

Proceedings of the Human Factors and Ergonomics Society Annual Meeting

<http://pro.sagepub.com/>

A Case Study of Transparent User Interfaces in a Commercial 3-D Modeling and Paint Application

Beverly L. Harrison and Kim J. Vicente

Proceedings of the Human Factors and Ergonomics Society Annual Meeting 1996 40: 375

DOI: 10.1177/154193129604000612

The online version of this article can be found at:

<http://pro.sagepub.com/content/40/6/375>

Published by:



<http://www.sagepublications.com>

On behalf of:



[Human Factors and Ergonomics Society](http://www.hfes.org)

Additional services and information for *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* can be found at:

Email Alerts: <http://pro.sagepub.com/cgi/alerts>

Subscriptions: <http://pro.sagepub.com/subscriptions>

Reprints: <http://www.sagepub.com/journalsReprints.nav>

Permissions: <http://www.sagepub.com/journalsPermissions.nav>

Citations: <http://pro.sagepub.com/content/40/6/375.refs.html>

>> [Version of Record](#) - Oct 1, 1996

[What is This?](#)

A CASE STUDY OF TRANSPARENT USER INTERFACES IN A COMMERCIAL 3-D MODELING AND PAINT APPLICATION

Beverly L. Harrison^{1,2}

Kim J. Vicente²

¹ Alias|Wavefront

² Cognitive Engineering Laboratory

110 Richmond Street East
Toronto, Ontario,
Canada
M5C 1P1

Dept. of Mechanical & Industrial Engineering,
5 King's College Road,
University of Toronto,
Toronto, Ontario, Canada
M5S 3G8

This paper describes a case study of transparent user interface tools in a commercial 3-D modeling and paint application. It represents another step in an ongoing research program evaluating transparent human-computer interfaces. Results from previous controlled experiments were used to inform our design choices in the working product. We collected data from 11 users one month after they received the application, using a semi-structured interview. Working sessions were also video taped and analyzed. Based on these results, modifications were made and a follow-up interview was conducted three weeks later. A number of transparency-related issues were identified. This case study illustrates the value and challenges that one encounters in transitioning from basic research to commercial applications.

INTRODUCTION

This paper describes a case study of transparent user interface tools in a commercial 3-D modeling and paint application. It represents another step in an ongoing research program evaluating transparent human-computer interfaces.

The interaction between limited screen real estate, graphical user interfaces, and human attention limitations results in a visual attention problem, towards which this research is aimed. Many HCI applications are designed with a work space or data area which is the primary focus of attention. The limited display size means that the tools to manipulate the data appear in windows, menus, dialogue boxes, and palettes over top of the work area, thereby diverting or blocking users' attention from the underlying data area. This area often provides feedback about the actions users apply (e.g., in drawing systems there are traditional tools to change paint brushes and colors), meaning users frequently have to move interface objects in and out of the way to complete domain tasks. As a result, users spend a great deal of time performing interface management tasks rather than productive work.

Transparency offers a potential remedy to this problem. If interface objects were semi-transparent, users might be able to allocate their attention either to the top layer of information (interface objects and tools) or to the bottom layer (data area) at will, since the latter could be visible under the former. As a result, the fluency of work should be improved since there is less need for interface management tasks.

There are a few examples of interfaces which exemplify this seamless task-tool integration through the application of transparency (Bier, Stone, Pier, Buxton, & DeRose, 1993; Bier, Stone, Fishkin, Buxton, & Baudel, 1994; Ishii & Kobayashi, 1991; Ishii, 1990; Knowlton, 1977; Zhai, Buxton,

& Milgram, 1994; Schmandt, 1983). However, most of these have not been the subject of empirical evaluation, and none have found their way into a commercially available product. The research program represented here addresses this gap in the literature by conducting a systematic series of experiments evaluating the impact of transparency on human performance, culminating in a case study of transparency in a commercial 3-D modeling and paint application that is the focus of the present paper.

PROGRESSION OF RESEARCH

The implementation of transparency in the commercial application observed here was preceded by several controlled experiments. The first set of experiments (Harrison, Ishii, Vicente, & Buxton, 1995) investigated the impact of transparency on interference between visual layers using a Stroop paradigm (Figure 1). Although this is not a very representative task, the study was well controlled and provided an initial conservative estimate of the level of transparency that is required to minimize interference between foreground and background layers. The subsequent set of studies (Harrison, Kurtenbach, & Vicente, 1995; Harrison & Vicente, 1996) sacrificed some experimental control but used a more representative set of stimuli to investigate the impact of transparency under more realistic conditions (Figures 2 and 3). The next step in the research program, described here, was to conduct a case study of transparency in a commercial application under even less controlled but more representative conditions.

This case study was conducted jointly with a company using their application in 3-D modeling and painting, "MagicPaint" (Product name is disguised for confidentiality.) An evaluation of existing "MagicPaint" users was conducted using this modified software.

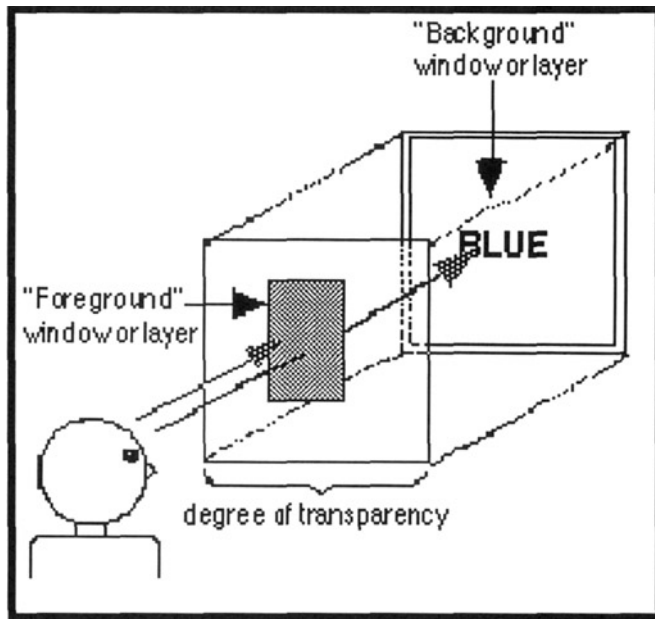


Figure 1. Model of Stroop Experiment Configuration. Same approach was used for all experiments, substituting image content.

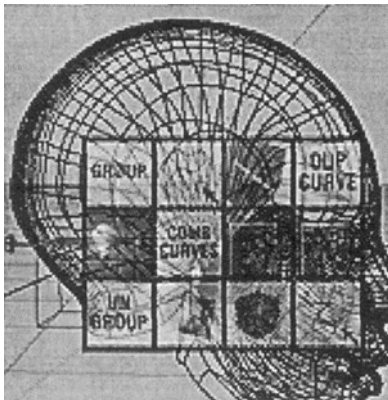


Figure 2. Sample Image from Icon Palette Selection Experiment. 50% transparent, wire frame background. Indep. variables: transparency level, icon type, background type. Dependent variables: selection response time and error rate.

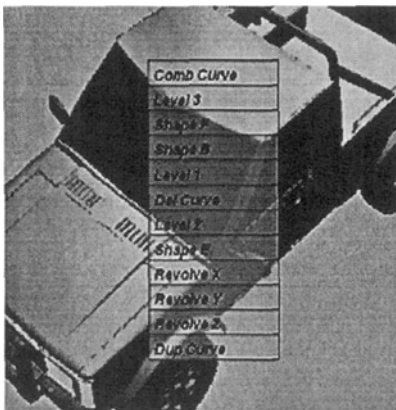


Figure 3. Sample Image from Text Menu Selection Experiment. 50% transparent, regular Motif style menu, solid background. User task - select a target icon from the palette.

RELATED WORK

Many applications are designed with a large work space or data area which is the primary focus of attention, while the tools to manipulate the data appear in windows and palettes over top of the work area. To simplify our conceptual model, we have classified the work space or data area as the *background layer* (a single large window typically) and the UI tools (menus, dialogs, palettes) as the *foreground layer* (multiple objects or windows). This reflects the visual representation of many task contexts, where the "work" is displayed in a full-screen window, thereby requiring visually overlapping UI tools. These tools divert or block our attention from our work, which is often providing feedback about the actions we apply, for example, painting or drawing systems and the traditional opaque UI tools to change paint brushes and colors. However, there are several examples of highly-advanced systems which exemplify more seamless task-tool integration through the application of transparent user interfaces.

In Heads Up Display (HUD) design, aircraft instrumentation (a graphical computer interface) is superimposed on the external real world scene, using specially engineered windshields (e.g., Larish and Wickens, 1991; Wickens, Martin-Emerson, and Larish, 1993). In the ClearBoard work (Ishii and Kobayashi, 1991), a large drawing surface, resembling an electronic drafting table, is overlaid on a life sized video image of the user's collaborative partner. The TeamWorkStation system (Ishii, 1990), predecessor to ClearBoard, created semi-transparent computer work space windows superimposed with video image windows (e.g., a person, an object being discussed). The ToolGlass and MagicLens work (Bier et al., 1993; Bier et al., 1994) reflects a tight coupling between tool function, target object specification, and transparency. These projects used clear or see-through palettes which could be aligned with underlying objects. Tools were invoked by clicking "through" the desired function, using alignment to specific the target object for the function. Other designs include such things as video overlays like those used in presenting sports scores in broadcast television.

Some designs combine transparency and 3-D projected views of the user interface. Several examples are: the work on "3-D silk (volume) cursors" (Zhai, Buxton and Milgram, 1994), the work by Knowlton (Knowlton, 1977), which used graphical overlays projected down onto half-silvered mirrors over blank keyboard keys to dynamically re-label buttons and functions keys (e.g., for telephone operators), and the work by Schmandt (1983), who built a system to allow users to manually manipulate and interact with objects in a 3-D computer space using a 3-D wand. Again a half-silvered mirror was used to project the computer space over the user's hand(s) and the input device. Disney has also developed a product called the "ImaginEasel" for animators and artists. ImaginEasel keeps the user's hand and input device in the workspace (using mirrors). In every case transparency seems to provide a more seamless integration between the data or work and the UI tools.

METHOD

The case study was conducted in two stages. Approximately one month after the first release of the transparency version of MagicPaint, we interviewed 11 users. Several modifications were made based on these comments. We then waited an additional three weeks and again interviewed these same users.

A semi-structured interview was used. All users were asked the same questions about their usage of MagicPaint to obtain a "user profile". They were also all asked to list the three aspects of the interface they liked the most and disliked the most. Finally, they were asked for their opinions about transparency in particular and any additional comments or observations. All responses were recorded. Those aspects of the interview relating to transparency are summarized below.

We intermittently video taped work sessions from randomly selected users at random times before and after the transparent user interfaces were introduced. These video tapes were analyzed to augment our interview data on work patterns or changes in work patterns. The results of the video tapes analyses are integrated into the results reported below.

User Profiles

The case study was conducted with product users who worked for the company that developed MagicPaint. We collected user profile information to categorize both levels of expertise and how extensively users had worked with the new transparent interface. Users were grouped into three categories based on these profiles (Table 1).

The first category was comprised of four users who worked primarily as programmers or product testers. Their use of MagicPaint was characterized by short "bursty" paint sessions used to produce scribble style drawings. A large combination of features and painting tools were used, with each being used only a few times. Their task goal was to ensure that the system worked at a basic level and that there were no apparent software bugs.

The second user group consisted of three "demo gurus". These users were primarily marketing and sales staff who were expert MagicPaint users. Their task goal is to produce fast, interesting, and flashy demonstrations to potential customers and at trade shows. These users typically import almost completed, high quality artwork and demonstrate key features through fast, dynamic touch ups to the image.

The last user category is the group comprised of four hired artists who use MagicPaint to produce images. These highly skilled users most accurately reflect the task goals and

Type of User	# of Users	# who used transparency at beginning of trial	reason for NOT using transparency
programmers, testers	4	4	
product "demo" experts	3	1	2 users did not know how to turn feature on
professional artists	4	2	2 users turned feature off after one week of usage

Table 1. User profiles for subjects in case study.

usage patterns of MagicPaint customers. Their task goal is to produce high quality images. They accomplish this through lengthy sessions which span days, weeks and sometimes months, depending upon the image complexity. They heavily customize the system and the defaults to suit their particular task and work style.

RESULTS

Both prior to and after the introduction of transparency the users maintained similar screen layouts. This consisted of a drawing area which was a full-screen window, with menu items and icon palettes minimized in size and pushed to either the top or the side of the drawing window (Figure 4). Interactive windows, overlapping the drawing area, were used to change tool attributes (Figure 5). According to the users, and in particular the artists who were extensive expert users, "no screen size is big enough" and "anything which minimized the number and size of overlapping windows is good".

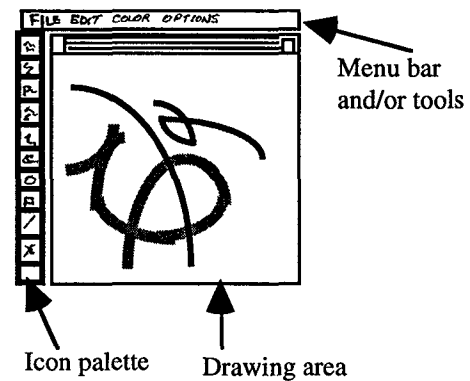


Figure 4. Diagram of typical screen configuration and layout of windows. Reflects task during drawing or painting, typical of a right handed user.

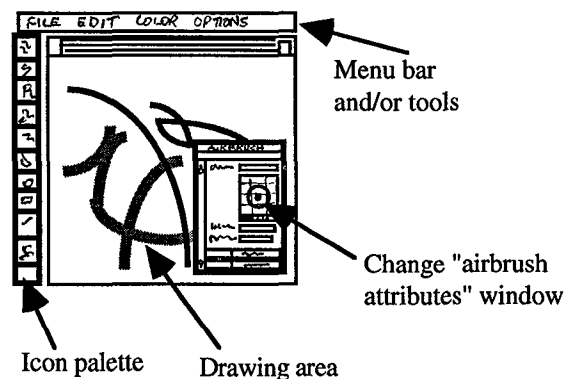
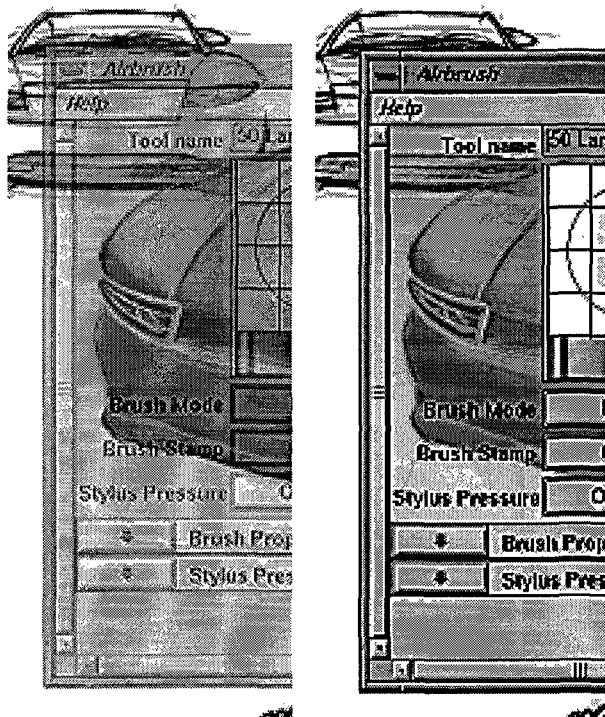


Figure 5. Diagram of typical screen configuration and layout with overlapping tool modifier window.

Based in part on the results of our previous research (Harrison et al., 1995, and Harrison and Vicente, 1996), we decided to create UI windows which were a hybrid of opaque and transparent components. The guideline we applied was that any selectable item within the window would be opaque, while non-selectable areas of the window



6 (a). labels & surface around labels are both transparent
6 (b). labels are opaque, surface around labels is transparent

Figure 6. Actual Images from MagicPaint showing design alternatives for transparency.

would be 50% transparent (see Figure 6). The window size and location were determined by the user (as in any normal windowing system). Users generally maintained the same window sizes throughout the trial. All UI windows were semi-transparent, with varying numbers and types of opaque components, depending upon the specific window properties. These opaque components were typically buttons, sliders, color wheels, and radius/opacity selectors. Whenever the user selected a particular tool item from the menu or from the icon palette the appropriate UI window would appear and remain until the user elected to explicitly close it.

In the first interview phase, we discovered that about half of the users interviewed had not been using transparency (see summary in Table 1). The people who did use it were: all four programmer/testers, only one "demo guru", and two artists. Two users had not yet received information about how to turn the feature on! Two users had turned transparency off after about one week of use. Specific comments and reasons are described below.

Issues Related to Technical Limitations

There were complaints that the font within transparency windows was difficult to read, particularly whenever it was over dark images. This was attributed to using black fonts primarily combined with a dark gray or black stippled window color. The result was distortion to the font edges. This was later fixed by changing the default window color to light gray, thereby avoiding conflicts with any of the font colors.

Another common complaint was about "flashing windows". Here users were referring to an unfortunate artifact

produced by drawing windows in the overlay plane. In particular, any window in the overlay plane can be assigned its own color map (roughly, a list of colors that window uses). Whenever the cursor moves over a particular window, that window and its color map become active. If the color maps are different between windows, different color maps are swapped in and out. This generates a flashing of the window borders and title bar. Although it may sound subtle, to artists this was so annoying that one turned off transparency altogether. This problem can be partially addressed by assigning a common color map. However, some slight but noticeable highlighting changes will still occur which are meant to indicate the currently active window. The complete solution to this problem is unclear at this time.

Users requested the ability to set their own level of transparency by using something like a slider. Unfortunately, due to technical limitations this is not feasible at this time. Some users also requested the ability to set their own window color. We have not decided whether to provide this capability as yet, since the choice of window color may interact with transparency to negatively affect performance.

Users commented that some windows have very little transparent area and many opaque objects, limiting the utility of transparency. In the current design of the product, there are a number of objects within a window which we left opaque. These were selectable objects (buttons, sliders) which could be entered from the keyboard (text entry fields), and selectable areas, such as the color wheel. In particular, there are a number of embedded diagrams, such as the color wheel or the radius diagram for adjusting brush size, where precise selection and accurate visual perception are critical. In these cases, the objects were left opaque (see Figure 6b). The area around these objects was transparent. Windows which contain many such objects are primarily opaque as a result. We did implement a version of the system where both the button and object borders were transparent (see Figure 6a). While this made more of the window transparent, it was not obvious that the "buttons" and text fields were selectable to users. This was partly because users often equate "grayed-out" or partially invisible items with "un-selectable at this time".

Issues Related to Work flow

In general, transparency was felt to be most useful when windows were poorly placed or there were too many open. The number of windows open at any one time and their locations are completely determined by the user. A typical session has anywhere from one to four windows open simultaneously (or a minimum of 10% of the screen obscured). One artist commented that with experience more than one open window should not happen often. Ideally, almost all the artists commented that *any* window was in their way. They wanted a system where no windows were required. Transparency was seen to be one step in that general direction.

Most users commented that transparency works best in situations where they wanted visual feedback about some operation, as opposed to situations where they were actually trying to draw or paint directly behind the window.

Several users wanted transparency in the "support" windows. This would include the Help System windows and any message dialogs presented to the user. The message

dialogs are generally system related issues. Serious errors would likely need to remain opaque to ensure that they were clearly seen and also because often with such errors it is important to interrupt the user's work. We are currently working on making the Help system transparent.

EXTENSIONS AND IMPLICATIONS

The exercise of implementing transparency in a product revealed many more technical challenges than we had anticipated. We made a series of compromises in the design parameters to adjust for this. A number of operating system level bugs were discovered and various temporary work-arounds had to be created. Finally, we discovered that our implementation approach is greatly impacted by the hardware configurations of specific machines. In order to successfully run transparent interfaces, we found it necessary to include procedures that determine the type of machine running the program and then adjust for its capabilities (or lack thereof). However, we believe that this reasonably reflects the complexity of creating new technologies and integrating these into commercial products for trials. Many useful insights can be attained and many creative solutions can emerge in such a situation.

A number of the users' comments resulted directly from the technical issues. It is clear that even seemingly minor issues to a programmer can be highly irritating to someone with an artist's eye. These can influence the user to the point of shutting off features! It is interesting to note that with transparency, it is sometimes difficult to "hide" the technical and implementation aspects from the user (as good UI practice might suggest). Again, this provided us with a number of valuable insights into our user group and we were able to address many of the apparently aesthetic yet critical issues.

The case study confirmed our earlier experimental findings that default settings have an enormous impact of the usability of transparency. Window background color, font color and type style, and transparency level itself interact and make the difference between a system that works and one that does not. Relatively minor adjustments result in major perceptual differences. We determined that 50% transparent windows in lighter colors seemed to work best.

We were not able to assess whether transparency has a significant impact on improving the user's overall productivity, as intended. Since we implemented transparency into a new product release, it is very difficult to separate out which aspects of work are influenced by the introduction of transparency and which are influenced by some other changes to the system or its features. The main changes introduced (unrelated to transparency) were: new icon palettes and a complete reorganization of menu item ordering, hierarchic menus, user definable and save-able "toolkits" (collections of tools in a sequence), and user constructable and save-able pop-up menus.

In summary, this case study illustrates the challenges that one encounters in transitioning from basic research to commercial applications. We believe that transparent interfaces hold great promise, but it is equally clear that, for now, technical implementation details stand in the way of realizing those benefits. Therefore, further technical innovation is required before we can put the insights gained from our previous experiments (Harrison, Ishii, et al., 1995;

Harrison, Kurtenbach, and Vicente, 1995; Harrison and Vicente, 1996) into practice. It is essential that these obstacles be overcome if human factors research is to have an impact on actual design practice.

ACKNOWLEDGMENTS

This research was sponsored by research grants from Alias|Wavefront and the Natural Sciences and Engineering Research Council of Canada. We would also like to thank Bill Buxton, Hiroshi Ishii, Gordon Kurtenbach, and members of the Cognitive Engineering Lab for their comments.

REFERENCES

- Bier, E. A., Stone, M. C., Pier, K., Buxton, W., & DeRose, T. D. (1993). Toolglass and magic lenses: The see-through interface. In *Proceedings of SIGGRAPH'93*. (pp. 73-80). New York: ACM.
- Bier, E. A., Stone, M. C., Fishkin, K., Buxton, W., & Baudel, T. (1994). A taxonomy of see-through tools. In *Proceedings of CHI'94*. (pp. 358-364). New York: ACM.
- Harrison, B. L., Ishii, H., Vicente, K. J., & Buxton, B. (1995). Transparent layered user interfaces: An evaluation of a display design space to enhance focused and divided attention. In *Proceedings of CHI'95*. (pp. 317-324). New York: ACM.
- Harrison, B. L., Kurtenbach, G., & Vicente, K. J. (1995). An experimental evaluation of transparent user interface tools and information content. In *Proceedings of User Interface Software and Technologies - UIST'95* (pp. 81-90). New York: ACM.
- Harrison, B. L., & Vicente, K. J. (1996). An experimental evaluation of transparency menu usage. In *Proceedings of CHI'96*. (pp. 391-396). New York: ACM.
- Ishii, H., & Kobayashi, M. (1991). Clearboard: A seamless medium for shared drawing and conversation with eye contact. In *Proceedings of CHI'91*. (pp. 525-532). New York: ACM.
- Ishii, H. (1990). TeamWorkStation: Toward a seamless shared workspace. In *Proceedings of Computer Supported Collaborative Work - CSCW'90*. New York: ACM.
- Knowlton, K. C. (1977). Computer displays optically superimposed on input devices. *Bell System Technical Journal*, *56*, (pp. 367-383).
- Larish, I. and Wickens, C. D. (1991). Divided Attention with Superimposed and Separated Imagery: Implications for Head-Up Displays. *University of Illinois Institute of Aviation Technical Report* (ARL-91-4/NASA HUD-91-1).
- Schmandt, C. (1983). Spatial input/display correspondence in a stereoscopic computer graphic workstation. *Computer Graphics*, *17*, (pp. 253-259).
- Wickens, C. D., Martin-Emerson, R., and Larish, I. (1993). Attentional tunneling and the Head-up Display. In *Proceedings of the 7th Annual Symposium on Aviation Psychology, Ohio State University, Ohio*, (pp. 865-870).
- Zhai, S., Buxton, W., & Milgram, P. (1994). The "silk cursor": Investigating transparency for 3D target acquisition. In *Proceedings of CHI'94*. (pp. 459-464). New York: ACM.