

Sponsored by:

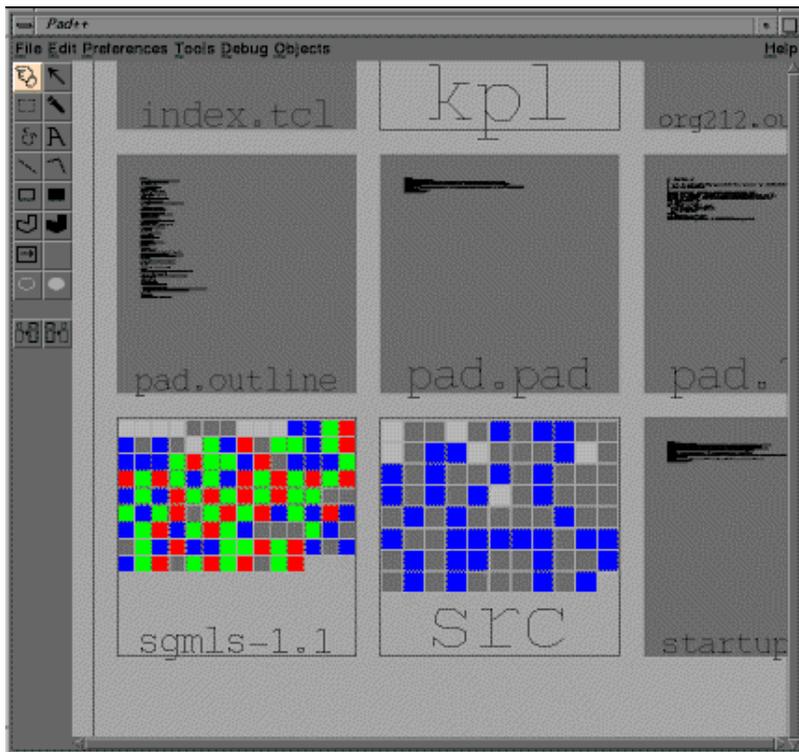


Peripheral Visions

Zooming Through Information Space on PAD++

by Gary Welz

Copyright © 1995 Gary Welz, All Rights Reserved. Used With Permission.



The X Advisor: June 1995 - Vol 1 No 1. <http://landru.unx.com/DD/advisor/>

DISCOVERY PUBLISHING GROUP

On a recent visit to the New York University Media Research Laboratory I learned about Pad++ a Unix system substrate for exploring zoomable interfaces. Pad++ enables groups of users to navigate in a shared information space by tapping into their "natural spatial and geographic ways of thinking."

The software is being developed by a consortium of researchers at the University of New Mexico, NYU, and Bell Communications Research. The project is funded by ARPA's Human Computer Interaction Initiative. Professor James D. Hollan is the principal investigator.

A "Zoomable" Surface

At first glance the Pad++ screen looks like the desktop of the Macintosh with folders arrayed as squares on a two dimensional surface. A Mac user would double click on a folder icon and see the contents in a window. To see the contents of any folder within that folder, the user would again have double click to create a new window. A Pad++ user, on the other hand, views a directory through a "portal" - a magnifying glass that allows him or her to roam over different parts of a single infinite shared desktop and "zoom" into any portion of the surface, descending into folders to reveal the directories without clicking.

Ken Perlin came up with the initial zoomable surface concept. He and David Fox implemented the first versions of Pad, which were the precursors to the current implementation, Pad++. Perlin and Fox's original work was supported in part originally by NYNEX, and then the National Science Foundation. The primary creator of Pad++ is Ben Bederson. The development team is now centered at UNM.

Describing Pad, Perlin says, "The concept is very natural since it mimics the way we continually manage to find things by giving everything a physical place. A good approximation to the ideal depicted would be to provide ourselves with some sort of system of 'magic magnifying glasses' through which we can read, write, or create cross-references on an indefinitely enlargeable ('zoomable') surface."

Although the zooming capability can only be suggested by still pictures, consider this view of the top level of the Pad++ desktop with a number of folders laid out upon it.

Figure 1 - Top level of the Pad++ desktop

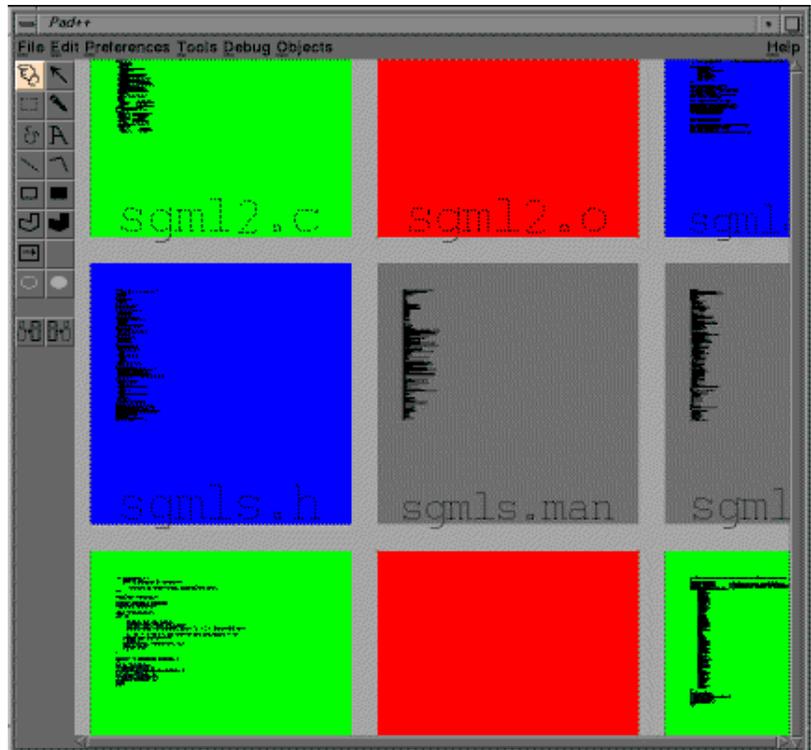


Figure 3 - Zoom to the document level

It's even possible to zoom in further to create or view notations on a still smaller scale - at the same size as the text or microscopically smaller.

Scanning the Information Horizon

An important distinction between the Pad++ universe and the universe of other window systems is that in Pad++ every interaction object possesses an absolute physical location. In this sense Pad is a two dimensional virtual reality. Yet a user's changing view can allow objects to appear larger or smaller.

This interface puts vast quantities of information directly at the user's fingertips. Several million pages of text could be fit on the surface making entire encyclopedias or databases directly available for perusing. Call it the "messy-desk syndrome" but I like to spread my work out in front me - there's something about being able to take it all in at a glance that makes it easier to understand the relationship between the various parts.

As the magnification of an object changes, the user can opt to see different types of information about it. For example, when a text document is small on the screen the user may only want to see its title. As it's magnified a short summary or outline would become visible and eventually the entire text would be revealed. This is called "semantic zooming."

Looking at the World Through "Non-literal" Lenses

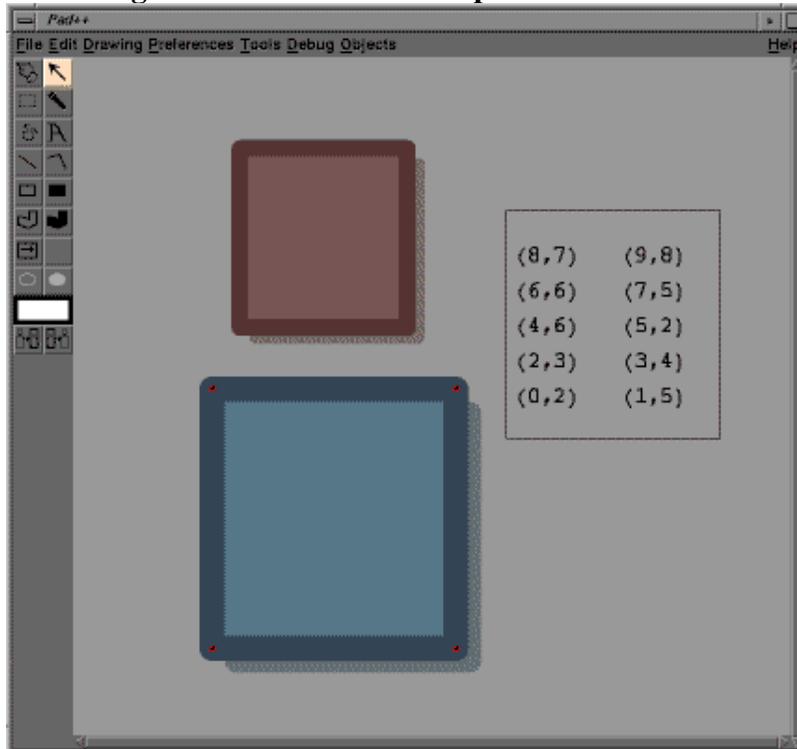
Objects can also manage "portal filters" - portals that show non-literal views of cooperating objects. These portals can be thought of as lenses that offer different ways to represent the same underlying data. For example, in a literacy application, lenses could give feedback to the learner by showing a picture

that represents the word underneath it (something they've implemented) - a different lens could pronounce that word, etc.

Another lens might show all tables of numerical data as bar graphs. An application could then allow text or spreadsheet style editing of the table. These edits would be seen as changes in the graph.

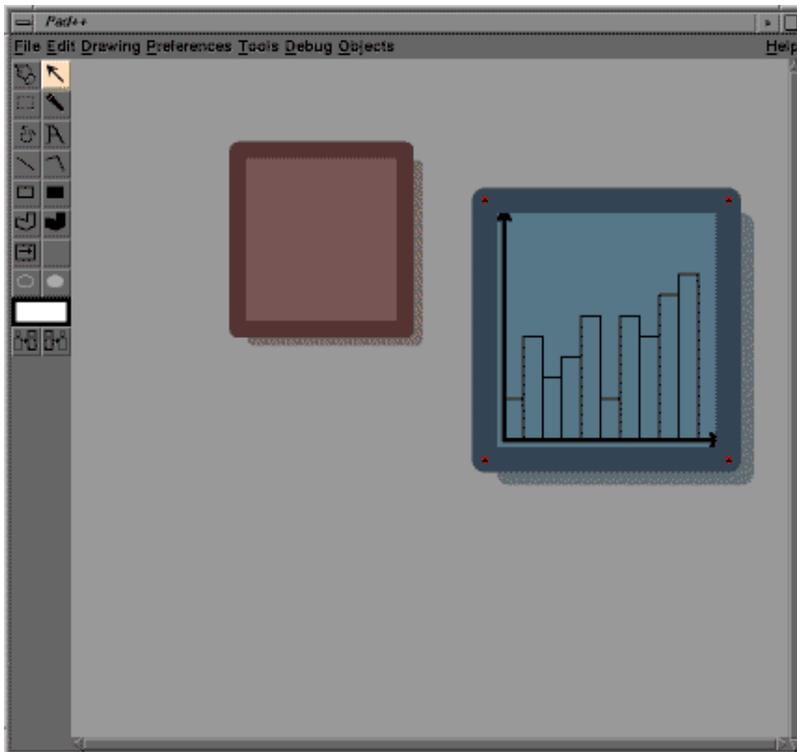
For example, here's a list of ordered pairs of numbers

Figure 4 - A list of ordered pairs of numbers



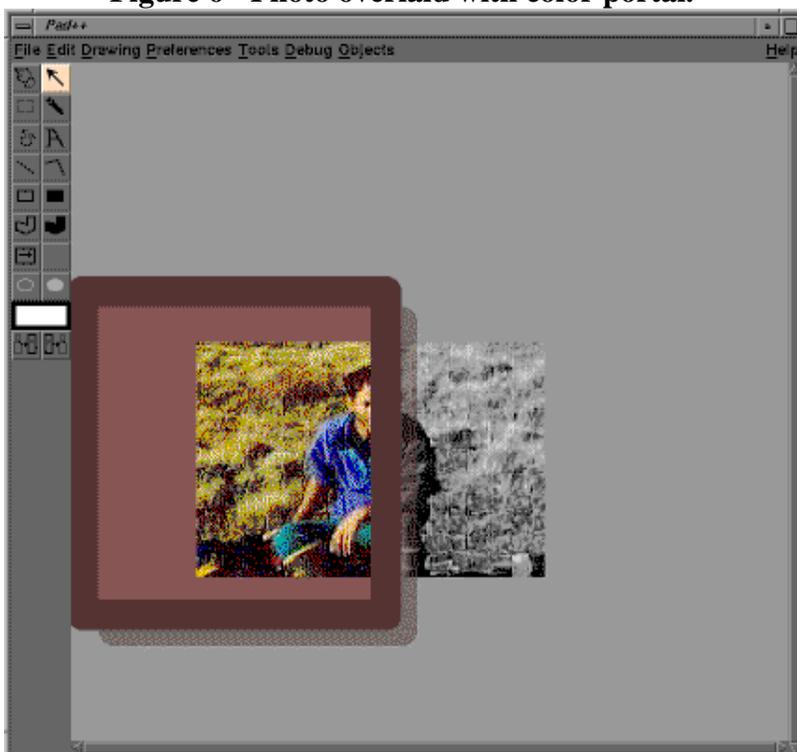
that can be represented as a bar graph

Figure 5 - A bar graph representing the numbers



Another type of portal filter can be used to expose different qualities of an an image. Here a color filter can be used to view a black and white picture in color.

Figure 6 - Photo overlaid with color portal.



Network Collaboration in a Pad++ Virtual Workspace

Pad++ also has tremendous potential as a long distance collaborative work tool because it allows multiple users on a network to work together in the same apparent information space. Different users could share and view multiple applications while assigning each user a desired degree of interaction. Documents could be visually nested and zoomed as they are moved back and forth between primary and secondary working attention. Because of the zooming capability one could see the layout (in miniature) of many collaborators' surfaces at a glance.

Ben Bederson says, "None of the network aspects of Pad++ is currently implemented. My plan is to have local Pad++ applications running separately communicating through a spatial database server. The interface would be such that you could open a portal onto a distant Pad++ surface."

Pad++ in 3D?

It's natural to ask why Pad++ isn't 3D? Jonathan Meyer says, "There are some very attractive 3D systems out there (the Information Visualizer is one of them). However, although they may look impressive, 3D systems are typically hard to navigate using current display and pointer technologies. They also require high levels of computing resources. It's not obvious that a 3D interface is appropriate for a hand held device or for a cheap home computer. Pad++, on the other hand, can be implemented very efficiently using a small home computer, and is easy to navigate using a mouse."

Two Views of Pad++

David Fox says, "Pad++ can be seen as a mechanism for creating applications which have a zooming component. Zooming seems to be particularly effective for applications which need to present large amounts of data in an intuitive and understandable manner." This is especially true for data that is hierarchical in nature. For example, there may well be projects using the Pad++ metaphor to do data visualization with taxonomic or economic databases.

Pad++ could also be viewed as a way to implement an alternative windowing system based on zooming. Your whole desktop could be zoomable. This seems especially attractive for systems which have small screens, such as hand held computers (i.e. PDA's). Jonathan Meyer thinks that it could be popular in set top controllers for interactive television systems. Advertisers in a 2-dimensional information space might find the zooming capability of Pad++ really allowed them to "suck in" customers.

Pad++ Development

The Pad++ group is engaged in a number of projects to exploit the potential of the zoomable interface. Among them is a Pad++ Web browser that allows users to view their entire session history on the desktop and zoom back and forth through it. They are also developing Pad++ applications for understanding large data sets. and investigating how to embed traditional WIMP applications within Pad++.

The Pad++ Team

The complete Pad++ development team consists of:

- University of New Mexico, CS Dept.; Benjamin Bederson, James Hollan, Eric
- DeMund, David Vick
- The NYU Media Research Laboratory; Ken Perlin, David Bacon, Jonathan Meyer,
- David Fox.
- Bellcore; George Furnas, Kent Wittenburg, Larry Stead

Pad++ Availability and More Info:

The developers are planning to make Pad++ available for education and research in the near future. Send email to pad-info@cs.unm.edu and they will notify you when it's ready.

You can take a "tour" of the Pad++ system and obtain information about its development via the Pad++ WWW site at <http://www.cs.unm.edu/pad++>.

There is a Pad++ ftp site at <ftp://ftp.cs.unm.edu/pub/pad++>. They currently have postscript versions of papers there, but eventually will offer all kinds of things (including code).

Gary Welz is a New York City based journalist, consultant and WWW designer. He's currently writing a book about online multimedia and can be reached via email at gwelz@holonet.net, or come visit his home page at <http://found.cs.nyu.edu/found.a/CAT/misc/welz/>

How would you rate this article?

No Opinion	Excellent	Good	Fair	Poor		
------------	-----------	------	------	------	--	--

Send comments to gwelz@holonet.net or webmaster@unx.com.



New Cool Popular [Mail] Search Help

This page and all its contents are Copyright ©1994-1995 Digital Discovery, Inc. USA, All Rights Reserved. The Discovery Web®, The X Advisor® and Cross Platform Solutions® are all registered trademarks of Digital Discovery, Inc.