MIDDesktop: An Application Framework for Single Display Groupware Investigations.

Garth B.D. Shoemaker and Kori M. Inkpen
EDGE Lab
School of Computing Science
Simon Fraser University
Burnaby, BC V5A 1S6, Canada
+1 604 268 6605
garths@cs.sfu.ca, inkpen@cs.sfu.ca

ABSTRACT
It has become apparent that the design of computers needlessly imposes a limitation of one user per machine. Research in the area of Single Display Groupware (SDG) explores how to best support simultaneous interactions of multiple users with a single, shared display [7]. This paper presents a framework that allows for SDG investigations without requiring the development of custom SDG software applications. This is an important development because the software required to investigate SDG related topics is often complicated and time consuming to develop.

Keywords
Single Display Groupware (SDG), multiple input devices, multiple mice, computer supported cooperative work (CSCW), synchronous co-located collaboration.

INTRODUCTION
People in a collaborative environment, such as a place of business or a school, often want or need to work together on a shared project. In the physical world this is easily accomplished by gathering around and interacting with physical objects. In the computer realm this is not so easily done. Computer interfaces have been designed to allow only one user to interact with a single computer at a time.

Single Display Groupware (SDG) research involves the investigation of how to support multiple people interacting simultaneously with a shared computer display [7]. Having multiple users working together around a shared display is fundamentally different than having a single user work alone. Many interface elements, such as menu bars and toolbars, that are effective for single user applications must be re-evaluated for this new interaction paradigm. This re-evaluation is necessary so that the benefits provided by this style of collaboration can be fully realized.

An obvious first step is to provide each user with control of a separate input device. Several researchers in the area of SDG have previously developed custom applications to provide support for multiple independent input devices, each controlling an independent cursor [1,2,4,5,7]. However, development of custom applications is a time-consuming process and there is currently a lack of available SDG software with which to perform research in this area. Hourcade and Bederson [3] have recently developed the MID toolkit to facilitate this process, but a significant amount of programming is still required.

This paper presents an application framework that allows researchers to explore SDG applications without requiring the development of custom software. This framework will allow researchers from a variety of disciplines to explore research issues surrounding support for synchronous co-located collaboration.

MIDDESKTOP
We have developed MIDDesktop, an application that allows commonly available single user Java JApplets to work as multiple-mouse aware SDG applications [6].

The Interface
As shown in Figure 1, MIDDesktop offers a typical desktop interface. In functionality, it provides support for multiple mice by drawing a separate independent cursor for each mouse plugged into the computer[1]. MIDDesktop is a Java application that provides a framework within which Java Swing Applets (JApplets) can be executed. The Swing Applets need not be designed or programmed as multiple mouse-aware SDG applications. Indeed this would be difficult, as there is no inherent support for multiple mice in Java. The Swing Applets simply execute in sub-windows within the MIDDesktop display area. The MIDDesktop framework handles input from multiple mice and filters the mouse events through to the appropriate Swing Applets.

When MIDDesktop is executed, it displays a blank desktop with icons representing all active drives (hard drives, network drives, removable media). Users can double-click on these icons and browse the files on that directory. Double-clicking an HTML file causes MIDDesktop to parse the file, identifying any JApplet tags[2]. These tags contain an

[1] Current support is for Windows '98 with Universal Serial Bus (USB) mice only.

[2] These are the same kinds of JApplet tags that appear in a normal web-page.

address of a JApplet, as well as parameters dictating how the JApplet is to be executed. The JApplet is opened in a new window within MIDDesktop. Users can then continue to browse the file hierarchy, open new JApplets or interact with already open JApplets. Each user can independently or cooperatively interact with any component within the MIDDesktop display area.

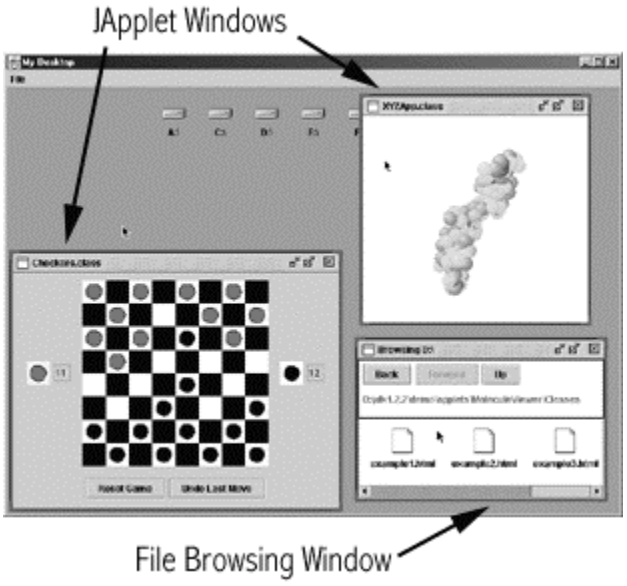

Figure 1. Screenshot of MIDDesktop with 3 mice.

The Implementation

MIDDesktop is implemented as an application in Java 2.0. It makes extensive use of MIDFrame, a Java class library that allows for the implementation of Swing based multiple-mouse aware SDG applications [6]. MIDFrame, in turn, was developed using MID [3], a Java class library that supports the development of Java applications with multiple input devices of arbitrary type.

LEVERAGING SINGLE USER SOFTWARE

There is a substantial amount of public software available; however, most of it was designed and implemented for single users. While multiple users could interact with the software, they would be forced into a turn-taking protocol, having to share the mouse and keyboard.

The MIDDesktop application framework allows researchers to take advantage of this already-existing pool of Java JApplet software applications. Not only can the researchers use these applications as multiple mouse aware SDG applications, but they can also use them as originally intended, in their single user modes. In this manner researchers can compare the applications as both SDG applications, and as single user applications.

FUTURE WORK

We have begun using the MIDDesktop application framework for longitudinal studies in the area of SDG. This will allow us to explore users interacting with a variety of SDG applications over an extended period of time, uncovering important, but sometimes subtle, issues related to synchronous co-located collaboration. Future work will also investigate usability issues related to MIDDesktop as well as technical improvements, including increased performance, broadened platform compatibility, and the addition of standard desktop interface features. It is also expected that the ability to browse remotely hosted JApplets (i.e. on the Internet) will be added.

ACKNOWLEDGMENTS

This project was funded by NSERC, BC Advanced Systems Institute and Electronic Arts Canada. We would like to thank Dr. Ben Bederson and Juan Pablo Hourcade for their help with the MID Java class. Thanks also to members of the EDGE Lab for their assistance.

REFERENCES

1. Bier, E.A., and Freeman, S. MMM: A user interface architecture for shared editors on a single screen. *Proceedings of UIST '91*. 79-86.
2. Bricker, L.J., Tanimoto, S.L., Rothenberg, A.I., Hutama, D.C., and Wong, T.H. Multiplayer Activities that Develop Mathematical Coordination. *Proceedings of CSCL '95*. 32-39.
3. Hourcade, J.P., and Bederson, B.B. Architecture and Implementation of a Java Package for Multiple Input Devices (MID), Technical Report CS-TR-4018, UMIACS-TR-99-26.
4. Inkpen, K.M., Ho-Ching, W., Kuederle, O., Scott, S.D. and Shoemaker, G.B.D. "This is fun! We're all best friends and we're all playing.": Supporting children's synchronous collaboration. *Proceedings of CSCL '99*.
5. Myers, B.A., Stiel, H., and Gargiulo, R. Collaboration Using Multiple PDAs Connected to a PC, in Proceedings of CSCW '98, ACM Press, 285-294.
6. Shoemaker, G.B.D. MIDDesktop & MIDFrame. At: http://www.sfu.ca/~garths/work/gmid.html
7. Stewart, J., Bederson, B.B., Druin, A. Single Display Groupware: A Model for Co-present Collaboration, in *Proceedings of CHI '99*, ACM Press, 286-293.