CALTECH/MIT
VOTING TECHNOLOGY PROJECT

ORIENTING GRAPHICAL USER INTERFACES REDUCES ERRORS: THE LOW ERROR VOTING INTERFACE

Ted Selker
MIT

Matt Hockenberry
MIT

John Goler
MIT

Shawn Sullivan
MIT

Key words: *ballot interface, residual vote, experiment, time*

Orienting Graphical User Interfaces Reduce Errors:
The Low Error Voting Interface

Ted selker[1], Matt Hockenberry[1], John Goler[2]
Shawn Sullivan[1]

[1] MIT Media Laboratory, 20 Ames Street, Cambridge Massachuesetts
{selker, hock, stsully}@media.mit.edu
[2] Berkeley Computer Science Department, Berkeley California
jgoler@Springer.de

Abstract. This paper demonstrates opportunities for reducing errors with orienting graphical interface for voting. We have built many interfaces to explore opportunities for keeping voters aware of the selections they have made and are making. Tests of our best prototypes show that missed races and incorrect selection errors are greatly reduced with orienting graphics. The interface reduces errors significantly while extending the time required to vote.

1 Introduction

Between 1.5 and 2.0 million votes were lost due to ballot problems in 2000. [1]

Difficulties being able to vote for any selection other than those put on a preprinted ballot was once commonplace. Coersion and security issues brought America to begin adopting lever machines in to deal with these issues in the late 1800s. These lever machines allowed a voter to see all selections simultaneously. To eliminate certain kinds of vote stuffing, the machines had a permanent record of the results on mechanical odometers. To eliminate coercion in the polling place they used curtains. Such voting machines show all voting options available simultaneously and spatially. They also give the voter physical and visual feedback when any selection is made. The also disallow mistakes such as voting for two presidents. Probably for these reasons they produced a strikingly low 'residual' with 1.5% not voting in the presidential race even decades after their constructions. Despite of all of their virtues, full-faced lever machines have a history of maintenance problems, malicious tampering and difficulties for people with disabilities. [2]

Various alternatives are used to avoid these problems. A small number of voters in the US still vote on hand counted ballots. Although these register few missing votes for president, they are widely thought to be the least accurately counted ballots available. Pre-scored punch cards that separate the selection from the selection descriptions have been available since 1964 but showed the highest missing presidential vote of any voting system in 2000. These punch cards ballots have alignment and orientation

problems when inserted into the machines, mechanical problems making selections and various other problems. Some punch systems print the selections on the ballot to be punched, these have fewer problems. A similar system, the optical scan ballots, simply requires a ballot with marks on it to be scanned after being marked. Still, the only voting system besides the lever machine to give instant feedback to prevent a voter from voting for too many candidates is the direct record electronic voting machines; these are voting machines that use computers to both present selections and record them. [3]

While these machines raise eyebrows for their lack of transparency, they have become associated with some of the best examples of improvements for accurate secure voting.

1.1 Approaches to Ballots

The various DRE ballots available have some differences in their ballot designs. Today's Sequoia ballots eliminate buttons for alternative selections when one is made. Deibold ballots and Sequoia ballots both show multiple races on some screens and not on others. Full-faced ballots on prototype LCD machines such as the Election systems and solutions LS might show all candidates simultaneously. The question is which of these systems are helping people vote accurately? [4] [5]

Believing that people have trouble finishing their ballots because they are not sure what they have already voted for, we began experimenting with showing all races simultaneously on a DRE machine. Our first idea [Figure 1] was to present graphical interfaces as levers so that the one lever would stand out from the rest. The idea may be a good one but when embedding levers in boxes to distinguish choices the salience of the representation became diminished.

Fig. 1. Early design with visual lever.

The next idea was to use the lever out of alignment feature that lever machines have to show that one candidate is selected. This is shown in Figure 2 with the second green candidate offset to the right. This mockup also explored use a series of pages, which overlap color and texture in order to remind a voter that gray races hadn't yet been voted and that green races have recorded selections.
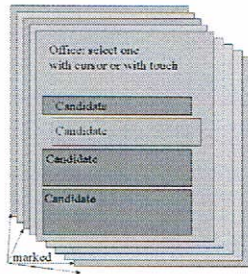
Fig. 2. Introduction of pages metaphor in early samples.

Reflecting on these working prototypes and mockups we came up with some important principles; the voter should always know:

- what races need voting on.
- what was voted.
- if a selection was made.

The Pages made it easy to see that you had voted and how many races you may not have made selections on, but it did not solve the other two goals.

Our next prototype became much more sophisticated at this, showing offsets and color-coding for selected candidates, offsets and color-coding on tabs for selections. This became the basis of a few years of improved prototypes. Further examples of this can be found in our prototype archive [6]
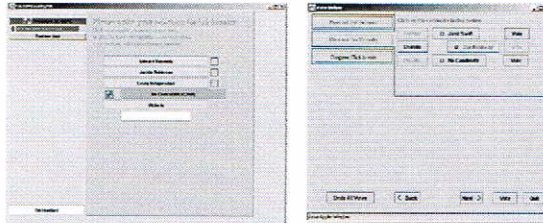

Fig. 3 & 4. A more sophisticated prototypes expands on the tab concept.

2 The Voting Interface Initiative

This interactive website presents layout and feedback options we considered for addressing our design goals. Issues of how to compare vote selections to alternatives, balance the likelihood of candidates getting selected, and ways of showing large

numbers of selections and races were all highlighted in interactive voting user interface simulations.

In prototype 1.3 for example many of our best ideas are explored: Fonts are larger for each race to attract the eye farther down to make selections on all races. The tab is merged with the selection page when a race is activity being voted on. The selections are highlighted partially in the tabs when the race is partially voted.
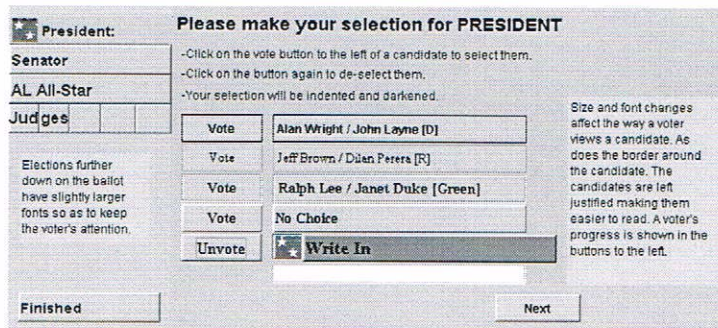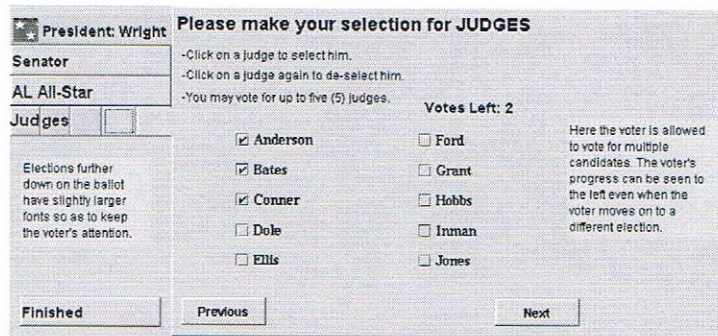

Fig. 5 & 6. Prototype 1.3 exemplifies some of the key ideas being explored.

Prototype 2.2 explores instant runoff and multiple selection approaches. It shows how font choice and layout can effect selection. It also demonstrates a variety of approaches for presenting instructions, some of which appear too easily to be selections themselves. When selecting numbers for colors the selection and the marker might get confused; are the numbers to order the colors or are the colors to name the numbers?
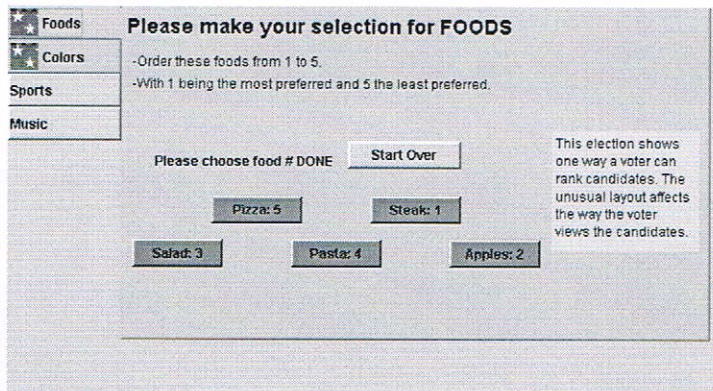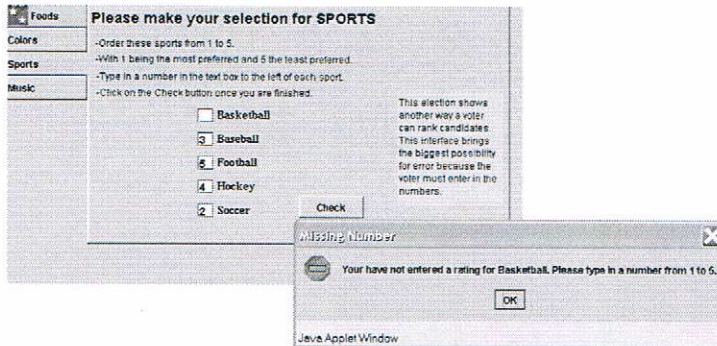
Fig. 7 & 8. Prototype 2.2 demonstrates exploration into other voting methods such as runoffs.

Prototype 3.1 takes on even harder problems of allowing voters to evaluate which selections are to matter how much. We encourage the reader to view these interfaces built and deployed with the help of Bruce Dibello.
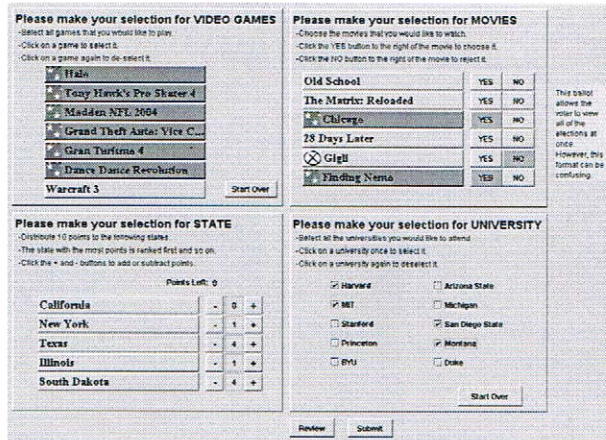
Fig. 9. Prototype 3.1 is flexible but potentially confusing.

2.1 The Low Error Voting Interface (LEVI)

To evaluate the simple ideas of overview and focus more rigorously, Shawn Sullivan wrote a new simulator in Java that could create various voting machines. We created simulations of currently available commercial voting machines interfaces for comparison with new ideas. The following figure shows it presenting a version of the Low Error Voting Interfaces that demonstrates how to present many races in a limited sized ballot. Even at the resolution in this paper, one can tell that a selection has been made on the race on which the voter is working. We can see that four races have had selections made on them. The fisheye [7] view allows us to see more races exist than we could easily view simultaneously.
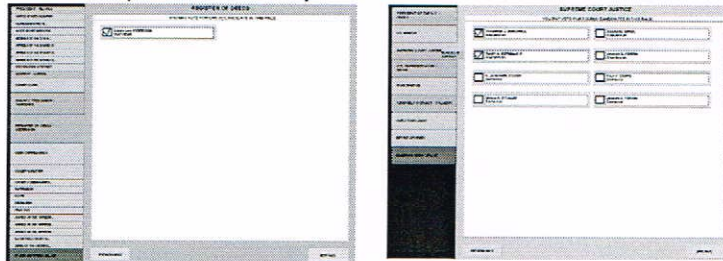

Fig. 10 & 11. Prototype versions of the LEVI. Green indicates a completed race.

Figure 11 shows the LEVI voting interface presenting the Erie county 2000 ballot as it was used in our experiment. Erie County was chosen because it has a fair number of races and we were able to get the actual voting materials for it. Review screens from the Sequoia like system, Deibold like system and LEVI are presented below for comparison.

Fig. 12, 13, 14. Diebold, Sequoia, and LEVI review screens.

LEVI was designed to be a voting interface which was capable of easily providing all three voter information criteria. The page and tab system, used in previous prototypes as mentioned before, was used in LEVI to show all races on the ballot simultaneously on screen, using the fisheye when necessary to allow many races to be shown. Thus, the voter could always see the races that were present in the election, our first information criterion.

By changing the color of the tab when votes are cast in particular races, again drawing from prototypes, LEVI is able to easily display to the voter which elections the voter has voted in. The color change can additionally be supplemented by texture change, useful for the colorblind. The color/texture change on the tabs therefore satisfies our second information criterion, to always show the voter which races they have already voted in and which races they have not.

Finally, by writing the name of the selection or selections the voter has made on the tab, LEVI is able not only tell voters which races they have voted in, but also what selections they have made in that race, satisfying the third information criterion of always allowing the voter to see what selections they have made, if any, in all races. With these three main features, we believed we had a prototype representing a significant improvement over existing interfaces, and prepared to test this belief.

3   Evaluation of the LEVI Interface

While our initial interface designs were conducted in house once we had achieved a flexible prototype we went about designing a study to evaluate its effectiveness as they compared with commercial voting interfaces. The study consisted of two parts, a final qualitative component and then a traditional quantitative evaluation.

3.1 Qualitative Study Design

The qualitative component was designed with two goals: to make any final revisions to the interface that addressed serious breaks for the user's interaction and to gain a more detailed insight in to the user's mental model of the voting experience than we would in the quantitative analysis.

We modeled our qualitative evaluation after the traditional Think Aloud Usability Study in order to get a sense of what key problem areas in the interface of the LEVI might be. A Think Aloud protocol requires the user to describe their internal monologue as they move through the interface. It is important to note that we don't want user's to justify or explain what they are doing, but to merely 'speak their mind' in an attempt to get as accurate a picture of their internal monologue as we can. While from a design standpoint we were interested in identifying key problem areas, the idea of getting at the user's 'internal monologue' appealed to us as researchers. The goal is to understand how the mental model the user develops when using a system like LEVI differs from those developed during the use of commercial voting systems.

This qualitative evaluation is based on seven subjects, a fairly good number for this kind of study. Usability experts generally believe that even five subjects will manage to find almost 95% of the major usability breakdowns within an interface [8] so with seven subjects we can be fairly sure that we are uncovering most of the breakdowns within the interface, as well as to give us some latitude if any of the evaluations encountered difficulties.

The subjects were given four tasks to complete in a limited evaluation ballot. The ballot had only four races and we had subjects start on race two (in order to force them to return and correct race one during the review process). The tasks are listed in figure one. The subjects were asked to complete these tasks for each of the three interfaces: the LEVI, the Sequoia commercial interface (referred to from now on as the S-commercial interface) and the Diebold commercial interface(referred to from now on as the D-commercial interface). We were also interested in observing if there were any significant changes in user interaction between the relatively similar commercial interfaces because this would allow us to conduct a simpler quantitative study later on.

We recorded the subjects audio and screen so that we could see what they were thinking when they made various actions with the interface. In some cases we asked subjects additional questions or clarifications after the study had concluded. We made every attempt to ensure that subjects remembered to "think aloud" throughout the course of the study.

3.1 Qualitative Results

As is typical with these sorts of evaluations we confirmed some pre-existing notions as well as being surprised by interesting user interactions. One of the major conclu-

sions we reached from this is that the overall form of the ballot is much more important to directing user behavior than slight details which are meant to reinforce behavior. For example, small labels throughout the interface are generally ignored unless the user is completely lost. In the cases that the user's needed to resort to reading these labels the form of the ballot didn't seem to influence the order in which they read the additional instructions.

Task List for Think Aloud Usability Study
Task One: Begin in race two. Vote completely for this race. You may choose candidates in any manner you wish. When you have completely voted this race move on to the next race.
Task Two: Begin by voting for a candidate OTHER THAN Al Smith. After you have voted for this candidate, change your vote so that you are now voting for Al Smith. When you are done you may move on to the next race.
Task Three: Vote for two candidates of your own choose AND vote for Bob Marle. After you have voted move on to the review.
Task Four: Review your ballot. Make sure that ballot is completely voted in all of the races. If this is not the case, use your interface to make sure that each race is completely voted before you complete your ballot.

Fig. 15. Task list for qualitative evaluation.

Users almost universally had difficult trying to change a vote. We had made it so that in order to change one's vote, one needed to click on a new candidate and then click on the old candidate to 'unvote' him or her. While some of the older iterations make this action more explicit (with the use of a 'vote'/'unvote' button) it is no longer clear. To further drive home this point, most users expected the ballot to behave like radio buttons, automatically deselecting a candidate to choose the new one. We had not enabled this kind of functionality because it didn't allow a consistent interaction – what do we do with multi-candidate races? It was very clear, however, that users are willing to spend more effort to think about voting for multiple people and expect this intuitive radio button style interaction on simpler races. This was particularly true of user's who had computer backgrounds.

User's (even for monetary compensation) don't spend as much time as one might imagine checking their votes. While we even included explicit instructions to check their votes, many users did not do this on the commercial systems. The reasoning for this is that simply didn't notice or didn't want to bother gathering information that is "off screen." This strengthens the LEVI systems use of a constant review scheme that is always available and readable. The commercial systems review screen presents information off of the screen (showing, in the case of the D-commerical system only a handful of races).

The last conclusion was that user's would generally prefer more iconographic information, such as party pictures next to candidates. This is a welcome suggestion, but made difficult do to existing election laws, which often don't permit this kind of behavior.

The qualitative study seemed to enforce the LEVI system as one, which user's preferred (overwhelmingly) and which seemed to have more comfortable interactions. The quantitative evaluation would allow us to confirm this conclusion.

3.3 Quantitative Design

We designed our quantitative study to explore the number of errors and the time required to vote for a small number of randomly selected users. We presented the users with two distinct interfaces. Our LEVI system as well as the S-commercial system. We limited the evaluation to these two systems because our qualitative evaluation had determined little difference between the S and D-commerical systems, and that S was generally considered to be the better design of the two (from a user interface standpoint).

Each subject was presented with a series of four elections. After completing some initial surveys they were directed to the first election. Each subject was coded and we utilized a latin-square design to help eliminate ordering effects for the two interfaces. After the first election, a second election followed using the same interface. After both elections were complete the subject was given some survey material as well as some unrelated tasks in order to further eliminate the possibility of some learning effect. For the second stage of the experiment the subject was asked to complete the same two elections again, this time utilizing the interface they had not previously used. Finally, they completed some additional survey material and concluded the session.

Our subject pool was mostly limited to 18-25 year olds who had some computer knowledge (and we found that other subjects did not differ significantly from this base in terms of performance). [9]

We measured two main quantitative variables: the user's time to complete a section and the number and kind of errors made in that section. We had previously identified three kinds of voting errors that subjects were likely to make: undervotes, overvotes, and accuracy errors. Undervotes occur when a voter unintentionally fails to vote for as many candidates as he/she is entitled. Overvotes occur when a voter attempts to vote for two many candidates (and in the case of the electronic systems we tested, do not occur because the systems forbid them). Accuracy errors occur when the voter intends to vote for someone and incorrectly marks a different candidate. Obviously these errors are hard to identify in practice. Our solution to this was to simulate a voting agenda for our subjects. In practice most voters have some voting agenda, whether it be to vote a party ticket or a couple of key people they want to vote for. We prepared palm cards that listed a partial voting agenda for the subjects. In some cases the voting agenda instructed the subject to vote simply for candidate(s) of their own choosing (a wildcard, if you will). This was done in order to still allow the possibility of accidental undervotes. Accuracy errors were identified as selections not on the subjects voting agenda and not located in a wildcard choice. Undervote errors were

identified as a positive difference between the number of the maximum allowed candidates and the voter's chosen candidates.

3.4 Quantitative Results

Of the twenty-three subjects we considered initially, we found that only 16, or 70% had made errors while voting. Of these 70% the number of errors made when using the LEVI system was significantly lower (T-Value = -1.87 P-Value = 0.037) when considering a p-value < .05. In other words, we are 95% confident that user's make fewer errors on the LEVI system if they make any errors at all. The average number of errors was about 1.8 for LEVI and 3.1 for the S-commerical system. We had initially seen a stronger difference, but the inclusion of some very error-prone subjects helped lesson this difference. Consider figure x for a boxplot of the total errors.

We also noticed that the time was consistently longer for subjects during their use of the LEVI system. LEVI requires, on average 237.7 seconds, or almost four minutes. S-commercial system requires, on average 194.9 seconds, or three minutes, fifteen seconds. So on average, Sequoia is fifteen seconds faster, (although averages can be misleading, subjects times show very large differences)

This is not particularly surprising as there were similar results in earlier trials. The LEVI system presents only one race at a time, and offers additional contextual information about the user's state in the process. The deliberate segmentation of the process slows the user down and forces them to be more deliberate in their actions. System such as the S-commerical system allow faster times precisely because they are willing to present more races and allow more mistakes. Although the LEVI system does present a lot of information, it is all grounded on the user's stage in the process. Systems like the S-commercial system may encounter these larger error rates because they attempting to present multiple parts of the process at one time, while offering little grounding to the user.

There were also some interesting trends with regard to the LEVI systems role in improving subsequent user performance. People who use LEVI first are 21% less likely to make errors on the S-commercial system and 22% less likely to make errors on LEVI. This seems to indicate that LEVI is better at placing the user in a good mental model for voting than the S-commercial system is.
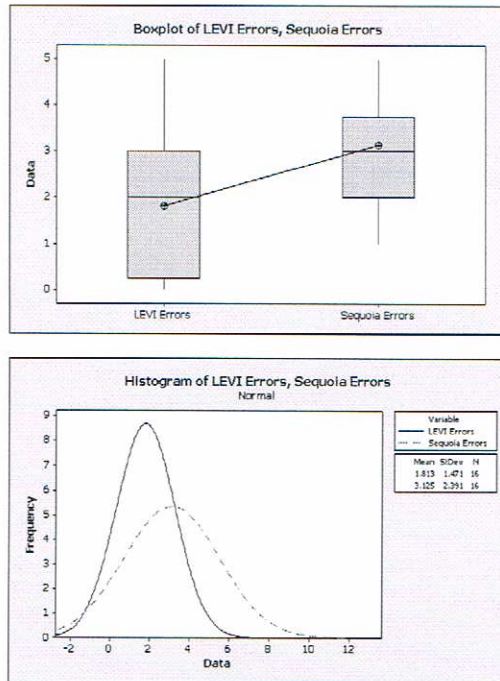
Fig. 16 & 17. Boxplot & histogram of LEVI and Sequoia error totals.

Overwhelmingly, there are some very real benefits to the LEVI voting interface. While it is not as fast as the commercial system, it improves user performance and seems to place user's in a better mental model to enable more successful voting. User's almost unanimously prefer LEVI saying that it is simply "more intuitive" and that they "know what they are doing" while using it. The study confirms some of the initial hypothesis about how an interface can enable user's to have a more secure place in the voting process.

5 Future Work and Conclusions

While we have made some advances in designing a more flexible and less error prone ballot, there are continual improvements that can be made. As the ballot is refined it has begun to take on a more friendly appearance while focusing on the strengths that

have been demonstrated through evaluation. There is also ongoing work to explore these same concepts in other modalities, extending the work to other kinds of voting systems such as optical scan machines and auditory ballots for the visually impaired.
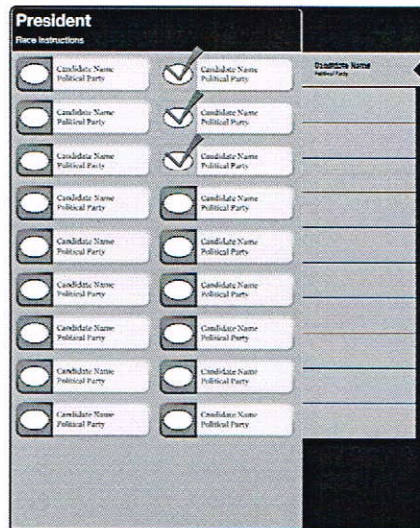

Fig. 18. Latest LEVI prototype version 5.0

These studies show how to make large improvements can be made to the accuracy of electronic voting through orienting user interfaces. Such interface improvements should decrease the errors in filling out ballots significantly. For the country this could save several million votes that are currently incorrectly marked. We don't have reasons to believe that mistakes of adjacency mistakes and race omissions appriori favor one candidate over another. On the other hand many ballot layouts are made to favor incumbents, these improvements should remove some of the down ballot omissions and mistakes and might remove some of the ballot bias towards incumbency.

Voting is a simple transcription of selections often written on a sample ballot onto voting equipment. The simplicity of the interface and actions and the stress of the value of them make voting an excellent laboratory for user interface research. Treating user interface as a process instead of action by action allows ancillary information to contribute to eliminating errors. The simple addition of tabs that demonstrate the value of selections to the whole solution is probably the biggest improvement that LEVI makes. While orient and focus is a mantra that we carry, never have we collected better data to show that overview and user placement can systematically reduce user interface errors.

We hope this paper encourages user interface researchers to add process orienting imagery to the necessary goals required to help people not make errors in user inter faces.

References

1. Alvarez, M., R., Ansolebehere, S., Antonsson E., Bruck, J., Graves, S., Palfrey, T., Rivest, R., Selker, T., Slocum, A., Stewart, C. III, etal What Voting Is and What it Could Be., Caltech/MIT Voting Project. 2001
2. Roth, S. Disenfranchised by Design: Voting Systems and the Election Process. Volume 9 of the Information Design Journal. 1998.
3. Fischer, E. Voting Technologies in the United States: Overview and Issues for Congress: CRS Report for Congress, Order Code RL30773 Congressional Research Services Library of Congress. 2001.
4. Sequoia Voting Systems. 2004.
5. Diebold Election Systems. 2004.
6. Caltech/MIT Voting Initiative. Prototype Archive. [http://cac.media.mit.edu/Voting/Prototypes.htm] 2005.
7. Furnas, G. W.: 1986, 'Generalized fisheye views'. CHI'86, pp. 16-23.
8. Krug, S. Don't Make Me Think. Indianapolis, IN Que. 2000.
9. Quesenbery, W. Defining a Summative Usability Test for Voting Systems. UPA Workshop on Voting and Usability. September 2004.