Interfacing with Interactive Workspaces: Extending the Desktop

Jim Wallace
Dalhousie University
6050 University Ave.
Halifax, NS, Canada B3H XXX ?
jrwallac@cs.dal.ca

Dr. Kori Inkpen
Dalhousie University
6050 University Ave.
Halifax, NS, Canada B3H XXX ?
inkpen@cs.dal.ca

CHI 2006, April 22–27, 2006, Montreal, Canada.
ACM 1-xxxxxxxxxxxxxxxxx.

Overview

The overall goal of our research is to explore ways of making interactive workspaces a usefull tool for casual collaborative work. As the technology becomes more available, future users will soon have many displays and processors available to them in their own homes, and perhaps even in public. Without a standard way of interfacing with these environments users will be left confused, and systems will go unused. There is clearly potential in this area to reshape the way computing is seen by the average user.

Introduction

The face of casual computing is changing. Projectors, large displays and handhelds are cheaper and more pervasive than ever before, and finding an increasing number of uses for the average person. This continuing trend is making common Interactive Workspaces (IWs) inevitable, providing the chance to take them out of the laboratories and into the home.

Projects such as PointRight [3] and ARIS [1] have proven that working on multiple screens is technically feasible. Now that the foundation has been laid through networking and software solutions, and the underlying

frameworks have been established the last piece of the puzzle is to build a comprehensive interface for controlling how devices interact. The pieces are all there, it is only a matter of presenting them to the end user in a manner that "makes sense"; designers must choose a metaphor to describe how systems interact with each other.

It is not clear that the traditional desktop metaphor extends in a manner that can adequately describe connecting and disconnecting devices from an IW; it was only intended to simulate a single desktop, not a flexible workspace that adapts to what devices are available to it. It is also not clear that users would be willing to exchange the traditional metaphor with which they are accustomed if a more fitting one were found.

Regardless of the metaphor used, there are two key aspects of an interface for an interactive workspace that must be considered:

1) How devices enter or leave an interactive workspace

2) How devices interact within a specific interactive workspace

This distinction is important because with the prevalence of mobile devices it will soon be impossible to consider interacting with all devices within an area under all circumstances.

To date, mostly the second class of problems have been addressed in the literature, yet the first class is at least as important – and is possibly a more difficult problem. Only a small number of obstacles have been encountered while extending the desktop metaphor to cover a virtual workspace consisting of several displays. Most projects have tried to ensure that features such as cut & paste, cursor movement and program movement work in a similar manner to single display systems.

Researchers have just assumed that devices are "connected" or are independent and have not worried about the methods by which this is accomplished. How users will control this state change is important since it sets the context within which the devices will interact. At some level the system must understand the relationships between two devices, and it will likely be impossible to distinguish these relationships without human intervention in some cases.

In the cases where human input is needed, it is important to make this interaction as lightweight as possible. Just as a user today can "plug" their iPod into their computer in order to update their music, they will soon need to be able to "plug" their mobile devices into interactive workspaces. It may not be as simple however since many mobile devices may be in the same area, and ideally no physical connection should need to be made. The implicit information contained in the physical connection with the iPod needs to be made explicit in the case of interactive workspaces. The key to successfully bringing interactive workspace to the common user is in making this task as lightweight as possible.

By the very definition of an interactive workspace, where devices of different natures are used in conjunction with each other, the metaphors used for a single device cannot be expected to extend to cover the entire space.

As IWs become more common it will be necessary for users to join and leave different environments throughout their day. A lightweight means to change these settings is necessary in order for this to be a feasible activity. Interaction techniques also cannot be tied to a single input device; a variety of devices will be expected to work in harmony in a single space. Creating a lightweight, device agnostic interface is certainly a challenge.

Past/Current Work

Our initial work has focused on interaction techniques within interactive workspaces. One of our projects, called Swordfish, focused on implementing lightweight mechanisms for specifying links between displays within an interactive workspace. Swordfish allows users to create "Bindings" between displays that their mouse cursor can later traverse. In this manner a user specifies their own private workspace within an interactive workspace.

This is similar to work done with the TeamSpot [5] commercial software package, but uses a different metaphor to accomplish the task. The advantage of working in this manner is that a single display is treated as a building block from which a large interactive workspace can be formed. Rather than creating entirely new metaphors, existing metaphors are extended to allow for connections with other displays.

As mentioned before, it is crucial to investigate methods for the formation of interactive workspaces. This work has been a preliminary investigation of such interaction techniques. Solutions used by commercial software [5] have not provided lightweight mechanisms for this task, the interface has been hidden behind multiple levels of menus. It is important to investigate lightweight methods in real world scenarios when possible so as to make this task as simple as possible for the end user. Even small gains in the simplicity of this task can translate into major usage gains.

A video which summarizes much of this project is available at www.cs.dal.ca/~jrwallac/Swordfish/Video.

This project also explored how users visualize the displays available to them within an IW and identified two strategies used to organize interactive workspaces: user- and environment-centred layouts. This finding supports the importance of supporting a variety of users within an IW. This is particularly important when considering co-located collaborative work.

Our current work in this area has focused on extending Swordfish to allow the redirection of images from one display to another within an interactive workspace, giving users the ability to create a private view of public materials or to share portions of their private display (similar to WinCuts [4]). We plan to study how this technology can be leveraged to help tablet PCs work within an IW. While previous work in the literature [3] provide methods for selecting which display within an interactive workspace to interact with, they lack the ability to redirect output from one system to another which may be crucial for tablet PCs.

We have two specific goals in mind for this project: to explore how attention is divided between public and private displays, and to investigate different interaction techniques for controlling these views. Current

prototypes are available through implementations in the Swordfish project listed above, but are limited. Future work will focus on exploring alternative interfaces and improving the underlying technology.

Conclusion

Designing interfaces for interactive workspaces is a hard problem because designers must break many of the existing paradigms used in single display systems, yet users will expect these environments to work in a similar fashion to their existing systems. Where previously interface designers could anticipate the system as a whole, now they can only consider the system as a small part of the overall picture. Even more importantly the device in question is at times the entire system, but later may only be a small piece. Creating a lightweight, understandable mechanism to control when a system should act on its own and when to depend on nearby devices is a critical problem to be addressed.

References

[1] Biehl, J.T. and Bailey, B.P. Aris: An Interface for Application Relocation in an Interactive Space. In Proc. GI 2004, Canadian Human-Computer Communications Society (2004), 107-116.

[2] Booth, K.S., Fisher, B.D., Lin, C.J.R. and Argue, R. The "Mighty Mouse" Multi-Screen Collaboration Tool. In Proc. UIST 2002, ACM Press (2002), 209-212.

[3] Johanson, B., Hutchins, G., Winograd, T. and Stone, M. Pointright: Experience with Flexible Input Redirection in Interactive Workspaces. In Proc. UIST 2002, ACM Press (2002), 227-234.

[4] Tan, D.S., Meyers, B., Czerwinski, M. WinCuts: manipulating arbitrary window regions for more effective use of screen space. Ext. Abs., CHI '04, ACM Press (2004), 1525-1528.

[5] Teamspot. www.teamspot.com. Accessed December 2005.