Convergent Usability Evaluation: A Case Study from the EIRS Project

Jeff Johnson
UI Wizards, Inc.
231 Moscow St.
San Francisco, CA 94112
jjohnson@uiwizards.com

Catherine Marshall
CollabTech, Inc.
395 Del Monte Center #200
Monterey, CA 93940
cmarshall@collabtech.com

ABSTRACT
Two non-profit organizations developed a Web application to help monitor U.S. elections: the Election Incident Reporting System (EIRS). The mostly-volunteer team had only four months to develop a workable system. The aggressive schedule, limited budget, and distributed team-structure challenged us to find creative ways to evaluate and improve EIRS' usability. We used an approach that combined expert UI review with opportunistic exploitation of venues for gathering data on EIRS' usability. This approach, which we call convergent usability evaluation, had, in the non-profit environment, advantages over the more formal methods typically used for commercial projects. In this paper we describe the usability evaluation methods we used for the EIRS project and discuss how they converged to provide a more complete picture than we would have obtained by conventional methods.

Author Keywords:
Rapid development, non-profit projects, open-source projects, social issues, user-centered design, usability evaluation methods.

ACM Classification Keywords
H.5.2 Information interfaces and presentation: User Interfaces.

BACKGROUND

In June 2004, two non-profit organizations, the Verified Voting Foundation and Computer Professionals for Social Responsibility (CPSR), received a grant from the Quixote Foundation to develop a web-based application to help monitor U.S. elections and capture data concerning voting problems. The Election Incident Reporting System (EIRS) [1] was to be used in the November 2004 U.S. election.

Intended Users

Intended users of EIRS fall into three classes: call-center volunteers who enter data about voting incidents into EIRS, call-center managers who manage user accounts, and analysts who use the election incident data [2] to detect and remedy voting problems. Most users entering election incident reports were volunteers who either: a) took incident reports by phone from poll observers or members of the voting public, or b) entered data from paper forms filled out by poll observers or call-center volunteers. Also, members of the general public could file voting incident reports directly at an auxiliary website (VoteProblem.org) although such reports were stored and reviewed separately.

CHI 2005, April 2–7, 2005, Portland, Oregon, USA.
ACM 1-59593-002-7/05/0004.

Components of EIRS UI

User-visible aspects of EIRS fall into four categories:

- *Data-entry forms*: EIRS provides many forms: election incident reporting, equipment test reporting, election official questionnaires, volunteer registration, and others.
- *Web pages*: EIRS includes dozens of web pages, containing everything from navigation controls, instructions, and background information to data-input forms and data-output displays.
- *Map displays*: The primary way EIRS currently organizes election-incident data is geographically – by state, county, and voting precinct. Accordingly, maps are used heavily as a display medium.
- *Incident detail displays*: EIRS users can "drill down" through states, counties, and precincts to the details of particular election incidents.

Distributed, Fluctuating Development

VerifiedVoting and CPSR quickly built a team consisting mainly of volunteers to design, implement, test, and document the application. Team members were located all over the U.S. – there were even members in Europe and Australia – so they worked together mainly by Internet and phone. During the development period, designers and developers continued to join the project, and some left it.

Despite the fact that the team was physically distributed and fluctuating, it designed and developed EIRS on "Internet time" with a hard deadline: election day, Nov. 2.

EIRS Worked

To date, EIRS has been a success. A preliminary version of it was ready in time to help document voting problems in the Florida Primary election on August 31, only two months after development had begun.

In the general election, volunteers organized by the non-partisan Election Protection Coalition, working at call-centers all over the U.S., handled over 175,000 calls, about half of which were on Nov. 2. Call-center volunteers used EIRS to log over 40,000 election incident reports, most of which were logged on election day.

In addition to collecting data that will be useful for research and policymaking, EIRS helped in real time during the election. It allowed election monitors to detect and remedy many serious problems at polling places nationwide. Call-center operators also helped tens of thousands of voters find polling places and resolve other minor voting obstacles [3].

THE CHALLENGE: ENSURING USABILITY

Because of the aggressive development schedule, EIRS was designed and assembled quickly, making heavy use of open-source software and ideas that team members brought into the project from their experience.

User interface and usability experts joined the project only after design and development had already begun. This ruled out a conventional front-loaded user-centered process. The early EIRS team did start with requirements analysis and use-cases, but developed minimal conceptual design or UI design specifications prior to starting implementation.

The UI and usability team usually found itself in the position of playing "catch-up": trying to ensure the usability of an application for which design was being done mainly by programmers, and on which implementation was proceeding rapidly. Therefore, usability evaluation played a more prominent role in ensuring EIRS' usability than did conventional up-front interaction design methods.

However, even conventional *evaluation* methods [4] were impractical. This was due to: a) the frantic pace of the project, b) the low budget, and c) the fact that the only call centers set up before election day were 3000 miles from the usability team. These factors ruled out even one round of formal usability testing prior to EIRS' deployment in the election. Normally, that would be regarded by HCI professionals as a recipe for disaster, especially for such an important, high-profile application. However, the usability team understood that the methods we typically use in developing commercial systems were infeasible. We adapted. We took advantage of any opportunities to assess and improve EIRS' usability. We minimized our impact on the project by piggybacking on already-planned activities.

EVALUATION METHODS USED

The UI team exploited a variety of opportunities to evaluate and improve EIRS' usability. These included: expert review, trial deployment, remote training observation, participation in Q/A testing, election-day field observations, and post-election interviews.

Expert Reviews

During development, the primary method of evaluating and improving the usability of EIRS was expert review EIRS management recognized the need for usability reviews even before any usability experts joined the project. Because the voting incident report form was so important, project leaders solicited and received feedback on a very early version of it from outside usability experts in July.

Once usability experts had joined the project, developers posted messages soliciting reviews of a specific aspect or part of EIRS they had implemented. Usually the design to be reviewed was in the current "test" version that team-members could access, but sometimes developers posted preliminary implementations or prototypes and asked for feedback before putting them into EIRS.

Usability team members generally provided reviews whenever asked, and sometimes offered feedback even when not asked. Other project members also often pitched in to provide review feedback as time permitted.

Early "Trial" Deployment in Primary Election

The state of Florida conducted its primary election on August 31. Although EIRS was far from complete by then, it was functional. Enough was implemented to allow it to be used in that state election to collect voting incidents.

Users and Election Protection Coalition leaders provided feedback through a variety of channels. An important complaint received from the Florida primary was:

- Volunteers cannot edit incident reports after submitting them. This is a problem because people sometimes realize that they entered incorrect data, or they later learn something new about the case.

Remote Training Session Observation

Throughout development, the EIRS team discussed the desirability of conducting formal usability tests: observing representative users performing prescribed tasks in a fairly controlled environment. This was spurred by a concern that putting such a complex, important, and visible application into use without first testing its usability was unwise – even foolhardy. However, as the election neared, it became clear that formal usability tests simply were not going to happen: the team did not have resources to design a test, recruit participants, find a site, create a realistic test-environment, conduct test sessions, and analyze the results. The election was too near and things were moving too quickly.

In mid-October, as this realization sank in, an alternative presented itself: groups of volunteers slated to work as Election Protection call-center volunteers and first-level managers were being trained on how to use EIRS. It occurred to us that observing training sessions might provide opportunities to see users' reactions to EIRS. We arranged to sit in on some training sessions.

We soon learned that the training sessions were not normal instruction in physical classrooms because the trainees were scattered around the U.S. Instead, the training was conducted by telephone conferencing. Presentation slides describing EIRS and illustrating its pages were emailed to trainees, who

joined a conference call at a prescribed time. An instructor led the group through the slides, explaining how to use EIRS and answering questions.

We initially wondered whether listening in on a conference call slide presentation would be an effective way to discover usability problems in EIRS. In fact, it turned out to be quite valuable, for three reasons:

1. The presentation was a fairly systematic and thorough walk-through of the functionality of EIRS. This allowed us to notice design flaws and potential usability problems that we might have missed in our self-guided reviews.
2. As instructors explained how to use EIRS, it was clear that they felt there were "rough spots" in the UI that had to be explained carefully to avoid errors on election day.
3. Questions and comments from trainees provided insight into their understanding – or misunderstanding – of how EIRS worked, and the context in which it would be used.

Between us, we "attended" a total of six training sessions. Examples of observations reported back to the team are:

- At the bottom of the incident summary page, there is an "Edit Task" button that isn't for editing the report. A trainee asked what it does. The instructor didn't know.
- Users are urged to "be sure to print" incident forms before submitting them. A "Print" link is provided at the bottom of the form for this purpose. Some trainees argued that if forms must be printed, EIRS should just do it, rather than relying on users to remember. Trainees asked what happens if the printer is busy or out of paper when a user clicks "Print". The instructor didn't know. Trainees wanted to be able to "print" completed forms to files.
- If a volunteer tries to edit a report s/he didn't create, EIRS doesn't explicitly deny permission; instead it asks the user to login (even though s/he is already logged in), because it wants the user to login as the user who entered the report. This relies on volunteers to understand why they are being asked to login again.

Problems discovered by observing training sessions were filed as bug reports so they could be corrected or further emphasized in training before election day.

Participation in Q/A Testing

As election day loomed, another evaluation opportunity arose: Q/A testing. We monitored the Q/A email list to learn what aspects of EIRS needed testing. We exercised EIRS, often following test-scripts devised by the Q/A team. We reported usability problems as well as bugs. One of us also helped create test scripts and noted issues that confused other Q/A testers.

An important usability problem this exposed was that the relationship between user accounts and access privileges didn't match users' expectations: EIRS often wouldn't let users – including testers – do what they needed to do.

"In Vitro" Field Observations& Interviews

Because EIRS will be used in elections beyond 2004, the usability team and EIRS management thought it would be valuable for the usability team to document how well (or poorly) EIRS worked for its intended users and tasks. We therefore spent election day at two call-centers, observing, interviewing, and videotaping volunteers using EIRS.

Two call-centers in San Francisco allowed us to observe their operation. These were large centers, each with over 50 operator workstations. Each handled voting-problem calls from a subset of western states. Both centers expected to be filmed on election day by news organizations (a correct assumption), and so had volunteers sign media-releases. This made it possible for us to videotape without having to obtain permission from each volunteer.

Although the call-center volunteers were very busy on election day handling calls from voters and poll observers, we were able to collect a lot of feedback on the design of specific EIRS features, especially the incident-reporting form. Our election-day observations included:

- EIRS' designers assumed each user would be registered with their own login account. However, management at both SF call centers considered that too much bother. They created generic user-accounts and had everyone use them. EIRS' developers feared that the system wouldn't work properly with multiple users simultaneously using the same account. VerifiedVoting staff at one SF call-center quickly assigned each station a unique login, but that wasn't practical nationwide. Fortunately, careful monitoring soon revealed that EIRS works properly with multiple users logged into the same account.
- With hundreds of volunteers entering election incidents all across the U.S. at more-or-less the same time, EIRS bogged down considerably. At times, it was intolerably slow, causing call-center volunteers to switch to taking incident reports on paper forms for later entry into EIRS.
- As some expert reviewers predicted, volunteers found it difficult to fill out the online incident form as callers talked, because they couldn't force callers to follow the order of the form when describing voting problems. Some users scrolled up and down the form as the caller talked. Others first wrote out a summary of the situation, either on paper or in the incident Description field, and later went back and filled out the form fields.

In addition to learning much about EIRS' usability, observing the call centers allowed us to see and understand the context in which EIRS was used. EIRS was only part of the process and only one of several tools used by call-center volunteers. For example, many used the website MyPollingPlace.com to help callers find where to vote.

The call centers also employed non-computer artifacts to support volunteers in helping callers. At both centers, late-breaking information was posted on walls. Printed incident reports were sorted into boxes according to their urgency.

An important insight that came out of the election-day field observations was that many calls were from voters trying to find their polling place. EIRS was designed under the assumption that most calls would be reports of voting irregularities. Although a significant number of voting irregularities were reported [3], it is useful for future planning to understand that the call-centers functioned primarily as voter information hot-lines.

Post-Election User-Interviews

After the election, the usability team suggested interviewing a sample of EIRS users to capture their comments, criticisms, and suggestions while their memories were still fresh. EIRS management agreed, and asked two organizations that had recruited and organized many of the call-center volunteers (People for the American Way Foundation and the Lawyers' Committee for Civil Rights under Law) to nominate users for us to interview.

None of the nominated users were near San Francisco (where the usability team is located), so interviews were conducted by telephone and email. The interviewers were the same two UI team members who had conducted the call-center observations and interviews on election day.

The post-election telephone interviews provided data similar to that collected during the election-day field observations. We collected more feedback on the design of the incident reporting form. We learned that the revolt against using individual user logins was quite widespread. Overall, these interviews helped us understand the variety of ways in which call-centers operate.

DISCUSSION: CONVERGENT EVALUATION

Initially, the opportunistic, multi-pronged, low-impact approach we used to assess and improve EIRS' usability seemed like making the best of an undesirable situation. In retrospect, we feel that the potpourri of evaluation methods we used yielded results that compare favorably to relying on more formal methods.

It certainly was less expensive. We worked as volunteers, but even had we been paid for our time, one formal usability test would have consumed more hours – and cost more – than did all of the methods we used combined.

Beyond cost considerations, using multiple evaluation methods allowed us to see potential usability problems from different perspectives: our own, that of trainers trying to teach EIRS to others, that of trainees trying to understand EIRS, that of users in the midst of fielding problem-calls, and that of seasoned users reflecting on their experience. This provided important insight into which usability problems were more serious, and which problems were best remedied by redesign vs. training. Examples of how the evaluation methods complement each other include:

- A browser-compatibility problem seen during an expert review was dismissed as low priority until it also was seen in a training session and Q/A testing.
- Reviewers of the incident forms questioned the need for certain fields to be "required". We later witnessed the trouble such fields can cause for users in the hectic call-center environment.
- Some call-center managers complained during training that giving unique logins to each user was problematic, but their complaint was poorly understood, so it was discounted. A subsequent review of EIRS' account-creation functions found problems, but none seemed serious enough to prevent managers from registering every user. The gravity of the problem was only recognized on election day when call-center managers refused to register volunteers, instead using generic logins. Fortunately, EIRS did not crash or corrupt data.
- Participating in Q/A testing primed us to spot, in our field observations, important usability problems that were hard for users to describe or that appeared only occasionally.

It is a fair question whether our findings were as valuable as those that more conventional methods might have produced. Some argue that discount methods are less effective and must be fortified to improve their value [5]. Perhaps, but we feel that combining methods as we did provided significant mutual fortification. Certainly many important usability problems in EIRS were exposed and corrected, and many more were identified for future revisions. It is also worth noting that a usability test would not have caught some of the problems we found, such as the fact that most call-center managers prefer generic volunteer accounts over giving each volunteer an account.

In any case, discount methods were all the project could afford. Therefore, comparing the efficacy of discount vs. conventional methods is moot in this case. The real alternative to what we did was to do nothing, and simply hope EIRS would be usable on election day. Without question, our methods were better than nothing.

The low-cost, low-impact usability evaluation methods used in the EIRS project shed light on each other. They did not simply *add* like using more experts in a heuristic evaluation. Because of their different perspectives, they converged, yielding a total evaluation that is greater than the sum of its parts. We therefore refer to using these methods in concert as *convergent usability evaluation*.

REFERENCES

1. Purpose and History of EIRS: verifiedvoting.org/eirs.
2. EIRS data: voteprotect.org (click *Research/Maps*)
3. Reports of EIRS usage and findings in Nov'04 election: verifiedvoting.org/article.php?id=5302, 5321, 5339.
4. Jeffries, R., Miller, J.R., Wharton, C., and Uyeda, K. "User Interface Evaluation in the Real World: A Comparison of Four Techniques", *Proceedings of CHI'91*, ACM, New York, pages 119-124.
5. Cockton, G. & Woolrych, A. "Sale Must End: Should Discount Methods be Cleared off HCI's Shelves?", *Interactions*, Sept./Oct. 2002, pages 13-1.