SDGToolkit: A Toolkit for Rapidly Prototyping Single Display Groupware

Edward Tse and Saul Greenberg
Department of Computer Science
University of Calgary
Calgary, Alberta, Canada T2N 1N4
[tsee, saul]@cpsc.ucalgary.ca

ABSTRACT
Single Display Groupware (SDG) is a field of study that explores how multiple users share a common display. The problem is that it is hard to develop SDG software, for operating systems offer little support (and considerable hurdles) for developing software that manages simultaneous use of multiple input devices, such as multiple mice and keyboards. Yet serious research in SDG demands that we have the ability to rapidly prototype our ideas. In this poster, we present SDG Toolkit, a toolkit for rapidly prototyping single display groupware. We describe its features, and illustrate by code how it works.

Keywords
Single display groupware, interface toolkits, CSCW.

INTRODUCTION
While there is much interest in single display groupware (SDG) [1,2,3], these kinds of systems are still notoriously hard to build. Typically, most researchers develop their own specialized applications from the ground up, resulting in SDG that is tedious to build, difficult to modify, and hard to replicate. This problem is exacerbated by our current generation of operating systems that make it difficult to do even the most basic SDG activities. For example, if multiple mice and keyboards are plugged into a computer, a programmer has to do low-level device manipulation to retrieve and process the various inputs as separate streams. Even when this is done, the programmer has to draw multiple cursors (one for each mouse), interpret delta coordinates into window system coordinates, attach a keyboard stream to a window, and so on.

Because we wanted to pursue research in SDG, we decided to build a toolkit that would help us and others rapidly develop SDG interface components and applications. This paper reports our progress on our toolkit, which we call SDG Toolkit. In particular, we focus on how we manage input from multiple mice and keyboards. We describe how SDG Toolkit appears to the SDG application developer, and then how it works under the covers.

Cite as:
Figure 1. A simple SDG program written in Visual Basic.

```vbnet
Dim WithEvents g As sdgControl
'Create the control & make returned coordinates relative to the main window.
Also change the way each mouse's cursor will look
Sub Form_Load()
    Set g = New sdgControl
    g.SetRelativeTo Form1 hWnd
End Sub

Sub g_KeyDown(KeyCode As Long, Shift As Long)
    MsgBox "Key: " & KeyCode & " & " & g.LastMouseClick & " Clicked"
End Sub

Sub g_MouseMove(MouseID As Long, Button As Long, X As Long, Y As Long)
    MsgBox "Moved: " & MouseID & " & " & X & " & Y
End Sub

Sub g_MouseWheel(MouseID As Long, WheelDelta As Long)
    MsgBox "Wheel: " & MouseID & " & " & WheelDelta
End Sub

Sub g_MouseClick(Button As Long, MouseID As Long, X As Long, Y As Long)
    MsgBox "Mouse " & g.LastMouseClick & " Clicked"
End Sub

Sub g_MouseWheel(end As Long, WheelDelta As Long)
    MsgBox "Wheel End Adjusted: " & WheelDelta & " & " & g.LastMouseClick & " Clicked"
End Sub

Sub g_MouseMove() As Long
    MsgBox "Moved: " & g.LastMouseClick & " Clicked"
End Sub

Sub StandardButton_Click()
    MsgBox "Mouse " & g.LastMouseClick & " Clicked"
End Sub
```

Figure 2: SDG Sketch. Users change colours by moving the mouse wheel up and down which handle transparency effects. Fourth, it raises events for all keyboard and mice actions, such as those illustrated in Figure 1. Finally, the SDG toolkit invisibly moves the real mouse cursor to the location of the currently active SDG mouse, and raises standard mouse events corresponding to the SDG mouse events. This allows any of the users to interact (albeit in a limited way) with non SDG-aware widgets and standard window controls.

**FUTURE WORK**

SDGToolkit currently provides only the primitives needed to manage multiple mice, keyboards and pointers. While one can now create SDG applications such as the one in Figure 2, the problem is how widgets (buttons, menus etc) are handled. SDGToolkit does support limited interaction with standard widgets, but it lacks SDG-aware widgets i.e., widgets that respond correctly to different people and/or to simultaneous actions over it. We are currently implementing an extension that lets us and others create true SDG widgets. We will test these widgets, and successful ones will be included in SDGToolkit.

**ACKNOWLEDGMENTS.** This research was partially funded by NSERC, ASERC, and Microsoft Research. Mike Boyle helped with Windows. Russell Kruger shared his thoughts on tabletop displays.

**Software availability:** see [www.cpsc.ucalgary.ca/grouplab/](http://www.cpsc.ucalgary.ca/grouplab/)