Using Cognitive Walkthrough Procedure to Prototype and Evaluate Dynamic Menu Interfaces: a Design Improvement

Xiangyu Wang
Lecturer, Key Centre of Design Computing and Cognition,
Faculty of Architecture, Design and Planning, The University of Sydney, Australia
x.wang@arch.usyd.edu.au

Abstract

Menu interface usability has been regarded as a critical factor affecting the quality of software applications. This paper developed a method for automatically generating intelligent and dynamic menu to reflect different needs from different types of end users. The contents and structures of the menu are dynamic and adaptive to the usage context (user and task). Such implementation was based on the Cognitive Walkthrough procedure and automatic instrumentation. This paper also described a pilot study performed to evaluate the effectiveness of the dynamic menu. Preliminary results showed that the dynamic menu interface reduced the time for menu search per task significantly than the normal menu interface. The total time for completing a series of tasks was also reduced significantly.

Keywords: dynamic menu, cognitive walkthrough, exploratory learning, usability.

1. Introduction

The aim of the work presented in this paper is to improve the effectiveness of interaction between end users and menu interfaces in computer software by making those menus more dynamic, intelligent and adaptive to the user's needs. Such dynamic menu can minimize the barrier between the human's cognitive model of what they want to accomplish and the computer's understanding of the user's task.

The method used in this paper to make menu interfaces dynamic is the cognitive walkthrough procedure, which is a usability evaluation method based on cognitive theory. The cognitive walkthrough is a technique for evaluating the design of a user interface (e.g., menu interface), with special attention to how well the interface supports "exploratory learning," (Rieman 1996). Exploratory learning refers to the cases where novices actually start to use interfaces without any formal training. The cognitive walkthrough procedure can support software developers and designers to identify requirements and refine specifications. This method can be combined with other user-centered evaluation techniques to produce software products that match more to users' working contexts.

In this paper, this method was performed by the author in the early stages of menu interface design, before an empirical pilot user testing was implemented. Both the prototyping of dynamic menu interfaces and the empirical pilot user testing were presented in this paper

2. Background of Cognitive Walkthrough

The cognitive walkthrough (CW) is practical evaluation technique grounded in Lewis and Polson's CE+ theory of exploratory learning (Polson, Lewis, Rieman, and Wharton 1992; Lewis and Rieman 1993; Wharton, Rieman, Lewis, and Polson 1994), which is an information-processing model of human cognition. This theory describes Human-Computer Interaction in four steps (Wharton, Rieman, Lewis, and Polson 1994):

1. The user sets a goal to be accomplished with the system.
2. The user searches the interface for currently available actions (menu items, buttons etc.).
3. The user selects the action that seems likely to make progress toward the goal.
4. The user performs the selected action and evaluates the system's feedback for evidence that progress is being made toward the current goal.

Wharton, Rieman, Lewis, and Polson (1994) stated that the CW is aimed at analyzing highly prompted interfaces and can help the analyst to examine the interplay between the user's intentions and the cues and feedback provided by the interface.

There are several prerequisites to the cognitive walkthrough procedure which include:

1. A general description of who the users will be and what relevant knowledge they possess,
2. A specific description of one or more representative tasks to be performed with the system

978 -1-4244-1651-6/08/$25.00 © 2008 IEEE

3. A list of the correct actions required to complete each of these tasks with the interface being evaluated.

When using the CW method, four questions are asked for each correct action involved in a task (Wharton, Rieman, Lewis, and Polson 1994):

1. Will the user try to achieve the right effect?
2. Will the user notice that the correct action is available?
3. Will the user associate the correct action with the desired effect?
4. If the correct action is performed, will the user see that progress is being made towards the solution of the task?

If there are positive answers to all four questions, this is termed a "success story" for the specified action. If there is a negative answer to any of the four questions, this is termed a "failure story" for the action.

The CW procedure has been used to evaluate the usability issues for many practical applications. For instance, John and Packer (1995) reported a case study in which "the Cognitive Walkthrough evaluation technique was learnable and usable for a computer designer with little psychological or HCI training.

3. Prototyping the dynamic menu

One avenue for developers to reduce the cost of evaluation and simultaneously make it more helpful in workplace settings is through program instrumentation (Rieman, Franzke, and Redmiles 1995). This paper developed an approach using intelligent agents which can adapt the cognitive walkthrough procedure to support the evaluation of prototypes with real end users in their work places. This agent system can monitor users working with the prototype and report mismatches between developers' expectations and a system's actual usage.

The agent basically involves an automatic logging system that has the computer collect statistics about the detailed use of the menu interface. The intelligence can be collected through the transaction log, which contains records of users' actions and the system response to them, or on the full-screen log, which contains a complete session transcript for detailed analysis. This agent program can contain statistics about the frequency with which each user has used each feature in the program, and the frequency with which various events of interest have occurred (Ivory and Hearst 2001). Statistics showing the frequency of use of commands and other system features were then used to optimize frequently used features and to determine what menu elements should appear in the dynamic menu for the next task. In addition, an analysis on patterns of use was made using the logging data. Statistics showing the frequency of various events, such as error situations and the use of online help, were used to improve the usability of future releases of the system. Figure 1 is a logging of time interval which user spends on searching menu item. The first column is user name. The second column records starting time. The forth column is ending time and the last column is time interval.

This paper decided to implement the dynamic menu interface concept in Management Information System (MIS) software because they involves much more complicated use of menu system. For instance, Microsoft Office Word has more than 100 items in menu and users have to spend significant amounts of time learning it and having to memorize where each menu item is located. The dynamic menu tested in this paper is based on a POS system (Point of Sale). Its main functions include Stock Control, Purchase, Sale, Report, Product Information, Salary Management, System Management and etc. The structure of the menu is organized according to its functions. Quantity of total items of the system is 65 items and first level items are 12.

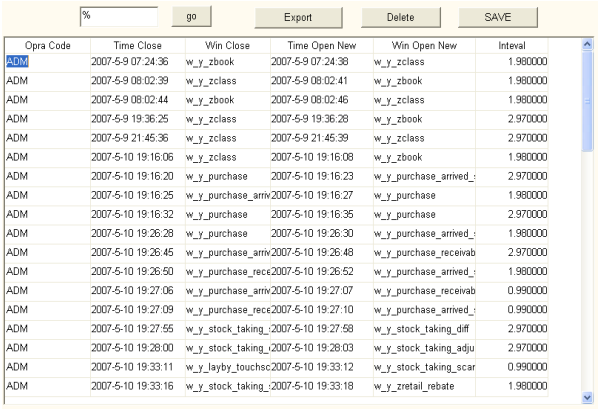

Figure 1. Time log of interval which user spends on searching menu item.

The type of MIS system is quite different from website with regard to the access control. Websites are designed to be accessed by public, while MIS is designed to accessed limited users that are usually staff within an enterprise. The dynamic menu is specifically devised to address several human-computer interaction principles that are, namely Tolerance, Simplicity, Visibility, Affordance, Consistency, Structure and Feedback (Sheiderman 1998).

The dynamic menu is devised as subsets of main menu (see Figure 2). It is changeable according to different users and their status. The most possibly used items are listed in the container of dynamic menu. Therefore, users do not need to search items hierarchically rooted from main menu. The dynamic menu has the following major attributes:

- Limited items: dynamic menu should not include too many items, no more than 30 items is best.

- Personalized items: content of dynamic menu is different with different users, etc, different user has different dynamical menu. Dynamic menu is based on the fact that personal job is repeatable. Only small part of main menu is frequently used. For example, managers usually focus on reports; staffs are more interested in transactions.
- Related context: dynamic menu records sequence of opened window. At which situation, the window is open, what next window user probably opened.

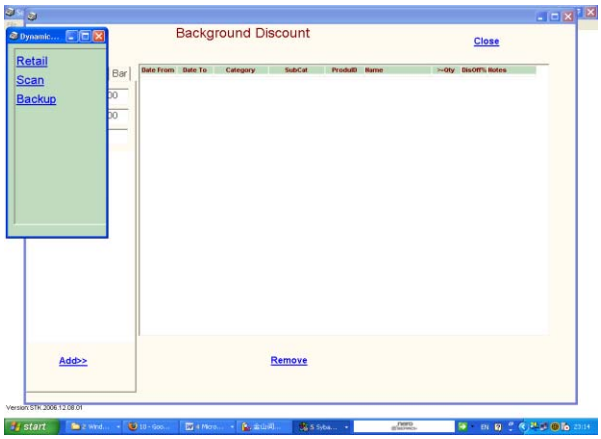

Figure 2. The screenshot of a simple dynamic menu

4. Evaluation methodologies

The HCI analysis methodology used in this study is a combination of the following two. Both the methods were used to collect different types of data to reflect different aspects:

- Questionnaires: this survey method can provide feedback from the point of view of the end users. It provides information regarding users' preferences and ideas about the design in many stages of the dynamic interface development. Users' reactions can have a strong impact on the design and development of an interface.
- Logging & Automated Metrics (Ivory and Hearst 2001): It involves having the computer collect statistics about the detailed use of the menu interface. The data can be collected through the transaction log, which contains records of users' actions and the system response to them, or on the full-screen log, which contains a complete session transcript for detailed analysis. This technique requires data retrieval from the logs, which were done by the intelligent agent designed to extract the data or by built-in report facilities. The next step in evaluating the data is the actual analysis, which was done by statistical software packages, report generators or playbacks of the recorded data.

5. Pilot study of evaluating the dynamic menu

This section presented a preliminary pilot study of evaluating the usability and effectiveness of the dynamic menu interface. The prototyped dynamical menu was evaluated by the two methods: questionnaires and automatic logging analysis. The method of questionnaires is to find out general view on dynamic menu. The aim of the second method is to find out time users spent on menu item search, from which the difference between normal menu and dynamic menu can be figured out.

5.1 Controlled Experiment

In order to make comparison between normal menu and dynamic menu, a group of tasks were set up for a controlled experiment. In the designed scenario, these tasks were executed in the order as follows.

1. Create Category
2. Add items
3. Purchase
4. Check payable
5. Check Stock
6. Sale Item
7. Check stock again
8. Generate Daily Report

Five subjects were invited into the evaluation as the end users of the software and menu interface. They have different background and work experiment. Their details are listed in the Table 1.

Full training and practice were provided to every participant. First of all, they were introduced with the main function of the software, categories, purchase, retail, stock, analysis; the main menu layout, and how to use the menu. Then they were given some time to practice and familiar with the software. After the training session, they were asked to complete the task list by using both with the normal menu and dynamic menu.

Table 1. Experience and background information of invited subjects

	Gender	Computer Background	Retail Experience
A	M	Yes	No
B	F	No	No
C	F	Yes	No
D	F	Yes	Yes
E	F	No	No

5.2 Automatic Logging/Metrics

Starting time and ending time of every task was recorded by the intelligent agent through the built-in automatic logging system. The time on searching menu can therefore be calculated to compare how long the two groups spent in searching the two menus respectively.

In addition, error rate by choosing wrong menu items were also calculated by finishing time of tasks. For instance, if the task is finished within 3 seconds, it is clear that participants did noting on the task, therefore, it can be treated as choosing wrong menu item.

5.3 Questionnaires

The following questions were asked to participants as follows:

1. Is the dynamic menu usable for you?
2. Which menu do you use first: dynamic menu or normal menu?
3. Do you think the dynamic menu help you finish work quickly?
4. On which side do you prefer to put right side, left side, top or bottom?
5. Normal menu has more than 100 items, do you easily find out a menu item from normal menu?
6. Dynamic menu can predicate what you should do next; do you think it is a good job reminder?

6. Data analysis and discussions

After the initial task logging analysis and questionnaires, some advantages and disadvantages of dynamic menu and major results from questionnaires were summarized as follows:

The dynamic menu reduced the average time for each menu search. In the pilot study, the average time for normal menu search is 5.6 seconds per task, while the average time for dynamic menu is 2.7 seconds per task. Therefore, dynamic menu is much faster than normal menu. The time for menu search includes cursor movement, mouse click, thinking (menu search = cursor movement distance + mouse click + thinking). Average distance between dynamic menu and mouse is equal to half screen, while average distance between normal menu and mouse is equal to half screen plus going down next level of menu. Quantity of mouse click for dynamic menu is one, while quantity of mouse click for normal menu is at lest two. Dynamic menu give hints to users what they should do next, while normal menu does not. Those are the reasons why the dynamic menu is faster to use than the normal menu.

The dynamic menu can become a reminder for the task that needs to be done immediately as next step. Without dynamic menu, there are two things users have to memorize, one is what they should do next, and the other is where its menu item is located. From the perspective of attention and working memory, these will distract users from their current task. However, the dynamic menu helps them focus on their job instead. The dynamic menu collects information such as who the users are, what they have done, therefore, it can predict what user will do next based on the system's automatic record, retrieval, reasoning, and prediction of their previous experience. In other word, it always tells users what they should do next.

The questionnaire results also show that the participant with less computer knowledge prefers to the use of the dynamic menu. The survey shows the less computer participant give higher marks to the dynamic menu, because they need less special training and less memory.

Even though most participants think dynamic menu has usable, dynamic menu also bring them inconveniences. The dynamic menu always shows on top of other window, so working window view may be blocked. Thus, participants have to move the dynamic menu sometimes.

7. Conclusions

This paper developed a method for automatically generating dynamic menu to reflect different needs from different types of end users, based on the Cognitive Walkthrough procedure and automatic instrumentation. This paper also presented the preliminary results from a pilot study performed to evaluate the effectiveness of the dynamic menu. Preliminary results showed that the dynamic menu interface (2.7 seconds) reduced the time for menu search per task significantly than the normal menu interface (5.6 seconds). Therefore, the total time for completing a series of devised tasks was also significantly reduced by using the dynamic menu. Future work includes more systematic evaluation of the dynamic menu with more representative tasks and more questionnaires included.

References

[1] Ivory, M.Y., and Hearst, M.A. (2001): The State of the Art in Automating Usability Evaluation of User Interfaces, *ACM Computing Surveys,* 33(4): 470-516.

[2] John, B. and Packer, H. (1995): Learning and Using the Cognitive Walkthrough Method: A Case Study Approach. *Proceedings of ACM CHI '95*, Denver, CO, May, 429-436.

[3] Lewis, C., and Rieman, J. (1993): Task-Centered User Interface Design Q A Practical Introduction. url: ftp.cs.colorado.edu

[4] Polson, P.G., Lewis, C., Rieman, J., and Wharton, C. (1992): Cognitive walkthroughs: A method for theory-based evaluation of user interfaces. *International Journal of Man-Machine Studies* 36, 741-773.

[5] Rieman, J., Franzke, M., and Redmiles, D. (1995): Usability Evaluation with the Cognitive Walkthrough, *CHI'95 tutorials,* url: http://acm.org/sigchi/chi95/proceedings/tutors/jr_bdy.htm

[6] Rieman, J. (1996): A field study of exploratory learning strategies, *ACM Transactions on Computer-Human Interaction (TOCHI)* Volume 3, Issue 3 (September 1996) Pages: 189 - 218 ACM Press.

[7] Sheiderman, B. (1998): *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. 3rd ed. Addison-Wesley, Reading, MA.

[8] Wharton, C., Rieman, J., Lewis, C., and Polson, P. (1994): *The Cognitive Walkthrough Method: A Practitioner's Guide. In Usability Inspection Methods*, J. Nielsen and R.L. Mack (Eds.), New York: John Wiley & Sons, pp.105-141.