceptual) difference between simple and advanced search is difficult to understand or separate. Confusion is introduced by the redundancy of search methods.*

C8.1 Difference between simple and advanced search not obvious to participants.

C35 User has to change their behaviour in order to adapt to the interface. Navigation is cumbersome. Participants were unable to figure out the *difference between the simple search and the advanced search*. 'Open Mode' is not meaningful. Would need a fairly major redesign. Would separate the browse mode from the search mode.

D18 Generally, participants had *difficulty separating the Simple Search (or Open Mode) functions from the Advanced (or Search Mode)*. When asked to complete a simple search (no limiting criteria), two participants were confused by the idea of searching and found need to enter the Advanced Search mode. Ratings: Participants were "neutral" about the location of the search functions, but found them "very difficult" to use. (4 of 7 participants rated the use of the search functions as "very difficult"). (Scale: 1 – very easy, 4 – neutral, 7 – very difficult)

E1 Distinction between advanced search and simple search is confusing, particularly with two different methods of entering these searches and their distance from each other on the screen.

E10 Redundancy in search methods can be confusing and may interfere with the ability to perform tasks.

(26) *Hitting the enter key is expected to start a search (after entering a search term in the file name field) but does not do so.*

A3 5 participants entered a search string in the 'File Name' text box and then pressed the Enter key on the keyboard.

C12 Doing a carriage return does not initiate the search. Should be able to type in your search criteria and hit carriage return to initiate your search.

D5 If a value is entered in the 'File Name:' field, the enter button does not activate the search. Only clicking the Search button does so.

E3 Users should be able to hit return to initiate a simple search. I presume this was a simulation issue rather than a design issue.

(27) *Clearing the search is required after performing a simple search but it is hard to find how to do that.*

B1 Users can't find the Clear Search option. When users conduct a search, they do not realize they need to clear it to conduct another search or to use other functions. Users do not tend to discover the search features available via the drop down.

B3 The modal approach to Search causes confusion. Users get stuck in "Search Mode", when conducting a search. There is no indication on the UI as to which mode users are in. Users may be frustrated due to the modal approach and may not be able to fully understand it.

D17 Clear Search function was completely not intuitive due to its location in the Search button picklist and the fact that this concept does not exist in the current Windows dialogue box. Participants had problems with the Clear Search function throughout the study. In most cases where they needed to continue to another task or needed a new search to continue within a task, the participants expected to be able to click in File Name and execute a new search.

F5 Performing multiple simple searches. 5 out of 6 users did not realize they needed to clear a search before performing another. The main reason why this difficulty occurred was because after performing a simple search, the search button is greyed out so users did not even know they could access it's pulldown list. Most users felt that the search should automatically clear when you input a new term to search for.

(28) Switching between open and advanced search mode is difficult because of the unintuitive switching widget.

B15 Three out of six users didn't associate the [search mode] button/icon as search function or a mode. Two users thought it provided a way to view a file or a document. The other user identified it as a "Preview" function, meaning it would provide a preview of the first page of a document.

C8.3 On rollover for 'Search Mode' icon, have 'Advanced Search Mode' instead of just 'Search Mode'. ...

E5 Open Mode and Search mode icons are difficult to understand even with the use of tool tips.

E14 Some users experienced difficulty getting out of advanced search mode by not understanding the relationship between open mode and search mode.

F2 Switching between Advanced Search mode and Open mode. 5 of 6 users did not understand that the upper left hand buttons controlled the modes. There seemed to be confusion with the Advanced search button in that users expected it to perform an action rather than change the view.

(29) Users activate a simple search by pressing the search button (without entering a search term in the file name field), expecting to get a search dialogue.

A2 7 participants did not rate the simple search (task 5) as 'easy' or 'very easy'. 5 participants clicked on the Search button in order to activate the search dialog. 4 participants thought that clicking on the Search button or selecting Search from the Search dropdown menu would open a Search dialog in which they could type in their search string. 3 participants did not realize where they could type in a search string in Open mode.

E12 Clicking search button without a file name entered should prompt the user to enter a file name. Users would assume that the search feature was not operational.

F3.3 Users expected a separate search dialog box to appear when the Search button at the bottom of the dialog box was selected.

(30) The file name field is not labeled as a search field.

C14 Nothing to tell you the 'File name' is a search field. Would separate browsing and searching tasks.

(31) Stopping a search is hard to find.

C34 Method of stopping a search not easily accessed and is not easily discoverable. Have search button change to stop search so that it does not require pulling the menu down to stop the search.

(32) Feedback on where search is performed and items are found is hard to find.

C36 Information of where the file that was searched for and found is missing. Provide better search feedback.

E15 Users experienced difficulty by conducting a simple search unknowingly through the wrong folder. They experienced difficulty understanding the problem and assumed the file did not exist.

(33) Participants are lost in the hierarchy of the file system.

A6 2 participants commented that they were unclear as to where they were in the directory hierarchy when navigating through the directories in the middle pane, if the tree pane was closed.

(34) The preview button is not intuitive.

B12 The Preview button/ icon in the lower left side of the dialog box is not intuitive. An alternate function exists ("Preview" in the "Views" drop down), but it is questionable if users will discover "Views" at all.

(35) Participants unintendedly close the dialogue with the Cancel button when they want to abort initiated commands.

E18 The cancel button is expected to cancel initiated commands and occasionally users would unexpectedly close the window. Users expect cancel to be the furthest button on the right, particularly since help is not considered to be particularly useful.

(36) There is no feedback on which shortcut location is active.

C21 No visual reinforcement that you've selected a shortcut. Once you have selected a shortcut, its visual appearance should change so that it looks selected.

Appendix F: Severity Ratings

Problem	Rater A	Rater B	Rater C	Median
1	2	3	2	2
2	3	2	2	2
3	3	3	2	3
4	2	2	2	2
5	3	2	2	2
6	3	3	4	3
7	3	2	1	2
8	3	2	1	2
9	2	2	1	2
10	2	2	1	2
11	3	2	1	2
12	2	2	1	2
13	2	2	1	2
14	2	2	1	2
15	3	2	1	2
16	3	2	1	2
17	3	2	1	2
18	3	3	1	3
19	3	3	4	3
20	3	2	2	2
21	3	2	1	2
22	2	3	3	3
23	3	3	3	3
24	4	3	3	3
25	4	2	3	3
26	2	4	4	4
27	4	3	4	4
28	4	3	1	3
29	3	4	1	3
30	4	3	4	4
31	3	3	3	3
32	3	3	4	3
33	3	4	1	3
34	2	2	3	2
35	3	4	1	3
36	4	3	4	4

Note. The following rating scale was used: 1 – Cosmetic Problem. 2 – Minor Usability Problem. 3 – Major Usability Problem. 4 – Usability Catastrophe.

Appendix G: Post Test Questionnaire, Resources Dedicated

	Team					
	A	B	C	D	E	F
Team Members	3	4	1	2	3	5
Person Hours, Total	42.5	164	25	39	24	83.5
- for Test Preparation and Recruiting	23	—	10	10	10	22.5
- for Actual Testing	8	—	7	15	7	23
- for Data Analysis and Report Writing	11.5	—	8	14	7	38
Audio Recording	No	Yes	No	No	No	Yes
Video Recording	No	Yes	No	No	No	Yes
Logging Software	No	No	No	No	No	No
One-way Mirror	No	No	No	No	No	Yes
Other Equipment	No	Yes ^a	No	Yes ^b	No	No
Number of Test Participants ^c	8	6	5	6	12	6
Number of Test Tasks ^c	6	7	3	13	9	5

Note. Team B did not provide a breakdown of their person hours. The person hours of team F displayed here are based on incomplete logs and therefore are underestimating the actual person hours. ^a Scan converter. ^b Cue cards. ^c The information about the number of test participants and the number of test tasks was compiled from the test reports rather than the post test questionnaire.

Appendix H: Teams' Email - Questions for the Designers

Two teams made use of the offer to ask the design team for additional information by email. Their questions are listed below.

A I was wondering if you could tell me how you edit properties of files? Is it just a matter of clicking on Properties in the View menu (thus getting the properties pane), or do you have to click on Properties in the Tools menu (thus getting properties dialog popping up)?

A The designers mentioned that they would like us to test to see if users can edit properties... do you know which one they would like us to find out about....edit attributes or edit notes?

A As well, the filtering for a particular file type doesn't seem to be working...is this a known bug or am I just not doing it right. I am assuming that all I have to do is go to the File Type drop down (near the top right of the dialogue) and select a particular file type and the display will be filtered for this. Is this correct?

A Lastly, I order to switch to Advanced search mode, is it just a matter of clicking on the Search mode button in the top left corner and having the search pane pop-up at the bottom of the screen?

A Can you access the Search mode (from the Open mode) by going down to the Search dropdown and selecting New Search? It doesn't seem to be working for me but I just wanted to check just in case participants try this route. Is this possible or is the only way to go to Search mode through the button on the left side of the dialogue.

A Did the designers mention what resolution they would like the dialogue tested at? Would they prefer 800 by 600, or is 1024 by 768 alright?

B Could the development team provide any information about the context in which a user might perform a search and, for example, want to save (the outcomes of) a search? (we'd like to be able to provide participants with realistic test scenarios)

B We've noticed something and want to know if it's design intent or a bug. When you navigate down from the Look-in tree the file selection window contents are the contents of what is high-lighted in the Look-in tree, BUT when you click the icon for navigating one folder up, the file selection window contents change but the folder highlighted in the Look-in tree does not. Is this design intent?

Appendix I: Post Test Questionnaire, Teams' Deviations From Their Usual Procedure

In what ways did you deviate from your usual procedure (e.g. severity rating scale)?

A Severity ratings – we usually use a rating scale of High, Medium, and Low (i.e. a scale of 3 vs. a scale of 4 used in this test). We usually have full access to the prototype designer – allows us to receive a demo of the prototype, and ask any questions we have (i.e. is a particular action, a bug, or actual functionality. Sign-off from stakeholder – normally we request sign-off from the stakeholder (i.e. requester of test) prior to the first test session in order to ensure that all of their questions are being answered. Normally we do not require internal participants to sign an NDA (Non-Disclosure Agreement)

B The only deviation from the usual procedure is the use of the severity rating scale.

C I would normally have a better understanding of the design intent and any constraints that the developers were working around. I normally classify things as major, minor or bugs. Bugs are unintended and by nature have to be fixed by the developers. I strongly recommend that major usability problems be fixed as they impede the user's ability to use the software as intended. I also recommend that minor usability problems be addressed but given time constraints, can be left until the future releases of the software.

D We normally evaluate material for our company web-site and not software as in this case. There is usually one usability person who oversees from 6-30 projects, thus no project with complete attention, but in this case, the junior staff person devoted a lot of attention to thoroughly understanding the prototype and running the study. We run sessions in 50min blocks. There is usually 1-3 products being evaluated in the 50min, thus about 10-30 minutes per product. For this study we used the entire 50min for one product. We usually look for only one or a few main things, sometimes doing comparisons of possible designs. For this study, we were doing more exploratory work. We usually work with a business owner who helps identify areas of concern and who has a vested interest. In this case, we were not sure of the owner and their motives and major concerns. Most of the time, we have an outside contractor who moderates the study and writes the scripts after discussing the areas of concern with the usability person. In this case, we wrote the script and ran the study. We usually have real customers as participants. For this study, we used company staff (they actually use the Windows products, so a good match, but not our normal pool). A great deal of time and effort was put into writing the report. Normally our contractor writes a brief summary 1-2 days after the study and sends the report within a week. Our normal reports address the questions identified before the study and are normally quite short. This report was much longer than normal and more open-ended in its scope. We don't normally use rating scales. Rating scales were used in this report (and I must admit, I

goofed when I was reviewing the test material and didn't thoroughly enough evaluate the scales used in this study - some can't be analyzed because of incorrect wording or use of a scale that was not evenly bi-polar). We often use material that are screen shots or not interactive, but in this study, the prototype worked (more or less). I have to explain our use of the outside contractor. This contractor has been running studies since before the usability staff were hired. They often worked directly with the business end, the product owners and ran studies according to the product owner's wishes. Since the usability staff have been hired, we work with the product owners and suggest evaluation techniques and identify the main areas of concern. We also outline what questions need to be asked in the study, but often still leave the writing of the actual usability script to the contractor. Since we've been hired, more evaluation methods are used, in addition to lab studies, and the lab studies are used more judiciously. Some business owners used to use the lab studies to justify designs or to predict acceptance, rather than to identify problems. We also run studies ourselves, and some staff write their own scripts, but most of the studies are still moderated by a contractor.

E My typical severity scale is a 3 point scale with major (return product, can't conduct important tasks, must be fixed), moderate (interfere with ability to use product, can't conduct less important tasks, may not return product but interfere with using it, should be fixed), and minor (Can still perform tasks but frustrating and irritating or better design choices, address if possible). Typically would conduct a 1 hour interview with the design team to prompt them in capturing issues that can be addressed in testing and to understand design rational and alternative design choices being considered. Typically would review test plan with designers to ensure issues are being captured sufficiently. Typically would not require involve an assistant. Typically would videotape sessions to capture data that may be missed in recording, to assist in the presentation of the results and to allow designers the opportunity to view a sample of sessions. Typically would invite designers to view sessions.

F All of our usability testing is done on projects being developed internally - we do not offer usability testing for external clients. Because of this, much of our testing is done less formally with a general results report.

Appendix J: Post Test Questionnaire, How realistic was this test?

Compared to your normal usability testing, how realistic was this test in terms of size an nature of the project, and the request of the design team?

A This project was very representative in terms of size and scope to our daily work.

B The dialog box interface usability testing (as compared to our normal usability testing) was realistic in terms of size, nature of the project, and the request of the design team.

C This test was quite realistic but as I've said, I would have a better idea of the design intent and constraints. I would also have a better idea of what should or should not be working in the load and when I made recommendations, I would know how far I could go in terms of recommending re-designs.

D In some ways, this project was larger than most we do. In our regular work, we often evaluate products that are the equivalent of 1-2 web pages or simply the wording in one area of a web page. Although in our regular work, we also evaluate products that are much, much larger than this one was. Some of the products we are working on include complete on-line courses or sophisticated software that run the equivalent of 20-30 web pages with a lot of interactive software and/or interactions with our database. For the junior staff person who did most of the work, this was a much larger product than he normally deals with. One of the biggest differences was the lack of contact with the business owner. We always work with a business owner and can ask questions like "what is the ultimate purpose of this product" or "what are your usage goals". We never work on a product without a meeting with the business owners first. This study was more like something a contractor might have to do, than an in-house usability staff such as ours. The product itself was unlike any that we work on, as we do not normally evaluate Windows products. On the other hand, the prototype was a product that we were all familiar with as Windows users.

E This would be considered a smaller project compared to my typical usability tests. The opportunity to discuss issues with the designers which did not occur in this project would have expanded the test to be more in line with my usual usability tests. That is, I would have taken this opportunity to expand the scope of the project, perhaps by comparing design alternatives or comparing to competitor open/save designs. I did expand the scope of the project to address issues that were not specified in the list of issues presented by designers but which I thought would be useful to designers (I.e. feature evaluation, first impressions, situational evaluation) because that is what I would have done in this situation. I would have went further if I had had the opportunity to discuss the test with designers.

F Most of our testing is done on web interfaces, not PC applications. Our testing is done most of the time with specific users more related to the application.