Opening Access to Environmental Software Systems

Daryl H. Hepting [a] and Timothy Maciag[a]

[a]Department of Computer Science
University of Regina
Regina, Saskatchewan
S4S 0A2, CANADA

Abstract: As individuals are increasingly being called to act in environmentally-conscious ways, they will seek out any and all resources which might help to inform their actions. Unfortunately, a 2002 OECD report indicated that there seemed to be declining trust of environmental information sources and increasing confusion about which actions could be most beneficial. Environmental Software Systems can provide relief in this context, if the software design is user-centered. The very specialized nature of much environmental software may discourage these design practices, but this is a false economy. Instead of software that is based on technical models developed by environmental scientists, consider that which can acquire and adapt to changing end-user models of the particular domain. Such adaptation would certainly benefit users, but research could also benefit considerably from data that could prioritize actions of the environmental scientists, from analysis to education. This shift in emphasis agrees with recent trends toward personalization and democratization of software system functionality. If the number of people who could meaningfully explore a model of a particular ecosystem could increase thousand-fold, there could be considerable benefit realized in the level of discourse on environmental issues pertaining to that ecosystem. Such an increased usage would require the removal of barriers for direct user access to the software systems in order to create satisfying user experiences. For the user to be satisfied when confronting a large and complex information space, he or she must not be overwhelmed but able to easily specify and locate that which is of interest. The theoretical basis for such an approach is presented, along with some evidence thus far collected. Opportunities for improvement are also discussed.

Keywords: human-computer interaction; environmental decision support; user-centered design; tradeoff support

1 INTRODUCTION

Whether one discusses environmental software systems (perhaps ESS), environmental decision support systems (EDSS), or web-based environmental decision support systems (WEDSS), the environment is the central application focus. However, it is hard to imagine any other area that is of greater general interest: humans cannot survive apart from the environment. The accessibility of all these forms of environmental software remains to be evaluated. Without detracting from the importance of issues specific to environmental software, one can easily find relevant work in areas such as e-commerce [Pu and Chen, 2005]. The purpose of all this software is to support the user in selecting his or her best alternative at any particular moment. This may mean easy specification of a particular alternative or evaluation of tradeoffs for many different alternatives.

There are a great many examples of valuable information which are underutlized because their existence is not known or it is practically impossible to put the desired information into a comprehensible form. A Natural Resources Canada website [1] provides an idling calculator but it presents results on a community basis where fuel savings might be on the order of millions of dollars and reduced emissions might fill several gymnasiums. What might be useful to an individual consumer is the number of trees he or she must plant in order to sequester the CO_2 emissions from his of her own automobile. Similarly, many products which contain hazardous materials are required to complete an MSDS (Material Safety Data Sheet). This MSDS information is available on the web [2], but not where a busy parent might look to evaluate his or her choice of cleaning

[1] http://oee.nrcan.gc.ca/transportation/tools/co2-calculator/
[2] for example, www.ilpi.com/msds/ and www.msdssearch.com/

product. To require a user to integrate these different sources of information is to violate a law of interface design [Raskin, 2000]: never make the user do more work than is absolutely necessary.

If one is to consider the impacts of a product based on its whole life cycle [Ciambrone, 1997], the number of items to consider may become truly staggering. Yet, decisions are likely made with relatively few bits of information, perhaps integrated from many others. How is it possible to choose which bits of information will be truly important to an individual's decision? Harder still if the vocabulary used to describe the problem is unfamilar. Consider volatile organic compounds (VOCs) as an example. These compounds are a contributor to smog, so they affect air pollution. However, some of these compounds are also known carcinogens. Which aspect of VOCs is more important, and to whom? What levels of VOCs are safe? Unfortunately, the answers to these questions are very complex. Environmental software systems have the potential to provide consumers with a great deal of useful information, and it must allow complex answers to be understood.

This paper will focus on the issues of creating an interface for satisfying and meaningful interaction, such that a user is able to complete a task without any sense of the computer as a tool, and he or she has confidence that the answer meets his or her needs. Therefore, the rest of the paper is organized as follows. Section 2 describes some of the background from a human-computer interaction perspective. Section 3 describes some of the performance evaluations done, with the interpretation that better performance (in terms of response times and task scores) corresponds to more satisfying interaction. Section 4 describes evaluation of the success participants had while working with the tools in finding their desired results. Section 5 presents conclusions and future work.

2 Background

Rizzoli and Young [1997] described three different types of users for EDSS: scientists, managers, and stakeholders. This paper is focused on the last group, since their insights will help to inform the activities of the other two groups. If the interface is accessible and easy to learn, individuals will be empowered to explore their unique combinations of value judgments and personal constraints which lead to solutions.

The world wide web is an excellent source of information for consumers, but the quantity and variety of this information might be hard to integrate without leaving confusion. Therefore, web-based decision support tools can provide an important structure within which consumers can search for product information. In the Fall of 2002, the United States Environmental Protection Agency's Environmentally Preferable Purchasing pilot project [3] was the destination returned by google [4] in response to the query "cleaning products environmental information" that was most suited to comparisons. The EPA provided three wizards for this purpose: single attribute ranking tool (SART), multiple attribute ranking tool (MART), and weighted attribute ranking tool (WART). These wizards provided access to a database of 29 cleaning products on the basis of eight attributes (skin irritation, food chain exposure (bioconcentration factor), air pollution potential (percentage of volatile organic compounds), contains fragrance, contains dye, product is a concentrate (reduced packaging), packaging is made of recyclable paper, and product minimizes exposure to concentrate). Each of these wizards presented results in a tabular format. Such a format is consistent with the "ranked list" format prevalent with many web search tools. However, this format is not conducive to tradeoff analysis. Although the wizards were intended for institutional buyers, it is appropriate to study them within the general population for two reasons: not all of these buyers might be experts in cleaning products and expertise in general is a difficult concept. The novice/expert distinction is less a dichotomy than a continuum [Raskin, 2000].

Faceted classification, as explored by Yee et al. [2003], has been used for searching and browsing activities. It does not easily support tradeoff analysis. Example-based approaches [Pu and Chen, 2005] can be preferrable because they can allow users to explore, are easy to use, and they support complex tradeoff tasks.

Pu and Chen [2005] use an apartment-finding problem as the basis for their study. The problems addressed by environmental software systems are less well-known, and the presentation of complex ideas such VOCs is difficult even when the vocabulary is shared. There are also technical concerns that may seem small and yet have wide-ranging implications. For example, the choice of ranges that are presented to users in the interface. The EPA wizards presented

[3] see `http://www.epa.gov/oppt/epp/pubs/cleaners/select/`. Although the wizards are no longer available, documentation about the project remains.

[4] http://www.google.com

the following choices for VOC levels: $< 1\%$, $< 5\%$, $< 10\%$, $< 20\%$, $< 35\%$, and $\geq 35\%$. While one hopes that these value choices were based on studies and agreement amongst policy makers, it is not clearly the case. Assumptions about the reasons for these choices can be dangerous. In the absence of regulation on the matter, the following set of choices would better suit the data: $\leq 2\%$, $\leq 6\%$, $\leq 11\%$, $\leq 15\%$, $\leq 35\%$.

There are some general concerns when the vocabulary about a problem is not shared. The research of Schooler and Engstler-Schooler [1990] clearly demonstrated the potentially disruptive effect of verbalization (possibly through an unfamiliar command syntax) when one is not articulate in the language needed to describe a perceptual memory. In order to enable democratization without falling into this verbalization trap, the ideas of Kellys personal construct psychology (PCP) [Kelly, 1955], can be employed. Using a device called a repertory grid [Gaines and Shaw, 1993], triads of stimuli can be presented to a person whereby he or she is asked to separate the one stimulus that is judged most different from the other two, and then describe the basis for that decision. In this way, an individual is given the means to articulate personally meaningful terms. Card sorting and conjoint analysis [Green and Srinivasan, 1990] are related techniques that can help to capture a persons model of a problem without the interference of a foreign lexicon. However, they may be time-consuming and off-putting for the average user. Various classification approaches can be used to help make the distinctions between user types [Maciag and Hepting, 2005].

The issue of dimensions or attributes becomes important in distinguishing one object from others. These differences may not be fully understood at the outset, so the ability to interact with a system to more fully specify requirements is a powerful capability [Wegner, 1997]. It is important for each user to be able to easily use meaningful concept names, or easily name the concepts that he or she finds important, and then carry out interaction on that basis. It is also important to be able to reorganize concepts to suit present needs. What may begin as a Klondike space [Perkins, 1995] (where the user prospects without a clear indication of the direction to take) can then be transformed into a homing space (where clues are plentiful and the user can home-in on his or her goal). The next sections will deal with the issues of how to evaluate software on the basis of performance and confidence.

3 Performance Evaluation

Over the past two years, an evaluation of access to the cleaning product database has been undertaken. The evaluation has been done in two parts, each with 28 participants recruited from the University of Regina Computer Science Department Participant Pool [5] For each of the three wizard interfaces described above, participants completed three questions involving one , two, and three attributes. The order in which participants used the interfaces was balanced, but always the same questions were used with each interface.

In the second evaluation, two variations (one graphical, one textual) of an example-based interface and were developed to explore the cleaning product data. These were tested against the wizard interfaces. The same question sets were used and they were associated with an interface through all trials. However, the mapping was not the same. This revealed some issues with the question selection and the ease of question completion with respect to the interface. Responses were scored based on the number of correct attributes. This second study is the focus of this paper.

A 3$\times$3 ANOVA (analysis of variance) was performed on the user response times and task scores to indicate whether there were any interactions between the number of attributes and the interface type for these two performance measures. Effect sizes are used to validate the strength of any significant trends and patterns observed in the results. Here, they are reported in terms of Cohen's f [Cohen, 1977].

Table 1 and Figure 2 illustrate significant interactions ($p < 0.05$), in terms the interface type and a combination of interface type and number of task attributes, for user response times. The size of these effects were medium ($\geq$ 0.25) to large ($\geq$ 0.40) [Cohen, 1977]. A similar trend appeared for user task scores between the different interfaces. Again, the example-based interfaces fared much better than the wizard interfaces, as illustrated in Table 2 and Figure 3.

When analyzing user response times and task scores across the interfaces, the benefits of the example-based interfaces become more clear as the complexity increases. For tasks involving only one attribute, user response times were consistent across the in-

[5] The participant pool allows undergraduates to participate in departmental research in exchange for a small bonus (1% for each hour of participation, to a maximum of 2% per course)

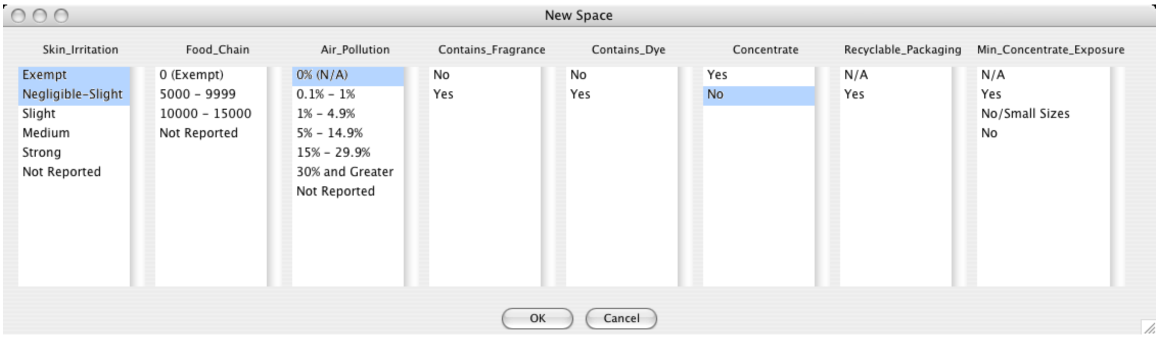

Figure 1: Screen shot of example-based query dialogue. Any selected value in a column is matched. If a column has no selections, all values in that column can be matched.

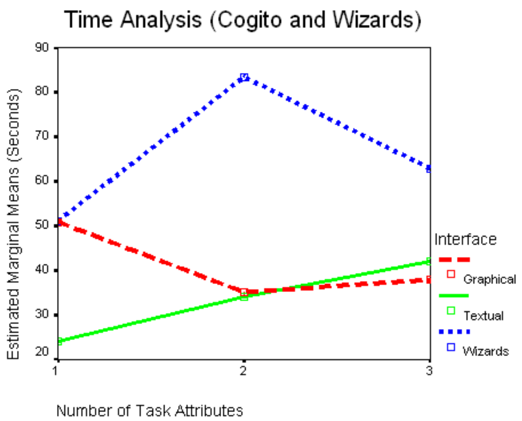

Figure 2: User task analysis comparing user response times on the wizard and example-based (graphical and textual) interfaces.

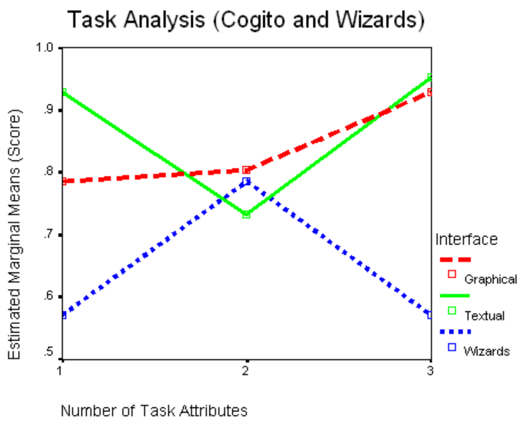

Figure 3: User task analysis comparing user task scores on the wizard and example-based (graphical and textual) interfaces.

terfaces. However, as the tasks involved more attributes, user response times increase dramatically only for the wizard interfaces. Fairly consistent task scores across interfaces were observed for tasks involving two attributes. However, the example-based interfaces are the clear winner for tasks involving three attributes. The trend indicates this performance benefit would carry on to four or more attributes, but it is not clear that users would employ any more than three attributes in any queries they may formulate.

Source	F	p	$Cohen's f$
int	**20.07**	**0.000**	**0.540**
att	3.87	0.028	0.127
int × att	**10.27**	**0.000**	**0.324**

Table 1: ANOVA within-subjects effect analysis comparing factors interface (int) and attribute (att) for user resposne times on the wizard and example-based interfaces.

Finally, participants were asked to rate their preference for each interface with each number of attributes. A 3×3 ANOVA was performed on this data and the results are presented in Figure 4. The example-based interfaces were preferred, and this preference became more strong as the number of attributes increased. This analysis does not deal with differences in preference between the three wizard interfaces (SART, MART, and WART).

The tasks used in this study were constructed to test the participant's ability to use the interface as it was provided. The tasks could be completed without detailed knowledge of the attributes.

4 Confidence Evaluation

Each of the 28 participants was asked to rate each attribute on a scale of 1 to 4 (from "unimportant" to "very important"). These ratings were used to compute a score for each product in the following way.

Source	F	p	$Cohen's f$
int	**9.476**	**0.001**	**0.313**
att	0.674	0.513	0.070
int × att	**4.417**	**0.006**	**0.264**

Table 2: ANOVA within-subjects effect analysis comparing factors interface and attribute for user task scores on the wizard and example-based interfaces.

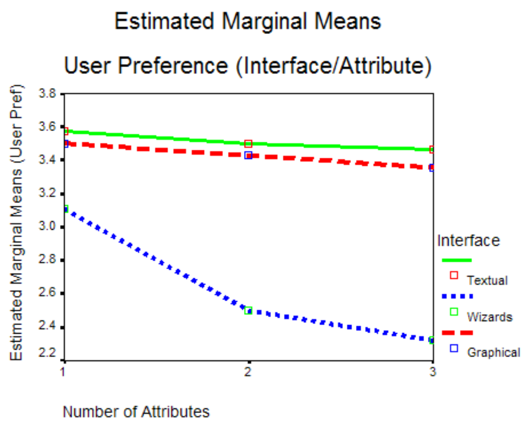

Figure 4: Graph showing the mean preference ratings for each interface, for each number of attributes. The textual and graphical example-based interfaces are rated consistently high, while the wizard interfaces are preferred less as the number of attributes increases.

First, the attribute ratings determined a multiplier for the attribute: a rating of 4 meant a multiplier of 1000; a rating of 3 meant a multiplier of 100; a rating of 2 meant meant a multiplier of 10; and a rating of 1 meant a multiplier of 1. In the product database, each product had a discrete value for each attribute that corresponded to the option menus in the software interface. For each product, a score was computed as the sum of the individual products of attribute multiplier (determined by the participant) and attribute value (stored in the product database). These scores were used to determine the ranking of all products for each participant. After completing the nine questions in task, described in Section 3, participants were asked to choose a product that they might use themselves. For each participant who replied, the rank of their chosen product was used to measure their error in finding a product that matched their stated values. 8 of 24 participants who selected a product for themselves chose the top-ranked product according to their stated preferences. On average, the error was 0.125 (SD=?)out of 1 which translated to choosing between the fourth and fifth ranked product on their list.

Furthermore, the participants were asked to provide definitions for four of the eight attributes which seemed less straightforward. Participant responses were scored and their average was 0.38 (SD=?) out of 1 (3 participants received a perfect score and 5 participants received a score of zero). These scores did not predict error in the personal product selection task, rather it seemed that participants did reliably equal or better on the personal product selection than their score on the attribute definitions. This result reinforced the observation that the product selection tasks were quite self-contained and could be completed without any additional knowledge.

5 Conclusions and Future Work

This paper has addressed some of the issues involved in opening access to environmental software systems. It has presented work towards two metrics of accessibility: performance (as an indicator of satisfaction) and confidence. Clear preference for the example-based interfaces has been shown, as was the case in work reported by Pu and Chen [2005]. Furthermore, this performance seems likely to extend to more complicated problems.

The question of whether people would choose to deal with more complicated problems (more than 3 attributes at once) is still open. The confidence of participants in the study described here is not clear. It is not possible to state whether users actually made errors in choosing their preferred product, because their new selections could be considered as a more accurate indication of preferences. Regardless, it seems that participants were not clear about the information which was available to them and so this presentation of information, in terms of these particular 8 attributes, needs to be revisited. The organization of the database and the names given to various concepts within it, which forms the user's interface to the database, is perhaps more important than the user's interface to the computer.

There were several issues that should be in future studies. Although the narrow segment of the population represented by undergraduate computer science students was appropriate for the performance evaluation, they were not appropriate for the confidence evaluation because of their limited shopping experience on the whole. Because the database used did not contain any pricing information, it was also difficult for participants to make meaningful comparisons and clear preference statements. The tested systems were found at times to return no results in response to a user's query. This could be quite damaging to a user's confidence. Instead, it seems to be a better alternative to present *similar* results if no direct matches can be found.

Pu and Chen [2005] studied apartment listings wherein the search terms had clear meanings to participants. However, their approach has many potential benefits for this type of study. In particular, Pu and Chen did two things of particular value: they worked to understand the preferences of their participants and how these preferences changed over time; and they began by asking their participants to make a personal choice of apartment and then performing tradeoff tasks to refine that personal choice.

Future work needed in this area includes better support for personalization in terms of attributes (which attributes are important to individuals) and better support for users at the computer interface. The first issue can be addressed by using repertory grids and conjoint analysis, to access which dimensions are important without resorting to explicit questioning. It is also important to develop methods to employ this personalization without taking too much of the user's valuable time in configuration of the system. The second issue can be tackled by considering and visualizing the structure of these spaces in terms of decision trees may provide some potentially valuable ideas about organizing decisions and understanding the underlying data. The starfield visualization technique, which enables display of dynamic query results based on slider movement, may also provide a valuable method of understanding ones choices.

ACKNOWLEDGMENTS

The authors wish to thank Peter Leavitt and Renata Bailey at the University of Regina for helpful comments regarding ecotoxicology. The authors also gratefully acknowledge the support of the Centre for Sustainable Communities at the University of Regina and the Natural Sciences and Engineering Research Council of Canada.

REFERENCES

Ciambrone, D. F. *Environmental Life Cycle Analysis*. Lewis Publishers, Inc., 1997.

Cohen, J. *Statistical Power Analysis for the Behavioral Sciences*. Academic Press, 1977.

Gaines, B. R. and M. L. G. Shaw. Knowledge acquisition tools based on personal construct psychology. *Knowledge Engineering Review*, 8(1): 49–85, 1993.

Green, P. E. and V. Srinivasan. Conjoint analysis in marketing research: New developments and directions. *Journal of Marketing*, 54(4):3–19, 1990.

Kelly, G. *The Psychology of Personal Constructs*. Norton, 1955.

Maciag, T. and D. H. Hepting. Analysis of user classifiers for personalization of environmental decision support system interfaces. In *Proceedings of ANNIE*, 2005.

Perkins, D. N. Insight in minds and genes. In Sternberg, R. J. and lastname Davidson, J. E., editors, *The Nature of Insight*, pages 495–533. MIT Press, Cambridge, MA, 1995.

Pu, P. and L. Chen. Integrating tradeoff support in product search tools for e-commerce sites. In *Proceedings of E-Commerce 2005*, pages 269–278. ACM Press, 2005.

Raskin, J. *The Humane Interface*. Addison-Wesley, 2000.

Rizzoli, A. E. and W. J. Young. Delivering environmental decision support systems: software tools and techniques. *Environmental Modelling & Software*, 12(2–3):237–249, 1997.

Schooler, J. W. and T. Y. Engstler-Schooler. Verbal overshadowing of visual memories: some things are better left unsaid. *Cognitive Psychology*, 22: 36–71, 1990.

Wegner, P. Why interaction is more powerful than algorithms. *Communications of the ACM*, 40(5): 80–91, 1997.

Yee, K.-P., K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 401–408. ACM Press, 2003.