

UNIVERSITY OF CALIFORNIA,
IRVINE

Large-Scale Collection of Application Usage Data and User Feedback
to Inform Interactive Software Development

DISSERTATION

submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in Information and Computer Science

by

David Michael Hilbert

Dissertation Committee:
Professor David Redmiles, Chair
Professor David Rosenblum
Professor Jonathan Grudin

1999

© David Michael Hilbert, 1999.
All rights reserved.

The dissertation of David Michael Hilbert is approved
and is acceptable in quality and form
for publication on microfilm:

Committee Chair

University of California, Irvine
1999

DEDICATION

To

my wife and friend Sara Armstrong

for her spirit, love, patience, and encouragement,

and my parents Robert and Angela Hilbert

and brother Daniel

for their love and encouragement.

TABLE OF CONTENTS

	Page
TABLE OF CONTENTS	iv
LIST OF FIGURES	viii
LIST OF TABLES	xi
ACKNOWLEDGMENTS	xii
CURRICULUM VITAE	xiii
ABSTRACT OF THE DISSERTATION	xvii
CHAPTER 1: Introduction	1
1.1. Improving the Design-Use Fit	1
1.2. Limitations of Current Techniques	3
1.2.1. Usability Testing	3
1.2.2. Beta Testing	4
1.2.3. Early Attempts to Exploit the Internet	6
1.3. Research Overview	9
1.3.1. Goals	9
1.3.2. Hypotheses	9
1.3.3. Approach	10
1.3.4. Evaluation	11
1.3.5. Contributions	12
1.4. Structure of the Dissertation	13
CHAPTER 2: Background	15
2.1. Definitions	15
2.2. Types of Usability Evaluation	17
2.3. Types of Usability Data	20
CHAPTER 3: Nature of UI Events	22
3.1. Spectrum of HCI Events	22
3.2. Grammatical Structure of Events	25
3.3. Contextual Issues in Interpretation	27
3.4. Composition of Events	30
CHAPTER 4: Related Work	34
4.1. Extracting Usability Information from User Interface Events	34

4.2. Synchronization and Searching	41
4.2.1. Purpose	41
4.2.2. Examples	42
4.2.3. Strengths	44
4.2.4. Limitations	44
4.3. Transformation	45
4.3.1. Purpose	45
4.3.2. Examples	46
4.3.3. Strengths	48
4.3.4. Limitations	49
4.4. Counts and Summary Statistics	49
4.4.1. Purpose	49
4.4.2. Examples	50
4.4.3. Related	51
4.4.4. Strengths	52
4.4.5. Limitations	52
4.5. Sequence Detection	52
4.5.1. Purpose	52
4.5.2. Examples	53
4.5.3. Related	58
4.5.4. Strengths	58
4.5.5. Limitations	59
4.6. Sequence Comparison	59
4.6.1. Purpose	59
4.6.2. Examples	60
4.6.3. Related	62
4.6.4. Strengths	62
4.6.5. Limitations	63
4.7. Sequence Characterization	64
4.7.1. Purpose	64
4.7.2. Examples	64
4.7.3. Related	66
4.7.4. Strengths	66
4.7.5. Limitations	66
4.8. Visualization	67
4.8.1. Purpose	67
4.8.2. Examples	68
4.8.2.1. Transformation	68
4.8.2.2. Counts and summary statistics	69
4.8.2.3. Sequence detection	72
4.8.2.4. Sequence comparison	72
4.8.2.5. Sequence characterization	73
4.8.3. Strengths	73
4.8.4. Limitations	74
4.9. Integrated Support	74

4.9.1. Purpose	74
4.9.2. Examples	74
4.9.3. Strengths	75
4.9.4. Limitations	76
4.10. Summary of the State of the Art	77
4.10.1. Challenges and Future Directions	79
CHAPTER 5: Approach	81
5.1. Example	81
5.2. Implementation	85
5.2.1. Basic Services	85
5.2.2. Event Service	87
5.2.3. Agent Service	89
5.3. Usage Scenario	94
CHAPTER 6: Problems and Solutions	101
6.1. Problems	101
6.2. Abstraction	103
6.2.1. Abstract Interaction Events	103
6.2.2. Relating Data to User Interface Features	106
6.2.3. Relating Data to Application Features	108
6.2.4. Relating Data to Users' Tasks and Goals	109
6.3. Selection	111
6.3.1. Selecting Events	111
6.3.2. Selecting Event Contexts	113
6.3.3. Selecting State	114
6.4. Reduction	114
6.4.1. Reducing Event Data	114
6.4.2. Reducing State Data	119
6.5. Context	123
6.5.1. Incorporating User Interface State	123
6.5.2. Incorporating Arbitrary State	124
6.5.3. Incorporating Application State	127
6.5.4. Incorporating Artifact State	128
6.5.5. Incorporating User State	128
6.6. Evolution	130
6.7. Interrelationships and Dependencies	135
CHAPTER 7: Methodological Considerations	138
7.1. Theory of Expectations	138
7.2. Data Collection	140
7.3. Data Analysis	143
7.4. Data Interpretation	145
7.5. Process Integration	148
CHAPTER 8: Evaluation	151

8.1. NYNEX Corporation: The Bridget Project	151
8.1.1. Overview	151
8.1.2. Description	152
8.1.3. Results	152
8.2. Lockheed Martin Corporation: The GTN Scenario	154
8.2.1. Overview	154
8.2.2. Description	154
8.2.3. Results	156
8.3. Microsoft Corporation: The “Instrumented Version”	157
8.3.1. Overview	157
8.3.2. Description	157
8.3.3. Results	158
8.3.3.1. How Practice can be Informed by this Research	159
8.3.3.2. How Practice has Informed this Research	160
CHAPTER 9: Challenges and Future Directions	164
9.1. Maintenance	164
9.2. Authoring	167
9.3. Privacy and Security	168
9.4. User Involvement	169
9.5. Post-Hoc Analysis	171
9.6. Possible Areas of Cross-Pollination	171
CHAPTER 10: Conclusions	175
10.1. Conclusions	175
10.2. Summary of Contributions	176
10.3. Other Potential Applications	177
REFERENCES	179

LIST OF FIGURES

	Page
Figure 3-1. A spectrum of HCI events	23
Figure 3-2. A multi-level model of events	31
Figure 4-1. Related work comparison framework	38
Figure 4-2. Synchronization and Searching	39
Figure 4-3. Transformation	39
Figure 4-4. Counts and Summary Statistics	39
Figure 4-5. Sequence Detection	40
Figure 4-6. Sequence Comparison	40
Figure 4-7. Sequence Characterization	40
Figure 4-8. Visualization	41
Figure 4-9. Integrated Support	41
Figure 4-9. Visualizing the results of event selection	68
Figure 4-10. Visualizing the results of event abstraction	69
Figure 4-11. Relative command frequencies ordered by “rank”	70
Figure 4-12. Relative command frequencies over time	70
Figure 4-13. Mouse click location and density	71
Figure 4-14. Visualizing the results of automated sequence alignment	72
Figure 4-15. A process model characterizing user behavior	73
Figure 5-1. A simple word processing application	82
Figure 5-2. “File Menu” agent	82
Figure 5-3. “File Menu” data	83
Figure 5-4. “All Menus” agent	84
Figure 5-5. “All Menus” data	85
Figure 5-6. Basic services	86
Figure 5-7. A single-triggered agent	90

Figure 5-8.	A dual-triggered agent	91
Figure 5-9.	Agent algorithm	94
Figure 5-10.	A prototype database query interface	96
Figure 5-11.	Agent authoring interface	97
Figure 5-12.	Agent notification and user feedback	98
Figure 6-1.	“Use Text” agent (abstract interaction events)	104
Figure 6-2.	“Use Non-Text” agent (abstract interaction events)	104
Figure 6-3.	“Use New” agent (abstract interaction events)	104
Figure 6-4.	“Value Initial” agent (abstract interaction events)	105
Figure 6-5.	“Value Provided” agent (abstract interaction events)	105
Figure 6-6.	“File Menu” agent (relating data to user interface features)	106
Figure 6-7.	“Edit Menu” agent (relating data to user interface features)	107
Figure 6-8.	“File Toolbar” agent (relating data to user interface features)	107
Figure 6-9.	“Edit Toolbar” agent (relating data to user interface features)	107
Figure 6-10.	“Print Window” agent (relating data to user interface features)	108
Figure 6-11.	“File->New” agent (relating data to application features)	109
Figure 6-12.	“File->Print” agent (relating data to application features)	109
Figure 6-13.	“Section 1” agent (relating data to users’ tasks and goals)	110
Figure 6-14.	“Section 2” agent (relating data to users’ tasks and goals)	110
Figure 6-15.	“File Menu” agent (selecting events)	111
Figure 6-16.	“File Toolbar” agent (selecting events)	112
Figure 6-17.	“File->Print” agent (selecting events)	112
Figure 6-18.	“All Menus” agent (selecting events)	113
Figure 6-19.	“All Toolbars” agent (selecting events)	113
Figure 6-20.	“All Commands” agent (selecting events)	113
Figure 6-21.	“Print Window” agent (selecting event contexts)	114
Figure 6-22.	“Section Events” agent (reducing event data)	115
Figure 6-23.	“Section Events” data (reducing event data)	115
Figure 6-24.	“Section Events” data visualization (reducing event data)	116

Figure 6-25.	“Section Transitions” agent (reducing event data)	116
Figure 6-26.	“Section Transitions” data (reducing event data)	117
Figure 6-27.	“Section Sequences” agent (reducing event data)	118
Figure 6-28.	“Section Sequences” data (reducing event data)	118
Figure 6-29.	“Submit Values” agent (reducing state data)	120
Figure 6-30.	“Submit Values” data (reducing state data)	121
Figure 6-31.	“Print Mode & Pages” agent (reducing state data)	121
Figure 6-32.	“Print Mode & Pages” data (reducing state data)	122
Figure 6-33.	“All Commands” agent (reducing state data)	122
Figure 6-34.	“All Commands by File Type” data (reducing state data)	123
Figure 6-35.	“Print Window” agent (user interface state)	124
Figure 6-36.	“Enter ZIP to complete City/State” agent (user interface state)	124
Figure 6-37.	“OK to Select Mode of Travel” agent (arbitrary state)	125
Figure 6-38.	“Not OK to Select Mode of Travel” agent (arbitrary state)	125
Figure 6-39.	“Mode of Travel Reselected” agent (arbitrary state)	126
Figure 6-40.	“Menu Count Increment” agent (arbitrary state)	126
Figure 6-41.	“Menu Count Reset” agent (arbitrary state)	127
Figure 6-42.	“Menu Count > 5” agent (arbitrary state)	127
Figure 6-43.	“File Type” agents (application state)	128
Figure 6-44.	“Mode of Travel Reselected” agent (user state)	129
Figure 6-45.	Data collection reference architecture	130
Figure 6-46.	Instrumentation-based data collection architecture	132
Figure 6-47.	Event monitoring-based data collection architecture	133
Figure 6-48.	Proposed data collection architecture	134
Figure 6-49.	Interrelationships between the problems and solutions	137

LIST OF TABLES

	Page
Table 2-1: Types of evaluation and reasons for evaluating	20
Table 2-2: Data collection techniques and usability indicators	21
Table 5-1: Basic Services API	87
Table 5-2: Event Specs	88
Table 5-3: Event Service API	88

ACKNOWLEDGMENTS

First and foremost, I want to thank my advisor and committee chair, Professor David Redmiles, for his patience, guidance, friendship, and openness. I am also indebted to the other members of my committee, Professors David Rosenblum and Jonathan Grudin, for their feedback and encouragement. Professor John King was a source of great wisdom in times of need, both professional and personal. Professor Dick Taylor was a source of great insight into how to run a large and successful research operation.

I couldn't have done it without the friendship of my colleagues and friends, Nenad Medvidovic, Jason Robbins, and Peyman Oreizy. Each contributed to my personal and professional well-being in ways perhaps even they will never know.

Thank you to the members of the Lockheed Martin C2 Integration Systems Team, Teri Payton, Lyn Uzzle, and Martin Hile, for their willingness to collaborate. And a very special thanks to the members of the Microsoft Product Planning, Program Management, and Usability teams, including David Caulton, Debbie Dubrow, Paul Kim, Reed Koch, Ashok Kuppusamy, Dixon Miller, Chris Pratley, Roberto Taboada, Jose Luis Montero Real, Erik Rucker, Andrew Silverman, Kent Sullivan, Gayna Williams. What a refreshing summer.

Thanks also to those who laid the groundwork for this research including Andreas Girgensohn, Allison Lee, David Redmiles, Frank Shipman, and Thea Turner.

Finally, I couldn't have done it without the money. This work has been supported by the National Science Foundation, grant number CCR-9624846, and the Defense Advanced Research Projects Agency, and Rome Laboratory, Air Force Materiel Command, USAF, under agreement number F30602-97-2-0021. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Defense Advanced Research Projects Agency, Rome Laboratory or the U.S. Government.

Permission to reproduce Figure 3-1, which originally appeared in *Human-Computer Interaction*, 1994, vol. 9, no. 3-4, p. 260, granted by P. Sanderson and Lawrence Erlbaum Associates, Inc., 1999.

CURRICULUM VITAE

1 Education

Doctor of Philosophy (1999, Cumulative GPA: 4.00 out of 4.00)

University of California, Irvine

Department of Information and Computer Science

Emphasis: Software

Advisor: David F. Redmiles

Dissertation: *Large-Scale Collection of Application Usage Data and User Feedback to Inform Interactive Software Development*

Master of Science (1996, Cumulative GPA: 4.00 out of 4.00)

University of California, Irvine

Department of Information and Computer Science

Emphasis: Software

Bachelor of Arts, Magna Cum Laude (1991, Cumulative GPA: 3.68 out of 4.00)

Tufts University

School of Arts, Sciences, and Technology

Major: Philosophy

2 Honors, Awards, Fellowships

1998	UCI Regents' Dissertation Fellowship
1995-96	GAANN Graduate Fellowship
1994-95	MICRO Graduate Fellowship
1991-Present	Phi Beta Kappa
1991	Philosophy Department Prize, Tufts University
1987-1991	Dean's Honor List (7 of 8 semesters), Tufts University

3 Professional Associations

Association for Computing Machinery (ACM)

ACM Special Interest Group on Software Engineering (SIGSOFT)

ACM Special Interest Group on Computer-Human Interaction (SIGCHI)

Institute of Electrical and Electronics Engineers (IEEE)

4 Professional Experience

6/98 - 8/98

Program Manager, Word 2000 Instrumented Version
Microsoft Corporation, One Microsoft Way, Redmond, WA.

6/95 - 9/95 & 12/95

Member of Technical Staff, Deep Space Network Automation Research
Jet Propulsion Laboratory, 4800 Oak Grove Drive, Pasadena CA.

3/93 - 9/94 & 12/94

Member of Technical Staff, Galileo Telemetry Subsystem
Jet Propulsion Laboratory, 4800 Oak Grove Drive, Pasadena CA.

5/90 - 9/90 & 5/91 - 3/93

Member of Technical Staff, EUCOM Decision Support System
Jet Propulsion Laboratory, 4800 Oak Grove Drive, Pasadena CA.

5 Publications

Refereed Journal Articles

David M. Hilbert and David F. Redmiles. "Extracting Usability Information from User Interface Events". ACM Computing Surveys (In Review).

David M. Hilbert and David F. Redmiles. "Collecting User Feedback and Usage Data on a Large Scale to Inform Interactive Software Development". ACM Transactions on Computer-Human Interaction (In Review).

Jason E. Robbins, David M. Hilbert, and David F. Redmiles. "Extending Design Environments to Software Architecture Design". Automated Software Engineering, 5, 1998.

Refereed Conference Publications

David M. Hilbert and David F. Redmiles. "Agents for Collecting Application Usage Data Over the Internet". In Proceedings of the Second International Conference on Autonomous Agents (Agents'98).

David M. Hilbert and David F. Redmiles. "An Approach to Large-Scale Collection of Application Usage Data Over the Internet". In Proceedings of the 20th International Conference on Software Engineering (ICSE'98).

Jason E. Robbins, David M. Hilbert, and David F. Redmiles. "Extending Design Environments to Software Architecture Design". In Proceedings of the 11th Annual Knowledge-Based Software Engineering Conference (KBSE'96). (Best of Conference Award).

Refereed Conference Demonstrations

David M. Hilbert, Jason E. Robbins, David F. Redmiles. "EDEM: Intelligent Agents for Collecting Usage Data and Increasing User Involvement in Development". Formal Demonstration at the 1998 Conference on Intelligent User Interfaces (IUI'98).

Jason E. Robbins, David M. Hilbert, David F. Redmiles. "Software Architecture Critics in Argo". Formal Demonstration at the 1998 Conference on Intelligent User Interfaces (IUI'98).

Jason E. Robbins, David M. Hilbert, David F. Redmiles. "Argo: A Tool for Evolving Software Architectures". Formal Demonstration at the 19th International Conference on Software Engineering (ICSE'97).

Refereed Workshop Publications

David M. Hilbert and David F. Redmiles. "Separating the Wheat from the Chaff in Internet-Mediated User Feedback". In Proceedings of the Workshop on Internet-based Groupware for User Participation in Product Development (CSCW'98).

Jason E. Robbins, David M. Hilbert, and David F. Redmiles. "Using Critics to Analyze Evolving Architectures". In Proceedings of the Second International Software Architecture Workshop (FSE'96).

Non-Refereed Publications

David M. Hilbert. "A Survey of Computer-Aided Techniques for Extracting Usability Information from User Interface Events". Technical Report UCI-ICS-98-13, Department of Information and Computer Science, University of California, Irvine, Mar. 1998.

David M. Hilbert and David F. Redmiles. "Why Let Perfectly Good Usability Data Go to Waste?". Boaster Paper at the Human-Computer Interaction Consortium Meeting (HCIC'98). Technical Report UCI-ICS-98-12, Department of Information and Computer Science, University of California, Irvine, Mar. 1998.

David M. Hilbert and David F. Redmiles. "Agents for Collecting Application Usage Data Over the Internet". Technical Report UCI-ICS-97-41, Department of Information and Computer Science, University of California, Irvine, Oct. 1997.

David M. Hilbert and David F. Redmiles. “An Approach to Large-Scale Collection of Application Usage Data Over the Internet”. Technical Report UCI-ICS-97-40, Department of Information and Computer Science, University of California, Irvine, Sep. 1997.

David M. Hilbert, Jason E. Robbins, and David F. Redmiles. “Supporting Ongoing User Involvement in Development via Expectation-Driven Event Monitoring”. Technical Report UCI-ICS-97-19, Department of Information and Computer Science, University of California, Irvine, May 1997.

ABSTRACT OF THE DISSERTATION

Large-Scale Collection of Application Usage Data and User Feedback
to Inform Interactive Software Development

by

David Michael Hilbert

Doctor of Philosophy in Information and Computer Science

University of California, Irvine, 1999

Professor David F. Redmiles, Chair

The two most commonly used techniques for evaluating the fit between application design and use — namely, usability testing and beta testing with user feedback — suffer from a number of limitations that restrict evaluation scale (in the case of usability tests) and data quality (in the case of beta tests). They also fail to provide developers with an adequate basis for: (1) assessing the impact of suspected problems and proposed solutions on users at-large, and (2) deciding where to focus scarce development and evaluation resources to maximize the benefit for users at-large.

This dissertation demonstrates technical and methodological solutions to enable usage- and usability-related information of much higher quality than currently available from beta tests to be collected on a much larger scale than currently possible in usability tests. Such data is complementary in that it can be used to address the impact assessment

and effort allocation problems in addition to evaluating and improving the fit between application design and use.

This research has been subjected to a number of evaluative activities including: (1) the development of two independent research prototypes at the University of Colorado and the University of California, (2) the incorporation of one prototype by independent third party developers as part of an integrated demonstration scenario performed by Lockheed-Martin Corporation, and (3) observation and participation in two industrial development projects, conducted at NYNEX and Microsoft Corporations, in which developers sought to improve the application development process based on usage data and user feedback.

The approach described herein involves a development platform for creating software agents that are deployed over the Internet to observe application use and report usage data and user feedback to developers to help improve the fit between design and use. The data can be used to illuminate how applications are used, to uncover mismatches in actual versus expected use, and to increase user involvement in the evolution of interactive systems. This research is aimed at helping developers make more informed design, impact assessment, and effort allocation decisions, ultimately leading to more cost-effective development of software that is better suited to user needs.