Using knowledge structures to redesign an instructor-operator station

Article *in* Applied Ergonomics · April 2011

DOI: 10.1016/j.apergo.2011.03.002 · Source: PubMed

4 authors, including:

Russell J Branaghan
Arizona State University
51 PUBLICATIONS **234** CITATIONS
SEE PROFILE

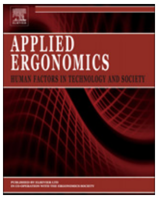
Christine Covas-Smith
Air Force Research Laboratory, Wright-Patterson Air Force Base
2 PUBLICATIONS **8** CITATIONS
SEE PROFILE

Applied Ergonomics 42 (2011) 934–940

Contents lists available at ScienceDirect

Applied Ergonomics

journal homepage: www.elsevier.com/locate/apergo



Using knowledge structures to redesign an instructor–operator station

Russell J. Branaghan [a,*], Christine M. Covas-Smith [a,b], Kenneth D. Jackson [a,b], Craig Eidman [b]

[a] *Arizona State University, AZ, USA*
[b] *Air Force Research Laboratory, Mesa, AZ, USA*

A R T I C L E I N F O

Article history:
Received 11 September 2010
Accepted 3 March 2011

Keywords:
Mental models
Menu design
Simulation training

A B S T R A C T

Frequently, user interface (UI) designers must choose between modifying an established, but suboptimal and familiar, UI or to avoid such changes. Changing the UI's, organization may frustrate users who have become familiar with the original design, whereas failing to make changes may force users to perform at an unsatisfactory level. This paper presents two studies that investigate whether users familiar with a poorly designed UI would find items faster, and prefer a reorganized UI that conformed to domain expert knowledge, or would their familiarity with the original UI yield faster performance and higher satisfaction.

This paper describes activities to redesign a menu structure in a simulator instructor–operator station (IOS) using hierarchical card sorting and cluster analysis (Romesburg, 2004). This analysis was used to reorganize the menu structure to reflect the knowledge representations of domain experts in accordance with the principle of proximity compatibility (Wickens and Carswell, 1995; Rothrock et al., 2006). The new design was validated with a separate set of users by a reaction time experiment and preference selection.

1. Introduction

When attempting to improve user performance and increase satisfaction, user interface (UI) designers are often faced with a dilemma. Is it better to modify an established UI or to leave a suboptimal UI as is? Making small modifications in color or background should be safe, but tampering with the organization of the UI, including the location of items, could be problematic. Reorganizing the UI risks frustrating users who have become familiar with the original design, whereas failing to reorganize forces users to adapt to a poor design, and perhaps perform at an unsatisfactory level.

This was the question faced by our team at the Air Force Research Laboratory (AFRL) in Mesa, AZ. The UI for a training simulator's instructor–operator station (IOS) had been designed with minimal input from subject matter experts. The rest of the design had evolved over time as users requested additional functionality. Observational research suggested that users had difficulty locating items efficiently. For example, the UI did not seem to conform to the principle of proximity compatibility (Wickens and Carswell, 1995; Rothrock et al., 2006), which states that concepts that are proximal in the user's knowledge structure should also be proximal in the UI.

The instructors possessed far more expertise with the operational domain than with the IOS, having spent years honing their skills as military pilots, joint tactical attack commanders (JTAC) and the like. As a result, their knowledge structures were more likely based on conducting the tasks in the real world than on the design of the IOS UI. This suggested that redesigning the UI to conform to the instructors' knowledge structures would improve the ease with which instructors could locate features and functions.

On the other hand, these instructors might perform better with the familiar, albeit imperfect, UI design. The present research sought to determine if instructors, familiar with the IOS UI would find items faster, and prefer a reorganized UI that conformed to their operational knowledge, or would their familiarity with the original UI yield faster performance and higher satisfaction.

This paper describes activities to redesign one important menu structure in the IOS, by 1) using hierarchical card sorting techniques to elicit relationships among items in a large menu structure; 2) conducting a hierarchical cluster analysis (Romesburg, 2004) to represent those relationships in a dendrogram; 3) reorganizing the menu structure to reflect the user's knowledge representation in accordance with the principle of proximity compatibility (Wickens

---

[*] Corresponding author. Cognitive Science and Engineering Program, Arizona State University, 150 D Santa Catalina Hall, 7271 E. Sonoran Arroyo Mall, Mesa, AZ 85212, USA. Tel.: +1 480 727 1390; fax: +1 480 727 1538.

E-mail address: Russ.branaghan@asu.edu (R.J. Branaghan).

0003-6870/$ – see front matter © 2011 Elsevier Ltd and The Ergonomics Society. All rights reserved.
doi:10.1016/j.apergo.2011.03.002

and Carswell, 1995; Rothrock et al., 2006); and 4) validating the redesign with a separate set of expert instructors using a reaction time paradigm and preference selection.

1.1. Scenario based simulation (SBS)

Training is vital to success in many knowledge intensive domains. Simulation training (Andrews and Bell, 2000; Harris and Kahn, 2003) enables students to experience realistic scenarios, communications, time frames and decision requirements without the risk or expense of *in situ* training. This provides practice that cannot be achieved with audio-visual presentation, lecture, or paper and pencil methods. Further, students can be trained without engaging in life-threatening activities, and debriefing capabilities enable the student to rerun the simulation, consider alternative actions, and discuss the training episode with an instructor. Simulator training engages users in synthetic environments in medicine (Park et al., 2010; Schwid et al., 2002), aviation (Wickens et al., 2002), military (Andrews and Bell, 2009), space exploration (Cyril et al., 2000), driving (Fisher et al., 2002), and other fields.

Scenario based simulation (SBS) immerses trainees in realistic situations, enabling them to practice skills, and receive feedback. Military pilots make extensive use of SBS (Andrews and Bell, 2009; Bell and Waag, 1998; Fowlkes et al., 1998). For example, the Experimental Common Immersive Theater Environment (XCITE) is used at the AFRL in Mesa, AZ to build and maintain scenarios for pilot and other military training. XCITE provides physics-based aerodynamic models, and the ability to control numerous entities within a scenario concurrently, facilitating realistic complexity and diversity within a task. Using XCITE, military trainees perform simulated missions emulating important facets of their mission, including equipment (e.g., ground to air missile systems), combat scenarios (e.g., air- to- air and air-to-ground combat), conditions (e.g., realistic terrain, weather, civilian populations), and infrastructure (e.g., dams and bridges).

These are crucial factors in military operations, which must often accommodate multiple constraints. For example, the pilot may need to destroy a bridge without killing civilians or being shot out of the sky by a ground to air missile. In this sense, the scenario itself acts as the curriculum by providing situations that expose the trainee to the knowledge and skills to be learned. Efficient training results from well-planned scenarios, which ensure practice with the correct content and timely assessment and feedback.

1.2. XCITE's instructor–operator station

Although XCITE facilitates training, the human instructor remains an active component of instruction, attempting to maximize effectiveness within constraints of time and cost. Instructors script scenarios, monitor training, control numerous entities, and provide feedback. In doing so, they consider factors such as the fidelity of the scenario (Satish and Streufert, 2002), variations in the circumstances, alternation of task modules (Goettl et al., 1996), unanticipated difficulties and barriers, as well as the frequency, timeliness and specificity of feedback (Cannon-Bowers and Salas, 1997). Thus, the tasks performed by the instructor are numerous, multifaceted, and dynamic. Often, instructors do these things for multiple trainees at once. As a result, it is important that they can locate items in the UI quickly.

These activities are accomplished using an instructor–operator station (IOS) composed of hardware and software elements. The hardware includes a desk with multiple displays and a data wall, which depicts the instrumentation and other elements that the trainees are shown. The software includes interfaces like XCITE, which enable instructors to view the entire scenario from a god's eye view as to zoom down into the scenario to look at individual entities and groups of entities.

XCITE's software user interface was designed by software engineers based on the opinions of a few instructor pilots, who had little experience in UI design. According to those involved, they focused on ensuring the required functionality rather than on ease of use. This is common in software design when the UI is assigned to computer programmers or engineers without specialized experience or training in UI design (Paap and Cooke, 1997). The result is often an interface that is difficult to learn, inefficient to use, and does not behave the way users expect.

1.3. Domain knowledge and proximity compatibility

Air Force instructors are usually domain experts with many years of experience conducting the activities they are training. For example, if they are training combat pilots, they have many hours of experience flying combat missions. As a result, one would expect that they have well developed knowledge structures of these domains and activities. On the other hand, the instructors usually do not have nearly as much experience with the IOS. Indeed much of the training they deliver is live, within classrooms or in actual cockpits during live training missions.

Consequently most instructors are domain experts, but are not experts in the use of the IOS. Thus, the usability of the interface is important. If instructors cannot locate options quickly they waste valuable training time. Since the users of the XCITE IOS are domain experts, the design should capitalize on the their existing knowledge to guide characteristics such as the organization of menu content.

The organization of the UI's menus within the IOS should adhere to the proximity compatibility principle (Wickens and Carswell, 1995; Rothrock et al., 2006). That is, concepts that are close to each other in the user's mental model should be placed close to each other in the menu structure of the UI as well. This way mental and perceptual (i.e. display) proximity agree.

Indeed, previous research has suggested that menu organization is enhanced by user input. For example Hayhoe (1990) found that a pull down menu organized on the basis of a card sorting exercise by users yielded fewer errors, faster selection and better recall, than a menu based on the card sorts of four programmers. Similarly, (Roske-Hofstrand and Paap, 1986) found that pilots could locate menu panels in NASA's advanced concepts simulator when the panels were organized according to the judgments of users rather than programmers.

1.4. Knowledge elicitation methods

There are many methods for eliciting judgments of relatedness among users (Cooke, 1994). Card sorting asks judges to assign items to piles based on their relatedness. Then related piles can be nested within other piles forming superordinate–subordinate relationships. A distance matrix is then created for each participant. In the matrix, pairs of items in the same pile (and within the same parent pile) are assigned a distance of 0. Pairs in adjacent piles receive a distance of one, pairs that are two piles away receive a distance of two and so on. These matrices are then averaged across participants to derive an average distance matrix. Sorting is faster than asking participants to provide relatedness ratings for each pair of items. This was necessary in our study, since access to instructors is difficult to obtain. They are busy with little time to spare.

Alternatively, we could have asked domain experts to simply redesign the interface for us. However, reconciling differences among the experts' opinions was challenging. Expert instructors can be opinionated and even argumentative. Thus, the time required to reach consensus is cost prohibitive. Instead, card sorting required only about 45 min per instructor.

2. Study 1. knowledge elicitation, representation and menu redesign

This study focused on reordering and repositioning the items in XCITE's declutter menu. At present, this menu is located on the right hand side of the main map screen, which is the screen that the users interact with most frequently. Fig. 1 depicts the graphical user interface of XCITE and the declutter menu that is located on the far right hand side of the interface. Each item in the declutter menu is related to a specific category of scenario management and design. Its purpose is to provide easy access to the most common functions needed by instructors and to enable users to show or hide information depending on the scenario. In essence, the declutter menu allows users to reduce the data on the screen. This is important because, if instructors are presented with more information than required, they may need to perform a detailed search to find required functions. This additional search time reduces the time that instructors can focus training. Anecdotal reports from instructors who use the XCITE IOS indicate that it is difficult to find items in this menu, and that it is disorganized. The order of items in the declutter menu is shown in Table 1.

Table 1
Order of Items in original "declutter" menu.

Air	Trails	Bullseye	Connect	Legend
SAM	OCC	Beam	DLINK	Target
AAA	LOS	C/S	ISR	Cultural
Ground	Control	Elevation	Attach	Static
Grid	Scripts	Alt-msl	MTI	Marker
Map	Groups	Alt-agl	Kills	K-Mode
LZ	Alert	Speed	Detonate	S-mode
CAP	XCITE	Heading	Hostile	S.A.E.
Route	J-Fire	IFF	Friendly	

Note. To preserve space in this document, the menu is depicted as a table in 5 columns. In the actual product, the menu is simply a one-column list of the forty-four items.

In this study, participants performed a hierarchical sort of the 44 declutter menu items, and the resulting data matrix was submitted to a complete hierarchical cluster analysis (HCA) (Romesburg, 2004). HCA transforms the distances to a hierarchical set of clusters, and items with smaller distances according to the original data matrix are most likely to be placed in the same cluster. HCA outputs a dendrogram, which illustrates each item and the point at which each pair of items intersect. The most related items intersect first, and the least related last (Cooke, 1999).

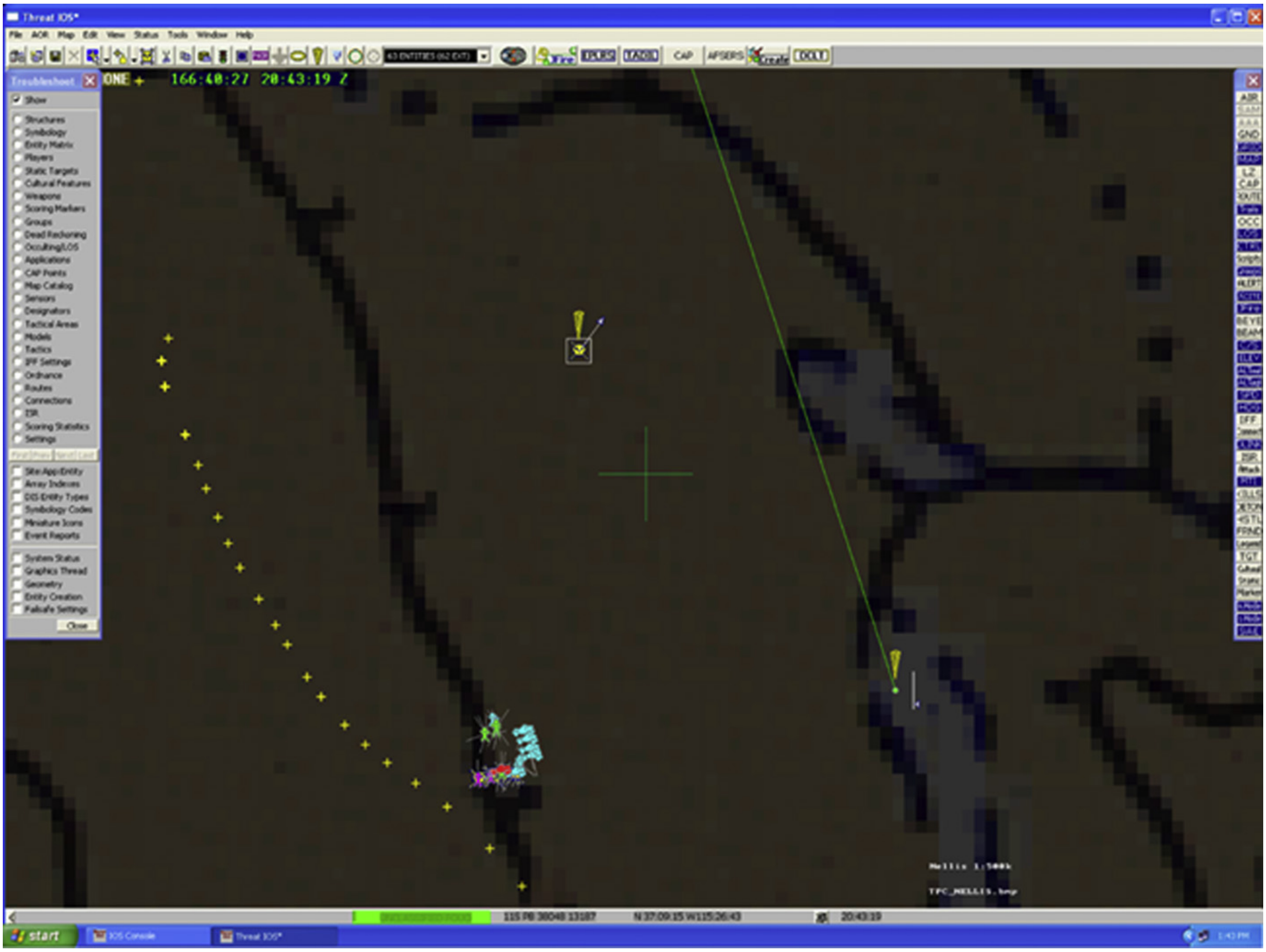


Fig. 1. Screen capture of XCITE's graphical user interface. The declutter menu is shown on the right hand side.

2.1. Method

2.1.1. Participants

Eight instructors (7 males, 1 female) participated in the card-sorting task. Each provided informed consent prior to participating. The mean age of participants was 40.3 years (range = 26–59 years). Flight experience ranged from fast-jet aircraft to fixed wing and heavy aircraft and mean flight hours were 1954 h (range = 1500–3500). Though all participants had a working knowledge of XCITE, their experience with it was primarily at the novice level (<6 months).

2.1.2. Stimuli and Apparatus

An Apple MacBook laptop was used to collect data. The 44 items within the declutter menu (listed in Table 1) were used as cards. Card sorting software, xSort (Arroz, 2008) was used to collect data. Fig. 2 depicts a screenshot from the exercise.

2.1.3. Procedure

Participants were provided with informed consent, and completed a brief survey about their familiarity with XCITE, years of flying experience, and frequency of use. Participants were then instructed to sort the cards so that the most related cards were placed into the same group. They were encouraged to create as many groups as they thought proper, and to nest groups within larger groups if appropriate. Mean completion time for the card-sorting task was 45 min. The order of the cards in the xSort application was randomized for each participant at the start of data collection.

2.2. Results and discussion

The card sorting data were converted to a distance matrix in which each cell refers to the distance between each item and each other item. Pairs placed in the same pile were given a distance measure of 0. Pairs in adjacent piles were assigned a distance of 1. Pairs that were two piles away were assigned a distance of 2, and so on. These matrices were then averaged over the eight participants and submitted to a complete-linkage hierarchical cluster analysis (Romesburg, 2004). The resulting dendrogram is shown in Fig. 3.

Determining the appropriate number of clusters requires some interpretation of the dendrogram in the context of the task. We attempted to identify a reasonable number of menu-item clusters and items per cluster. To identify a cutoff criterion for cluster identification, Fig. 4 plots the number of clusters against the cluster coefficient. The graph suggests two possible cutoff values. The first, 0.55, results in 13 clusters, or an average of about 3.4 menu items per cluster. However, with this value, some clusters contain only two items. The second criterion value was 0.75. This criterion yielded six clusters containing between five and nine items per cluster. Opting against some clusters with only 2 items, the second criterion was chosen as the most reasonable.

2.2.1. Menu redesign

The resulting clusters are shown in Table 2. An expert instructor, who was not involved in the study, then ranked the six clusters according to their importance. Clusters were then entered into the menu according to those rankings, with the most important items at the top, followed by the second most important, and so on.

One premise of cognitive engineering is that user interfaces should often mirror the knowledge of domain experts (McDonald et al., 1988; McDonald and Schvaneveldt, 1988; Paap and Cooke, 1997), as well as the principle of proximity compatibility (Wickens and Carswell, 1995; Rothrock et al., 2006). Study 2 investigates the merit of the redesigned menu structure. Specifically, we compared the speed with which experienced XCITE users (all of whom were expert instructors) could locate and select menu

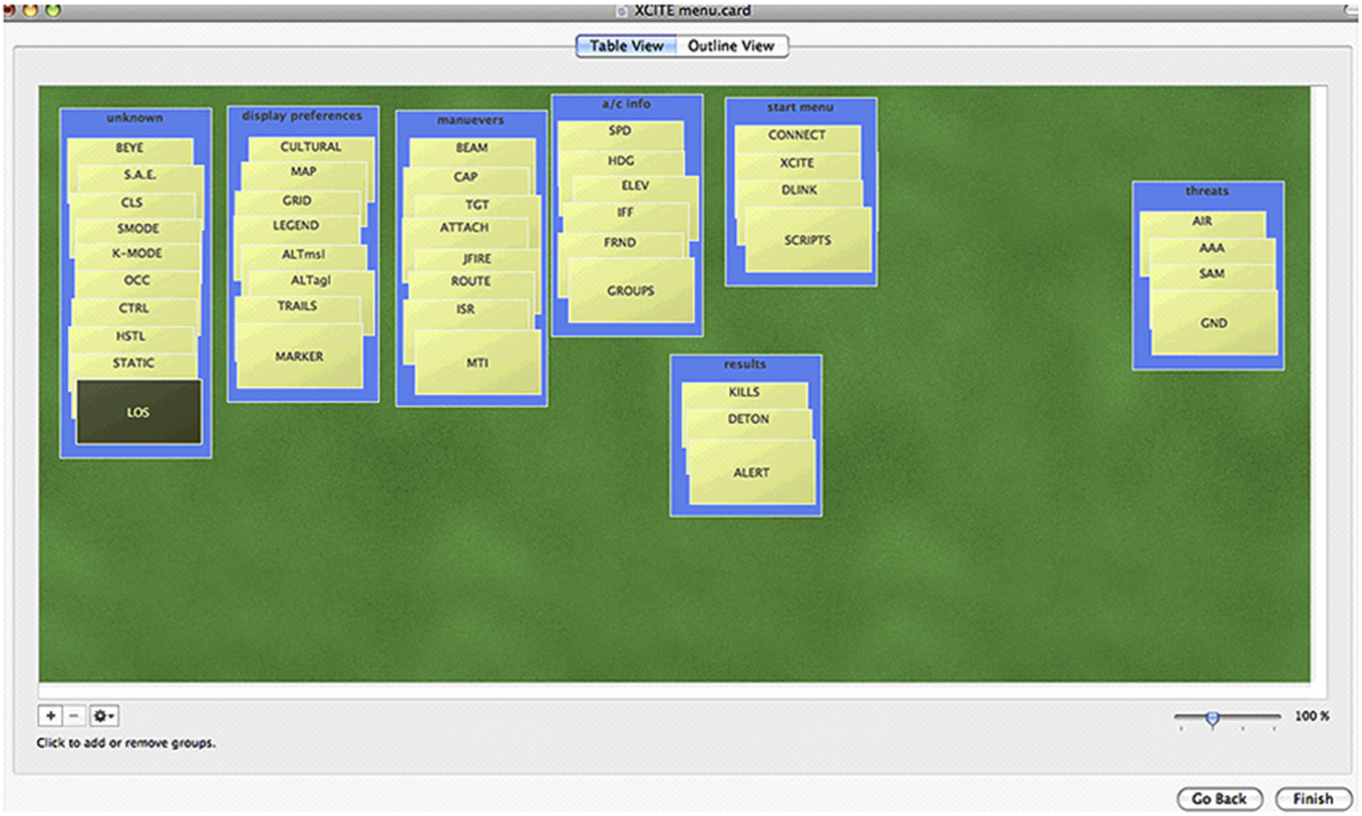


Fig. 2. Screenshot taken from the xsort application depicting a completed card sorting task.

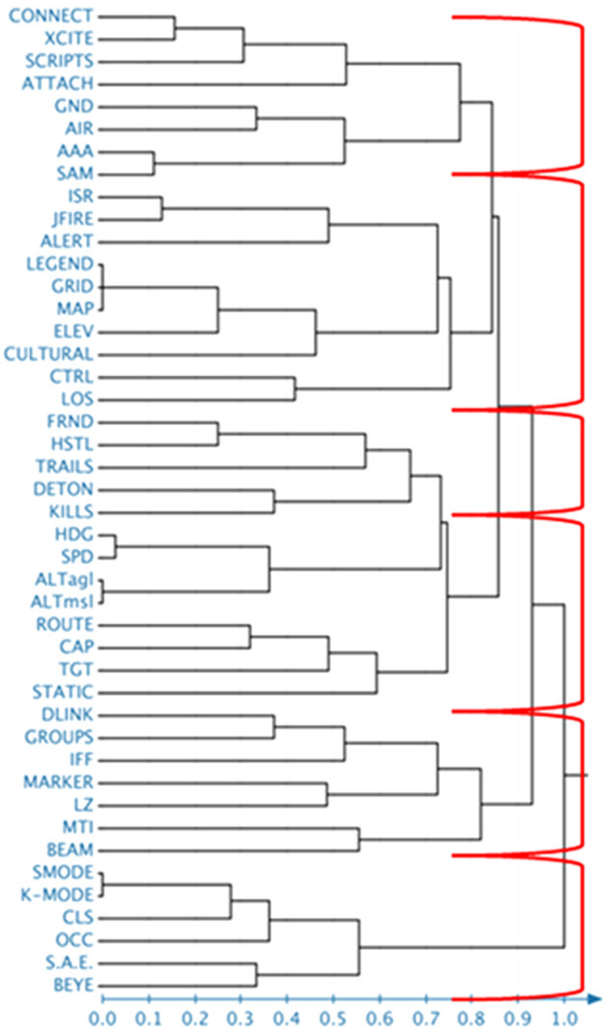


Fig. 3. Dendrogram resulting from the analysis of the data matrix by a complete-linkage hierarchical cluster analysis. Terms in the left designate the terms that were submitted to the matrix. The *x* axis represents the cluster coefficient. The brackets on the far right represent the clusters that were taken as the order of items in the clusters within the redesigned menu. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

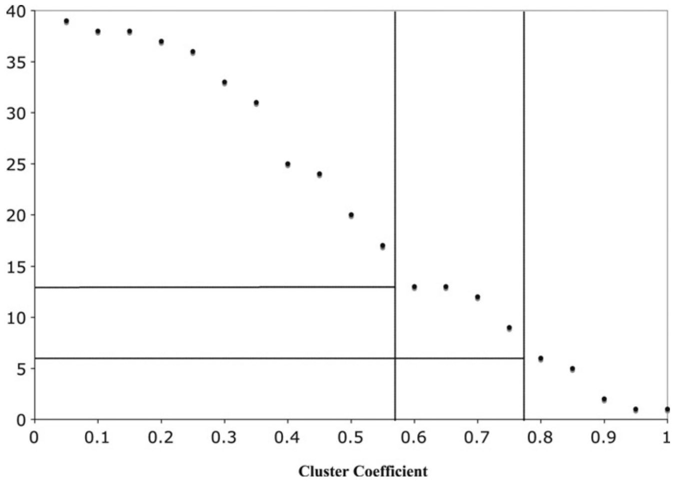


Fig. 4. Plot of the number of clusters by cluster coefficient.

Table 2
Order of items in the redesigned menu.

Beam	Detonate	AAA	Bullseye	Alt-agl	Alert
DLINK	Friendly	Air	C/S	Alt-msl	Control
Groups	Hostile	Attach	K-Mode	Cap	Cultural
IFF	Kills	Connect	OCC	Heading	Elevation
LZ	Trails	Ground	S-mode	Route	Grid
Marker		SAM	S.A.E.	Speed	ISR
MTI		Scripts		Static	JFIRE
		XCITE		Target	LOS
					Map

items in the new menu and the old menu. Additionally, we investigated which of the two menus these experienced XCITE users preferred.

3. Study 2: redesign and evaluation of declutter menu structures

Study 2 evaluated the effectiveness of the redesigned menu structure by investigating the speed with which experienced XCITE users could identify and select menu items from the redesigned menu and the original menu. At the end of this task, participants were presented with both the original and redesigned menus and asked to choose the one they preferred. If the new menu structure is better, and if the methods of card sorting and HCA were valid, we would expect participants to find and select items faster from the redesigned menu. Further, we would expect them to prefer the redesigned menu to the original. On the other hand, since the participants were all current and frequent users of the original XCITE menu structure, we might expect that they would locate and select items faster using the original menu structure, and they might prefer it as well. Because of the participants' familiarity with the original menu, this represents a strong test of the redesign.

On each trial, participants were presented with one menu-item label to locate in either the original or redesigned menus (depending on condition), and were timed as they located, and clicked on it. Participants were instructed to click each item as quickly as possible. After completing the reaction time tasks, participants were shown both menu configurations and asked to indicate which structure they preferred. We predicted that participants would locate and select items on the redesigned structure faster than the original menu structure. We also predicted that participants would also prefer the redesigned menu structure.

3.1. Method

3.1.1. Participants

None of the participants in this study participated in Study 1. Thirteen XCITE users (12 males and 1 female) participated in the experiment. All participants received and provided informed consent prior to participation. Mean age of participants was 41 (S.D. = 9.9, range = 27–50 years). Four of the participants had used XCITE for 6 months or less, 2 had used XCITE between seven and 12 months, four had used XCITE between one and three years and three had used XCITE for more than three years. Three participants indicated that they used XCITE less than once per year, three indicated that they used it monthly, three weekly and two daily.

3.1.2. Stimuli and Apparatus

The stimuli consisted of the original and redesigned declutter menus. The experimental trials were administered via Inquisit (Inquisit, 2006), a software package used to design surveys and reaction time experiments. Inquisit enables the experimenter to send a link directing participants to an experiment, and then downloads a java applet to the participant's computer, enabling the computer's internal clock to be used for collecting reaction times.

3.1.3. *Procedure*

Prior to the evaluation, the participants were sent an informed consent document via email. As soon as they returned the document, they were emailed a link to the software. Upon opening the link to the experiment, participants were asked to complete a brief demographic survey that also included questions related to XCITE experience.

Participants were instructed that, because timing measures were being collected, they were to complete the evaluation in their office, alone, with the door closed and without any distractions. These instructions, and the reasons for ensuring that there should be no distractions were reiterated verbally by their commanding officer. Once the participants clicked the link to start the task, Inquisit presented the demographics/XCITE usage survey followed by instructions on how to complete the task. For the menu-search task, participants were shown the name of an item taken from the declutter menu, in large print on the center of their computer screen. As items were displayed, participants were asked to locate and click the item within the menu on the right hand side of the screen as quickly as possible. The experiment was run within-subjects (that is, all participants received both conditions) and the order of conditions was counterbalanced across participants. After completing the task with one menu, the participants were then provided with the alternate menu. As a result all participants received both conditions and counterbalancing ensured that there was minimal if any effect of the order presentation of one menu vs. another.

There are 44 items contained in both the current menu and redesigned declutter menus. Thus, participants performed a total of 88 individual search and click trials. Completion of the entire task took less than 30 min.

3.2. *Results and discussion*

As hypothesized, participants found and selected items in the redesigned menu ($M = 3903$ ms, $SD = 672.9$) faster than in the original menu ($M = 4594$ ms, $SD = 1067.8$; $t\,(12) = 2.47$, $p = 0.03$; Cohen's $d = 0.68$). Overall, subjects responded 15% faster with the menu designed based on the expert knowledge structures. Additionally, participants preferred the redesigned menu structure to the original structure ($\chi^2\,(1) = 6.23$, $p = 0.013$).

Overall, we found that the redesigned menu structure significantly improved reaction time and search time to locate items within the menu. Further, not only were the interactions with the menu improved by the redesign, we also found that even the majority of experienced users preferred the redesigned menu structure. These results support the benefits of using expert domain knowledge as a starting point for the design of menu structures, even when redesigning a UI users are familiar with.

4. General discussion

UI designers are often faced with a dilemma of whether to redesign an established UI or to leave it as is. Reorganizing the UI can frustrate users who are familiar with the original design, whereas failing to reorganize forces users to continue employing a poor design. This paper described activities to redesign a menu structure in the XCITE IOS by using card sorting, HCA, and the principle of proximity compatibility (Wickens and Carswell, 1995; Rothrock et al., 2006). It then validated the redesign using a reaction time paradigm and preference selection.

Scenario based simulation products like XCITE have become popular in training, enabling trainees to practice practical tasks in a realistic context without substantial risk or cost. Although a simulator is used, the instructor remains an essential part of training. As a result, the UI for the IOS should be easy to learn and efficient to use. Unfortunately, as in other domains, the usability of these products can take a back seat to functionality. Often this is because programmers with little training in human factors are responsible for the user interface design.

A cognitive engineering approach (McDonald and Schvaneveldt, 1988; Norman, 1988) to UI design exploits the knowledge structures of domain experts. In this way, concepts that are close together in the domain expert's knowledge structure are placed close together in the UI, thus adhering to the principle of proximity compatibility (Wickens and Carswell, 1995; Rothrock et al., 2006). Since the instructors who use XCITE are domain experts who possess many years of experience, XCITE represents a prime candidate for this approach.

Study 1 employed hierarchical card sorting to elicit the relationships among concepts in XCITE's largest (44 items) and most frequently used menu, the declutter menu. A hierarchical cluster analysis of these data illustrated that the menu could be divided into six clusters. The order and position of items within the declutter menu were then rearranged to match the hierarchical cluster analysis.

In Study 2, XCITE users were shown target menu items, and asked to locate these items in the original or redesigned menu (depending on condition) as quickly as possible. Participants were significantly faster finding items in the redesigned menu than in the original menu. Moreover, participants preferred the new menu structure to the original. These findings are particularly strong since all of the participants were active users of the original XCITE menu system.

As has been demonstrated in previous research, (Hayhoe, 1990; McDonald and Schvaneveldt, 1988; Roske-Hofstrand and Paap, 1986), the findings reported here provide support for the benefits of organizing menu systems based on user knowledge. Designers, engineers and programmers are rarely experts in the domain they design for. Consequently, methods like the one demonstrated here, that enable them to uncover expert knowledge structures and then to configure UIs accordingly, would be welcome. This seems particularly helpful for designers who consult for many clients in a wide variety of industries. In consulting designers and engineers need to get up to speed very quickly. This approach enables them to mine and represent the knowledge of experts in short order.

Importantly, the current study provides validation using both performance and preference data, rather than just one. Moreover, the participants used in Study 2 were active users of XCITE who were quite familiar with its original menu structure and completely unfamiliar with the redesigned menu. This finding suggests that, when a redesign reflects expert knowledge, the benefits of changing the design can outweigh the benefits of remaining consistent with a poor design. If the original design is bad enough, and the new design is good enough, making changes can have an immediate impact on user performance and preference.

This study has a few limitations, which present opportunities for future research. We relied heavily on the cluster analysis, without asking the expert instructors to provide an interpretation of the dendrogram. In applied settings, it may be constructive to ask users to explain the dendrogram, thus providing a starting point for further interview. The dendrogram could provide a helpful shared point-of-reference, and the responses provided by the experts might provide further insight for design. Following this approach, the designer could generate an initial menu configuration, and follow that with usability testing and iterative design activities.

A limitation of Study 2 is that it focuses on only the fastest and simplest type of menu search, identity matching (Paap and Cooke, 1997). This happens when the search item (stimulus) is identical to the target item. For example, if the stimulus item was "Map", the target item was also "Map". A different type of search, equivalence matching (McDonald et al., 1983; Paap and Cooke, 1997) occurs when the user knows what he is looking for, but does not know its name. In this type of task, the user might receive a stimulus such as "a picture that shows how things are laid out in space". The target item, of course, would be "Map". This type of matching requires a more thorough semantic analysis. Future research should also investigate this search type.

References

Andrews, D., Bell, H., 2000. Simulation-based training. In: Tobias, S., Fletcher, J. (Eds.), Training & Retraining: a Handbook for Business, Industry, Government, and the Military. American Psychological Society, New York, pp. 357–384.

Andrews, D., Bell, H., 2009. A virtual environment application: distributed mission operations. In: Cohn, J., Nicholson, D., Schmorrow, D. (Eds.), Virutal Environments for Training and Education: Developments for the Military and Beyond. Praeger Security International, Westport, CT, pp. 77–84.

Arroz, M., 2008. xSort. IPragma, Lisboa, Portugal.

Bell, H., Waag, W., 1998. Evaluating the effectiveness of flight simulators for training combat skills: a review. International Journal of Aviation Psychology 8, 223–242.

Cannon-Bowers, J., Salas, E., 1997. A framework for measuring team performance measures in training. In: Brannick, M., Salas, E., Prince, C. (Eds.), Team Performance Assessment and Measurement: Theory, Methods and Applications. Erlbaum, Mahwah, NJ, pp. 45–62.

Cooke, N.J., 1994. Varieties of knowledge elicitation techniques. International Journal of Human-Computer Studies 41, 801–849.

Cooke, N.J., 1999. Knowledge elicitation. In: Durso, F., Nickerson, R., Schvaneveldt, R.W., Dumais, S.T., Lindsay, D., Chi, M. (Eds.), Handbook of Applied Cognition. Wiley, New York, pp. 479–510.

Cyril, X., Jaar, G., St-Pierre, J., 2000. Advanced Space Robotics Simulation for Training and Operations Paper Presented at the AIAA Modeling and Simulation Technologies Conference.

Fisher, D., Laurie, N., Glaser, R., 2002. Use of a fixed base driving simulator to evaluate the effects of experience and PC-based risk assessment training on drivers' decisions. Human Factors 44 (2), 287–302.

Fowlkes, J., Dwyer, D., Oser, R., Salas, E., 1998. Event-based approach to training (EBAT). International Journal of Aviation Psychology 8 (3), 209–221.

Goettl, B., Yadrick, R., Connelly-Gomez, C., Regian, W., Shebilske, W., 1996. Alternating task modules in isochronal distributed training of complex tasks. Human Factors 38, 330–346.

Harris, D., Kahn, H., 2003. Response time to reject a takeoff. Human Factors and Aerospace Safety 3, 165–175.

Hayhoe, D., 1990. Sorting-based menu categories. International Journal of Man-Machine Studies 33, 677–705.

Inquisit, 2006. Inquisit 2.0.60616. Millisecond Software, Seattle, WA.

McDonald, J., Dayton, T., McDonald, D., 1988. Adapting menu layout to tasks. International Journal of Man-Machine Studies 28, 417–436.

McDonald, J., Schvaneveldt, R.W., 1988. The application of user knowledge to interface design. In: Guindon, R. (Ed.), Cognitive Science and Its Application for Human Computer Interaction. Lawrence Erlbaum & Associates, Hillsdale, NJ.

McDonald, J., Stone, J., Liebelt, L., 1983. Searching for items in menus: The effects of organization and type of target. Paper presented at the Human Factors Society, Norflok, VA.

Norman, D.A., 1988. The Psychology of Everyday Things. Harper, New York.

Paap, K.R., Cooke, N.J., 1997. Design of menus. In: Helander, M., Landauer, T.K., Prabhu, P. (Eds.), Handbook of Human Computer Interaction. Elsevier Press, Amsterdam.

Park, C., Rochlen, L., Yaghmour, E., Higgins, N., Bauchat, J., Wojciechowski, K., et al., 2010. Acquisition of critical intraoperative event management skills in novice anesthesiology residents by using high-fidelity simulation-based training. Anesthesiology 112 (1), 202–211.

Romesburg, C., 2004. Cluster Analysis for Researchers. Lulu Press, North Carolina.

Roske-Hofstrand, R.J., Paap, K.R., 1986. Cognitive networks as a guide to menu organization: an application in the automated cockpit. Ergonomics 39, 1301–1311.

Rothrock, L., Barron, K., Simpson, T.W., Frecker, M., Ligetti, C., Barton, R.R., 2006. Applying the proximity compatibility and control-display compatibility principles to engineering design interfaces. Human Factors and Ergonomics in Manufacturing 16, 61–81.

Satish, U., Streufert, S., 2002. Cognitive simulation in medicine: towards optimizing decision making performance of healthcare personnel. Quality and Safety in Healthcare 11, 163–167.

Schwid, H., Rooke, G., Carline, J., Steadman, R., Murray, W., Olympio, M., et al., 2002. Evaluation of anesthesia residents using mannequin-based simulation - A multiinstitutional study. Anesthesiology 97 (6), 1434–1444.

Wickens, C., Carswell, M., 1995. The proximity compatibility principle: its psychological foundations and relevance to display design. Human Factors 37, 473–494.

Wickens, C., Helleberg, J., Xu, X., 2002. Pilot maneuver choice and workload in free flight. Human Factors 44 (2), 171–188.