Navigation/Zooming

Presented by Peiqun (Anthony) Yu

Papers Reviewed

- A Multi-Scale, Multi-Layer, Translucent Virtual Space Henry Lieberman, IEEE International Conference on Information Visualization, London, September 1997.

- Constant Information Density in Zoomable Interfaces Allison Woodruff, James Landay, Michael Stonebraker, Proceedings of AVI '98, pp. 57-65.

- Domain Name Based Visualization of Web Histories in a Zoomable User Interface. R. Gandhi, G. Kumar, B. Bederson and B. Shneiderman. *In Proceedings of the Second International Workshop on Web-based Information Visualization (WebVis'00), pages 591-598, Sep. 2000.*

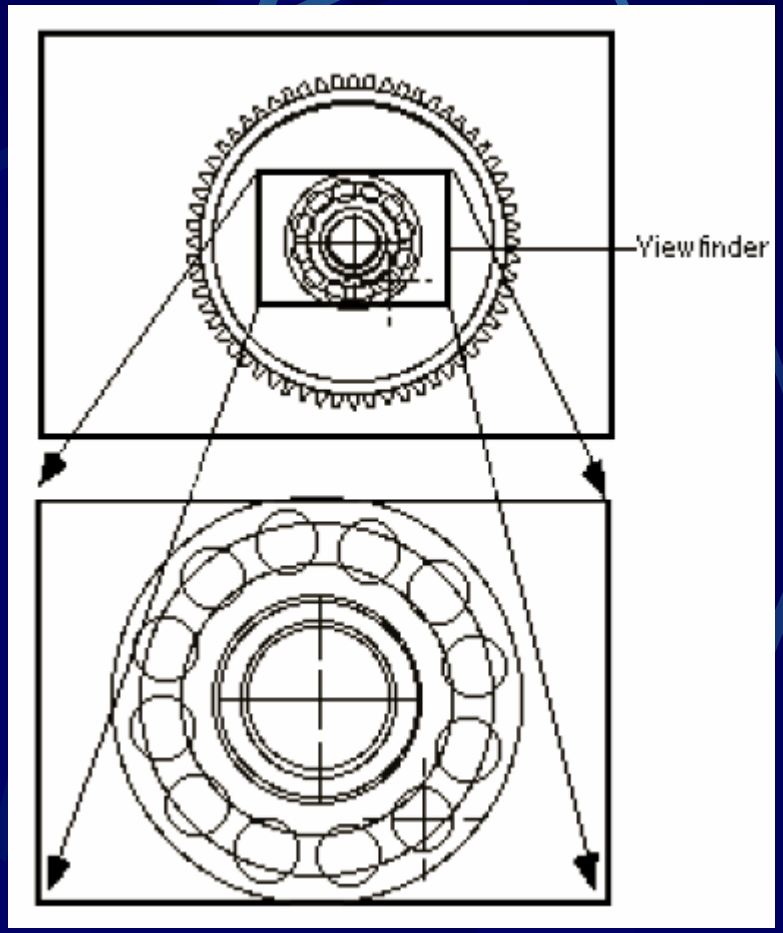
A multi-scale, multi-layer, Translucent Virtual Space

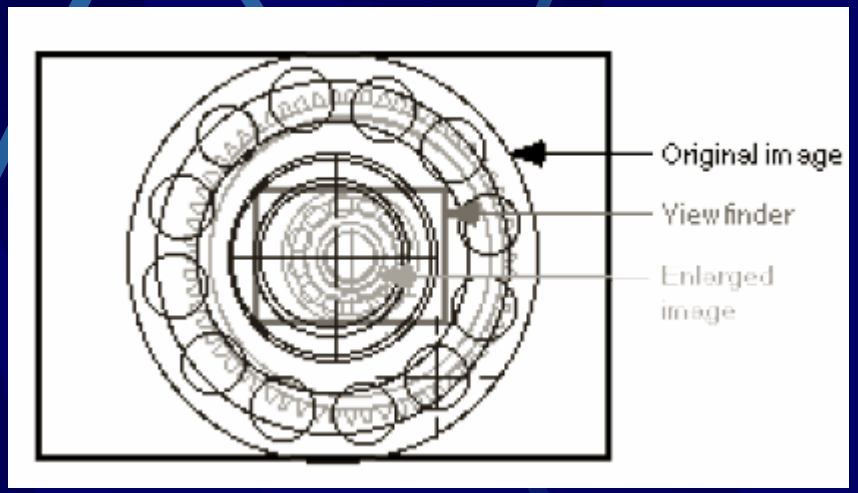
- Browsing very large spaces of displayed information at different scales
- Introducing multiple translucent layers to avoid the problem of losing visual context

The Macroscope: Translucent Zooming and Panning

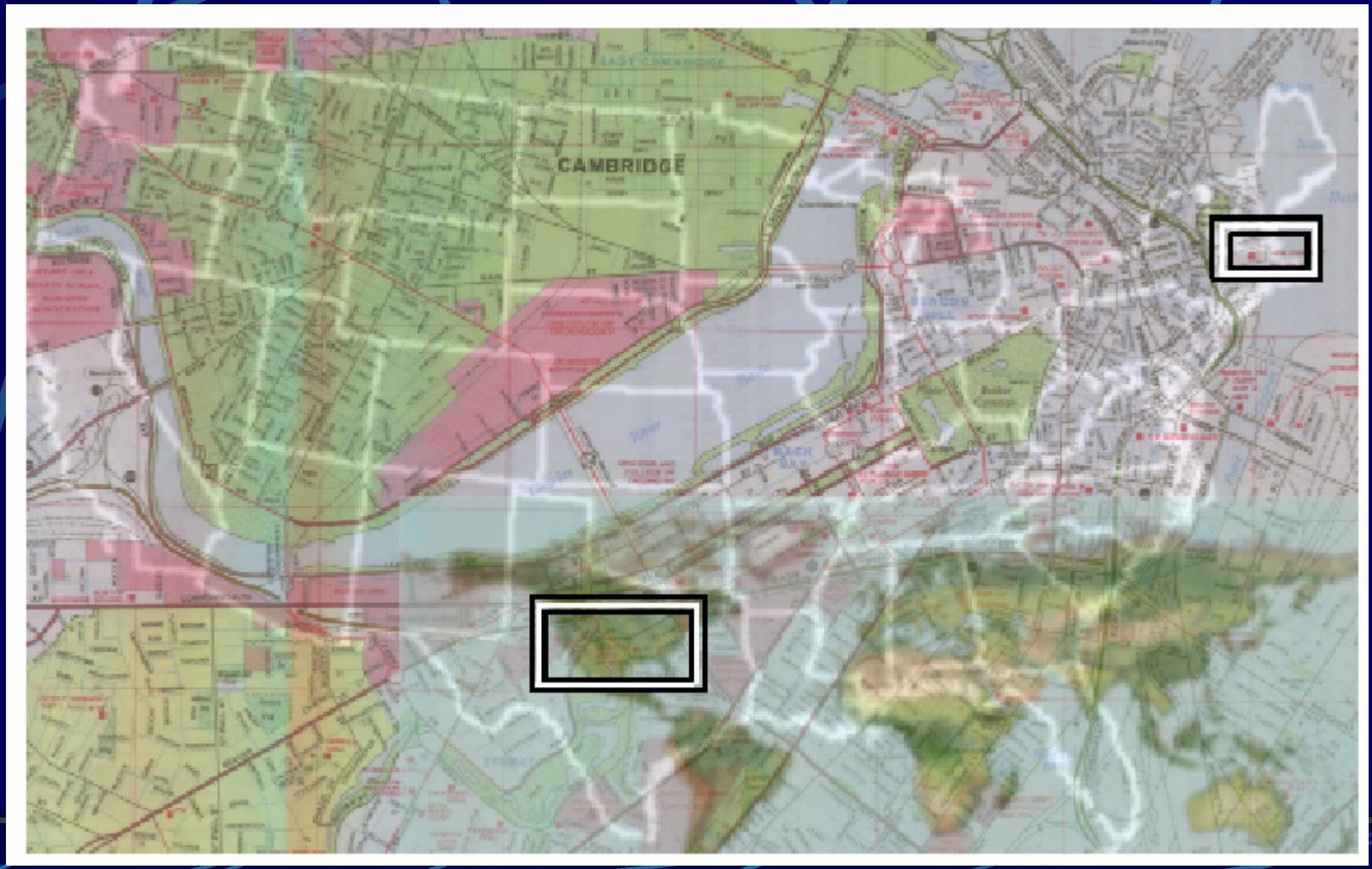
- A typical zooming operation and its problem.
  - Blows up the viewfinder to fill the entire image
  - Problems: the viewer loses the context of where the blown-up image came form
- Solution – the macroscope:
  - Makes the zoomed-in and zoomed-out views share the same physical screen space by displaying them in multiple translucent layers

Typical Zooming vs. macroscope





An Example of a Three Layer Zoom

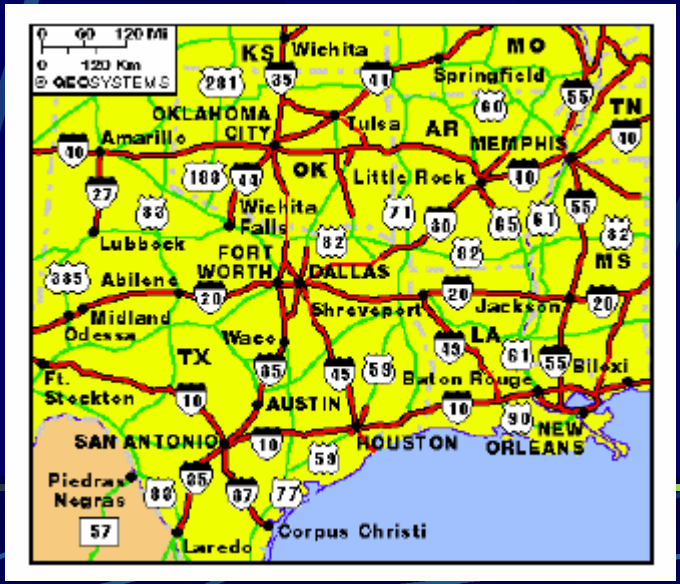


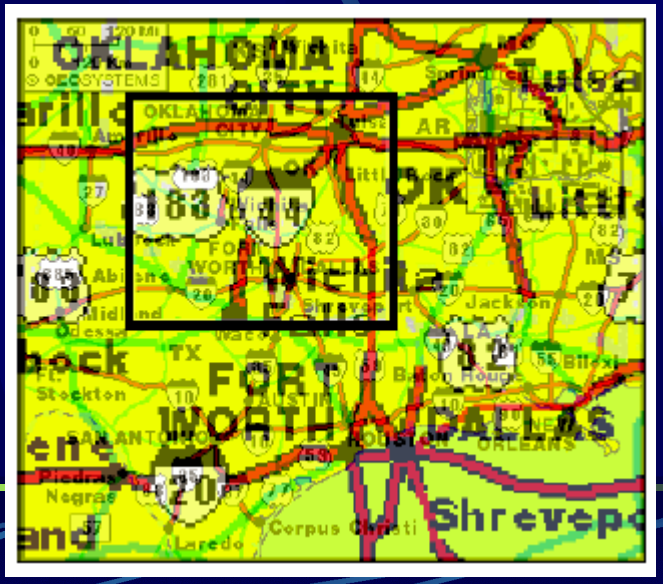
Interactive Control of the Macroscope

- Multiple scales can be seen simultaneously
  - Viewers can select the viewfinder in the layer that is at the appropriate scale and adjust it
- The system can make the correspondence between viewfinders and their layers
- Viewers can dynamically adjust the translucency levels between layers

Examples of Macroscope

- Superimposed on the original map is the enlarged image of the viewfinder area
- The resolution and the sizes of features (roads, city, names) help in distinguishing the two layers





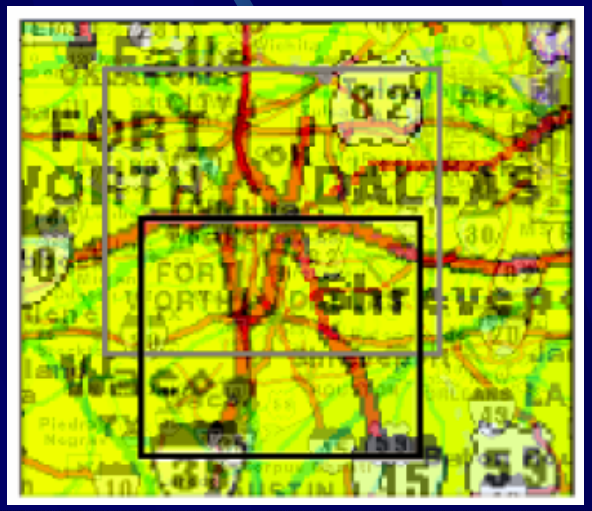
Examples of macroscope (Cont.)

- As the viewfinder is dragged, the scale of the zoomed-in view changes size
- When panning the viewfinder, the background remains the same, but the superimposed layer changes

Shifting Attention and Generating Multiple layers

- Relative translucency of the layers can be dynamically adjusted to emphasize either the higher or lower layers

- Selecting a rectangular portion of the image can generate a three-layer macroscope

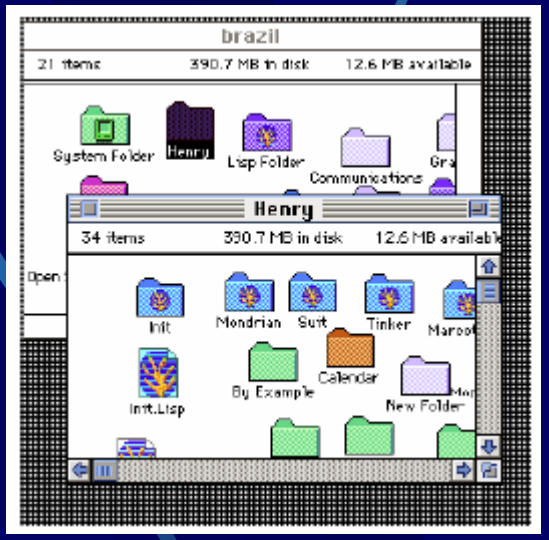


Multiple resolution maps

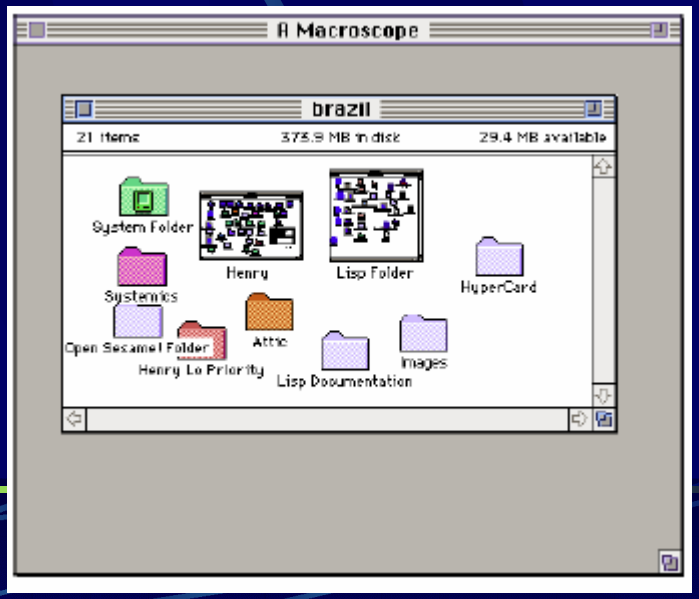
- One can also use multi-resolution maps, so that zooming into a map bringing up a map of higher resolution

Another Example: Graphic Display of Hierarchical file System

- A conventional graphical display of a hierarchical file system

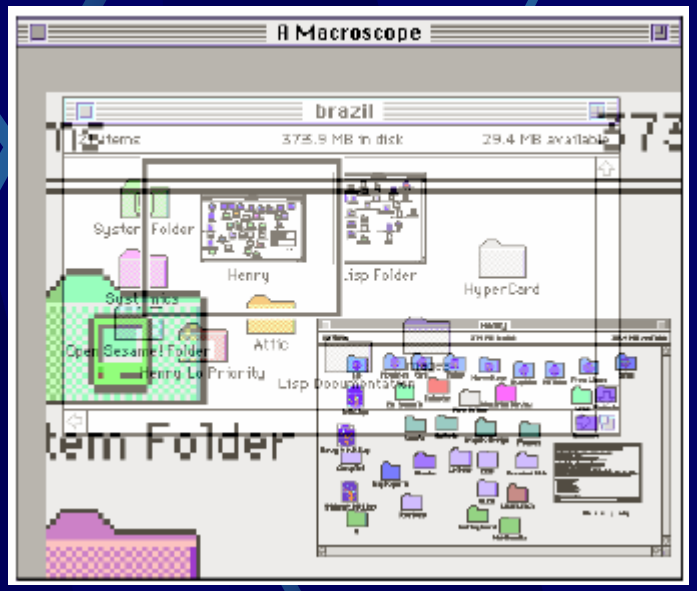


- In the macroscrope version, each icon graphically contains all of the files and folders within it, at a much reduced size

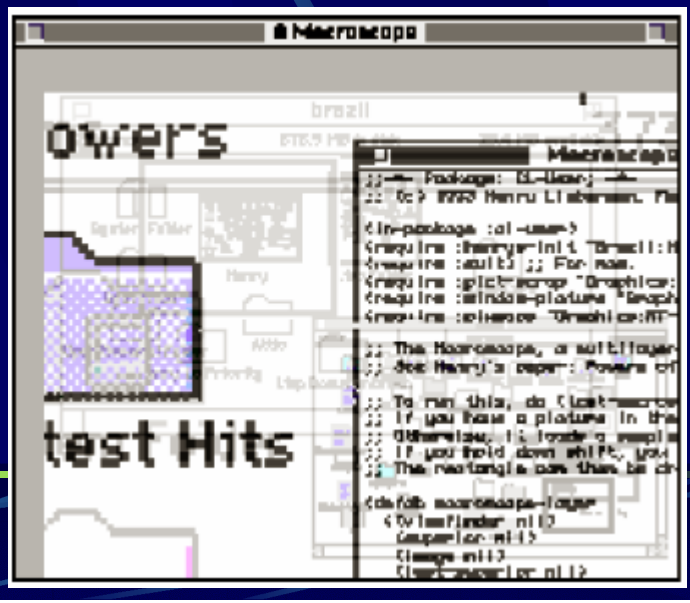


Hierarchical File System (cont.)

- No opening or closing of folders, just zooming into the contents of a folder



- One can zoom into the contents of an individual file containing text.



Critique

Strong Points

- Effective and sufficient examples
- Effective techniques
  - Interactive control
  - Multiple layers

Week Points

- The figures and pictures are not labeled
- Doesn’t show how to adjust the translucency levels
- No implementation details

Where We Are

- A Multi-Scale, Multi-Layer, Translucent Virtual Space Henry Lieberman, IEEE International Conference on Information Visualization, London, September 1997.
- Constant Information Density in Zoomable Interfaces Allison Woodruff, James Landay, Michael Stonebraker, Proceedings of AVI '98, pp. 57-65.
- Domain Name Based Visualization of Web Histories in a Zoomable User Interface. R. Gandhi, G. Kumar, B. Bederson and B. Shneiderman. *In Proceedings of the Second International Workshop on Web-based Information Visualization (WebVis'00), pages 591-598, Sep. 2000.*

DataSplash

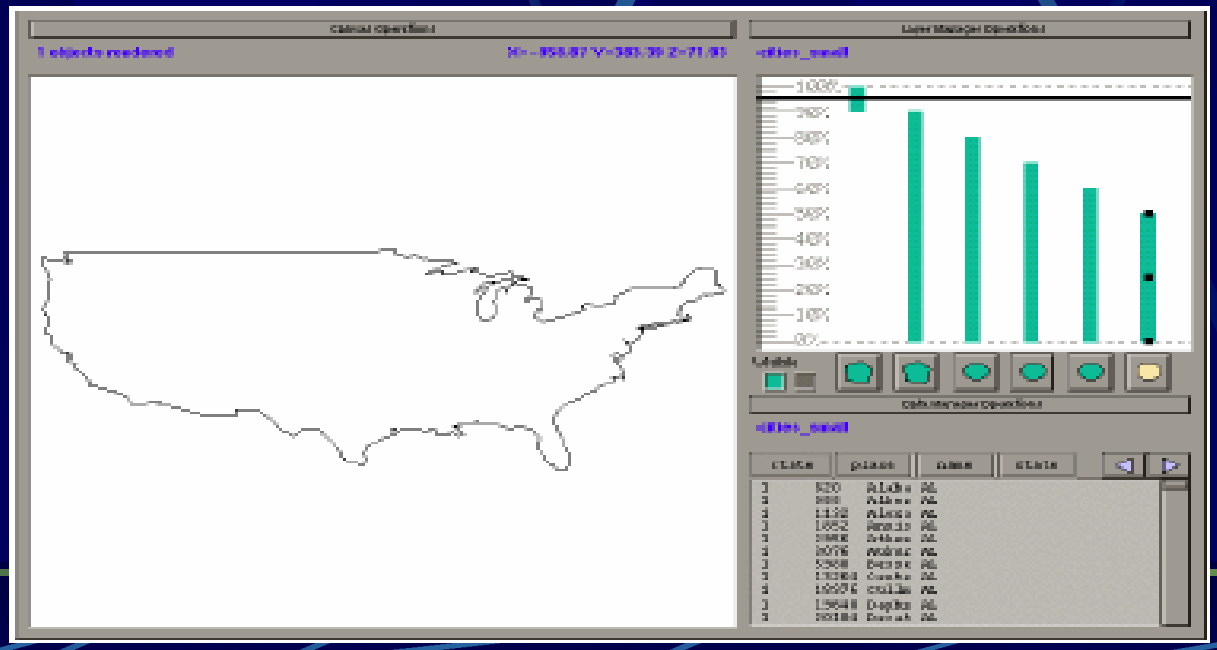
- DataSplash is a direct manipulation system in which users can construct and navigate visualizations
- DataSplash provides a layer manager, which allows users to visually program the way objects behave during zooming
- It's difficult to construct visualizations that display an appropriate amount of detail at all elevations
- This paper proposes an extension of the DataSplash database visualization environment (VIDA)

Principle of Constant Information Density

- Number of objects per display unit should be constant
- The amount of information should remain constant as the user pan and zoom
- To maintain constant information density
  - Either, objects should be shown at greater detail when the user is closer to them
  - Or, more objects should be appear as the users zooms into the canvas
  - Or, both

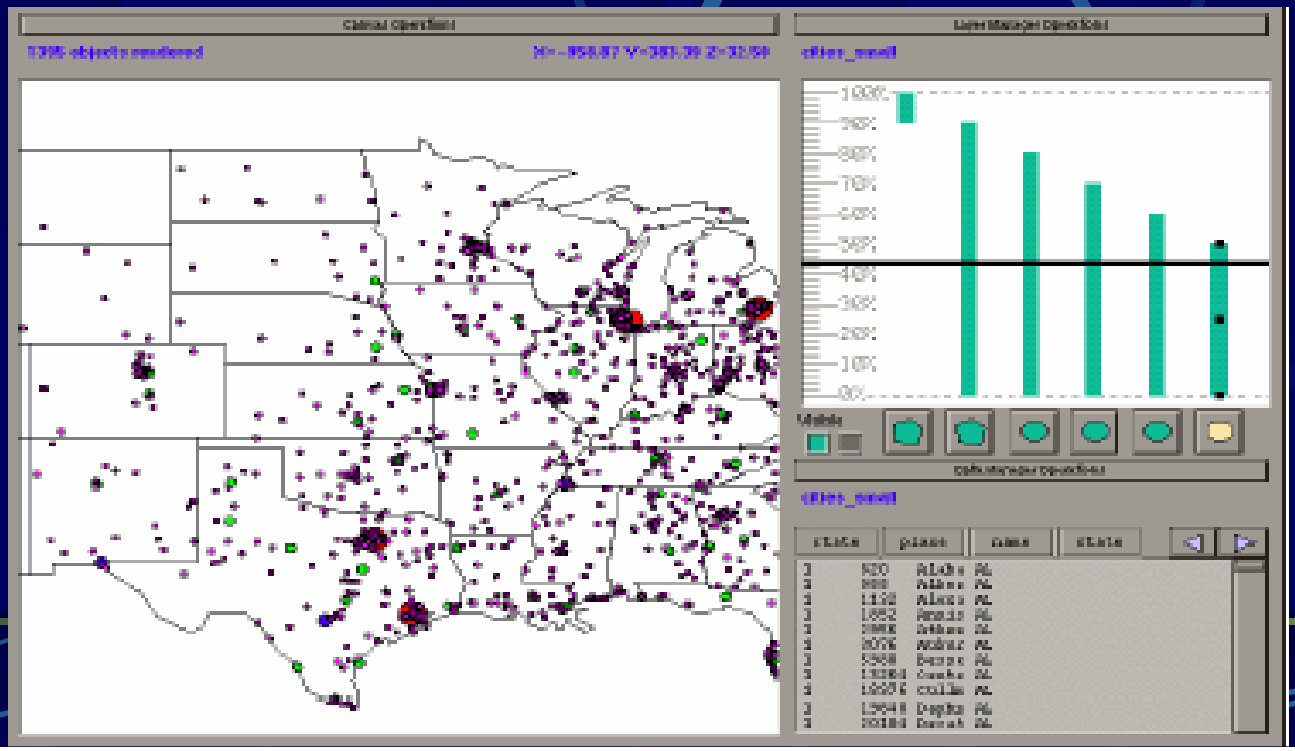
The DataSplash environment

- Each layer appears as a vertical bar in a layer manager
- All objects in a canvas are organized into layers
- Each object is a member of exactly one layer
- Each layer is associated with exactly one database table
- Each row in the table is assigned an x,y location in the canvas



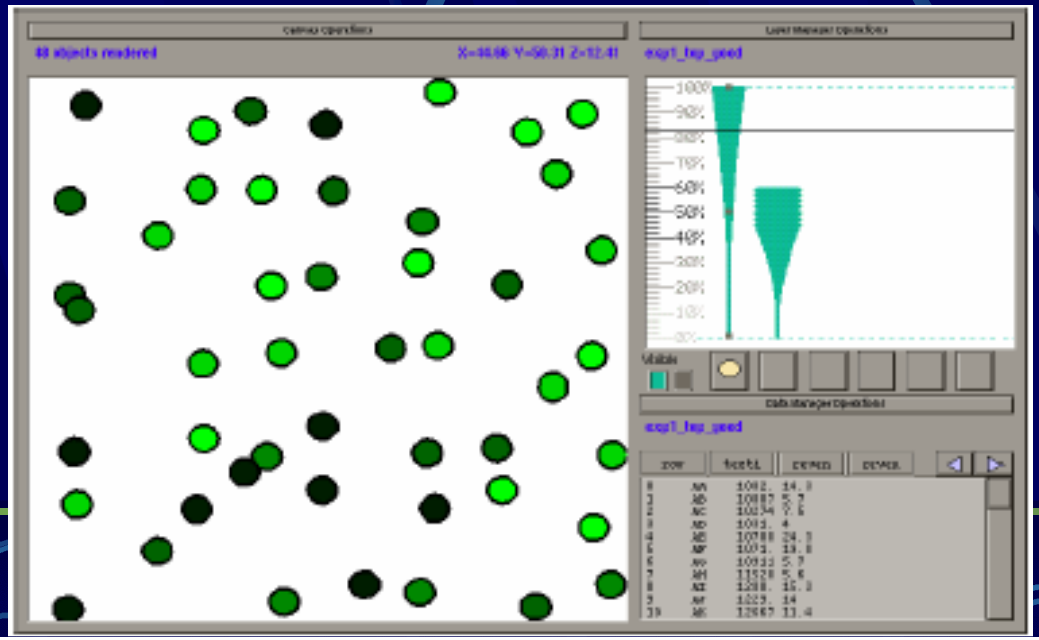
The DataSplash environment(cont.)

- The current elevation is shown with a horizontal elevation bar
- Any layer bar that is crossed by the horizontal elevation bar is considered to be active and objects are rendered
- An icon of the type of the object displayed by each layer appears in the button below its layer bar



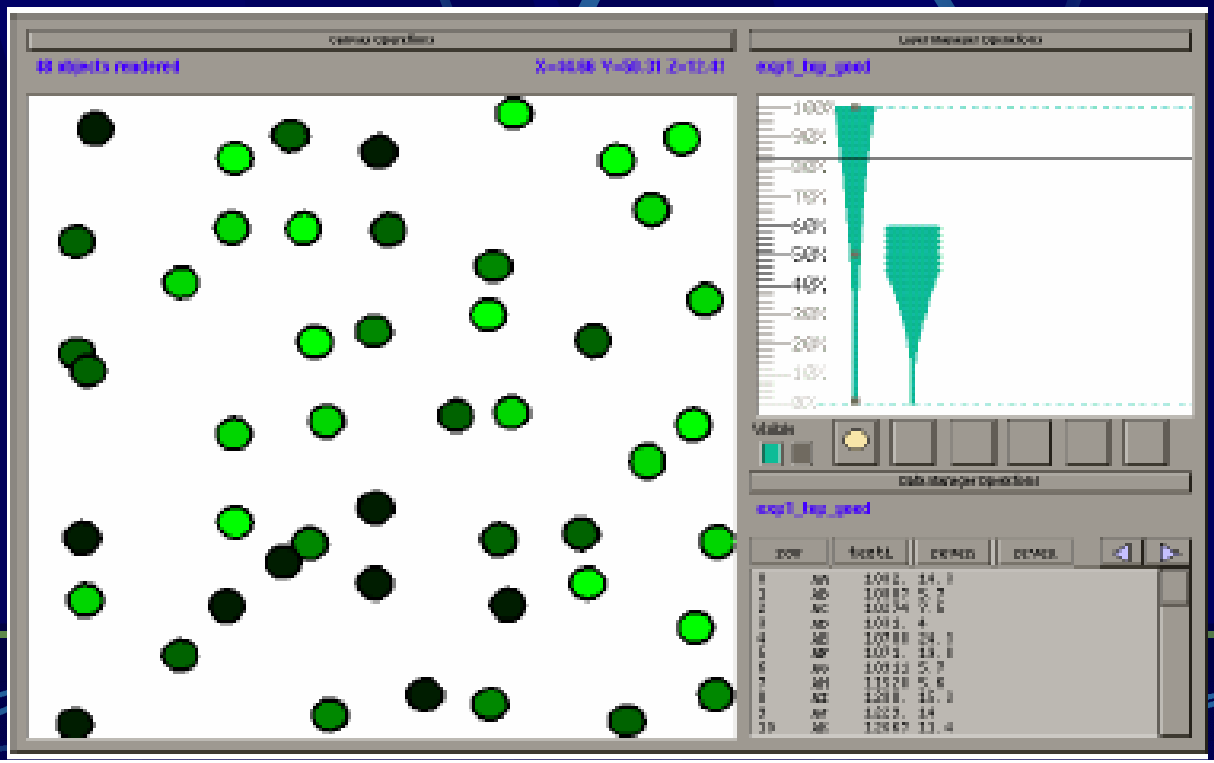
VIDA - Providing visual density feedback

- The width of each layer bar reflects the density of corresponding layer at the given elevation
- Tick mark is assigned one of three colors to indicate which condition pertains at a given elevation (Users can specify the bound to define a range of acceptable densities)
  - lie within the density bound
  - fall below the minimum density bound
  - exceed the maximum density bound



What does the figure tell us?

- Elevations 40%-60% are too dense
- The area of the native space visible increases quadratically, therefore, the object density increases quadratically as the elevation increases
- The rate of change in width is more pronounced by the layer bar on the right, because the right-hand layer bar contains more objects

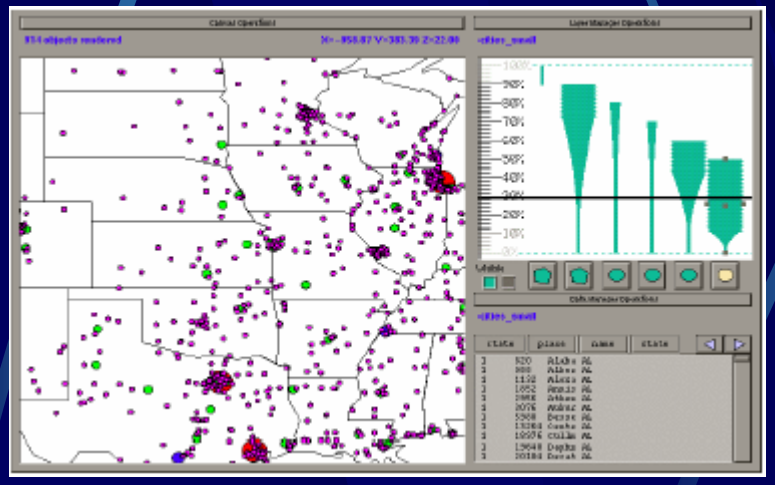


User Interaction With the New layer Manager

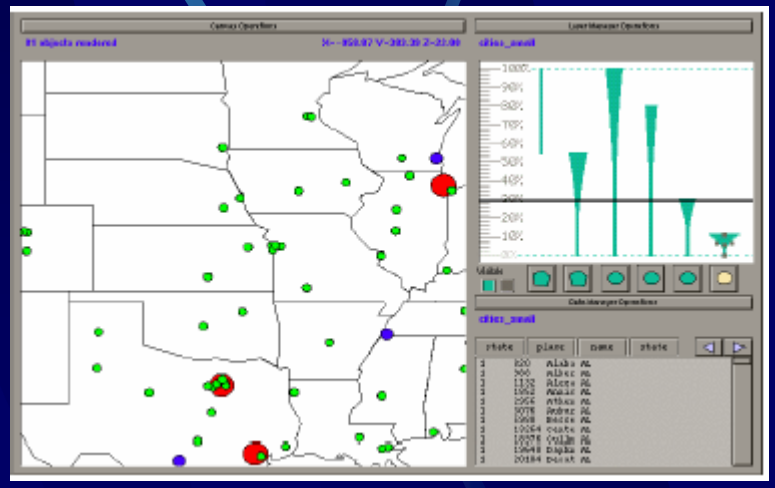
- Users can modify the layer manager
  - Adjust the top or bottom of a layer bar
  - Drag the entire layer bar up and down
  - As the user modifies the bar, the colors of the tick marks change to reflect the modification
- Users can change the contents of layers
  - Use the paint program interface to modify the contents of a layer
    - For example, to modify the number of objects
  - Use the visual select and join mechanisms. These operations affect the number of rows in the table associated with the layer
- The extensions of the layer manager also teaches the user about the properties of density function in general

Non-uniform Data

A Clutter Application



Improved Version



Conclusions

- Introduced the notion of well-formed applications, ones that display an appropriate amount of information at any given elevation
- Introduced a system, VIDA, that helps users construct well-formed applications in the DataSplash database visualization environment
- Conducted a pilot study that suggests that information density affects user navigation

Critique

- Strong points
  - The density feedback is effective and informative
  - Interaction with the layer manager is intuitive
- Weak Points
  - Not sure how effective with other density metrics
  - Modification tasks may not be easy when the density metrics and data objects are more complex
  - Semi-automated adjustment of layer density is still in progress , which would better be put in the future work

Where We Are

- A Multi-Scale, Multi-Layer, Translucent Virtual Space Henry Lieberman, IEEE International Conference on Information Visualization, London, September 1997.

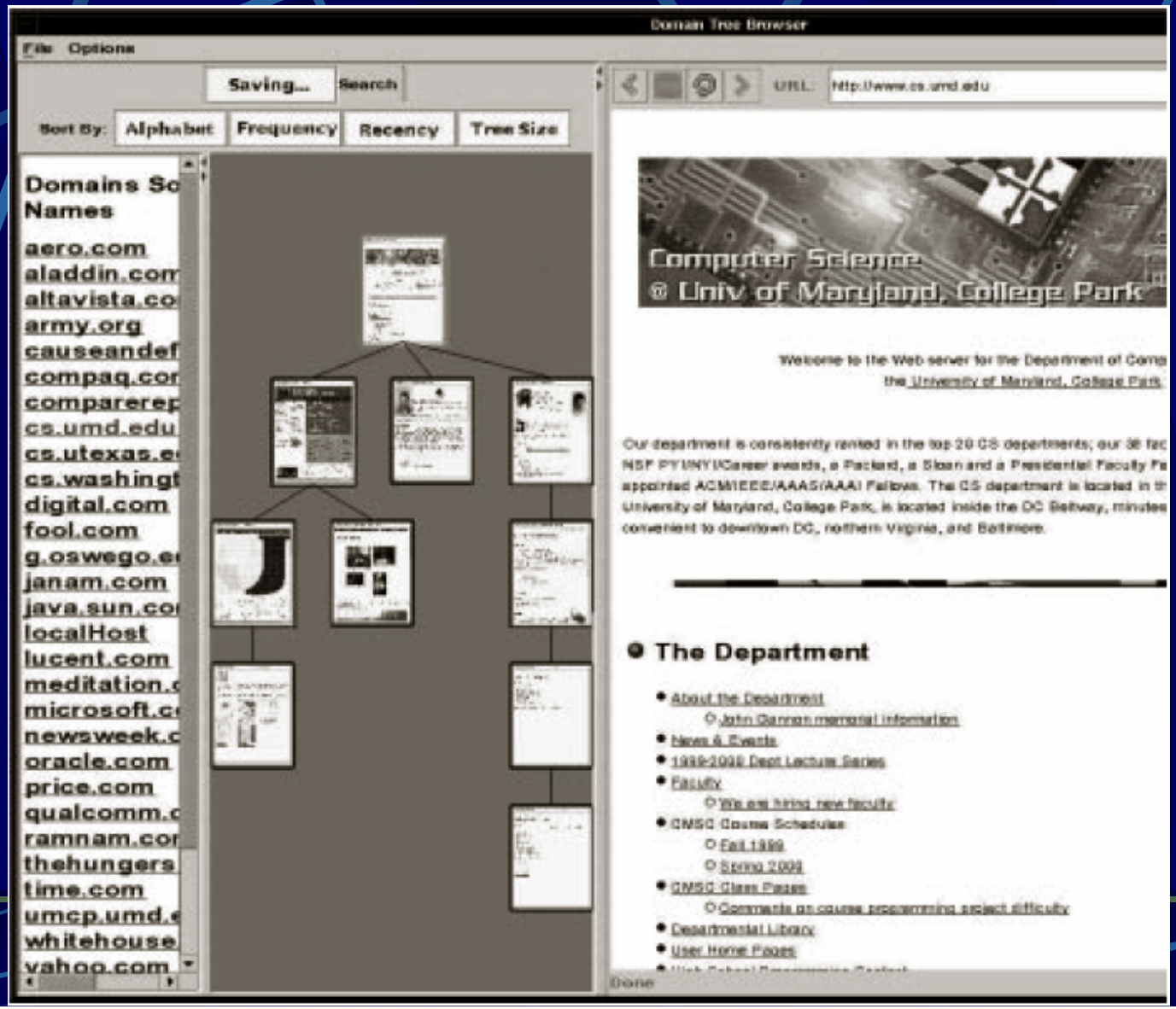
- Constant Information Density in Zoomable Interfaces Allison Woodruff, James Landay, Michael Stonebraker, Proceedings of AVI '98, pp. 57-65.

- Domain Name Based Visualization of Web Histories in a Zoomable User Interface. R. Gandhi, G. Kumar, B. Bederson and B. Shneiderman. *In Proceedings of the Second International Workshop on Web-based Information Visualization (WebVis'00), pages 591-598, Sep. 2000.*

Motivation

- After following a number of links, users often have trouble revisiting a page that was previously viewed
- The history mechanisms in the current browsers are not appealing to users
  - 42% of the pages were revisited using the Back_Button
  - 0.1% of the page accessed used the history list
- The shortcomings of the common history mechanisms are:
  - Whenever a user follows a branch point, a large part of the history is lost
  - The history list is textual and page titles may lack cues needed to find a particular page
  - The history list is cumbersome

The Domain Tree Browser



The Domain Tree Browser

- It constructs a hierarchy as the user traverses the links
- The tool organizes the visited URLs based on web-site domains
- The zoomable user interface automatically resizes thumbnails to fit the window
- Domain Tree Browser (DTB) is divided into two parts
  - The domain panel displays all the domains visited so far
  - The tree panel display the tree visualization of the visited URLs of the domain selected on the domain panel (In a top-down manner)
- A node is a rectangle which contains the screen grab of the web-page it represents

Some Features of DTB

- Color coding is used to indicate the last visited node in the tree
- Size coding on a tree node is used to indicate the number of visits to the corresponding URL
- When a user visit a web-page, and its corresponding domain does not exist, a new domain is added to the domain list and is made current (in red color)

Some features of DTB (cont.)

- All the frame separators are elastic.
- Domain names are searchable
- Users can sort the domain names.
- Users can prune a tree
- DTB provides zooming and centering. Users can also manually zoom in or zoom out of the tree

Pruning Along with Zooming and Centering

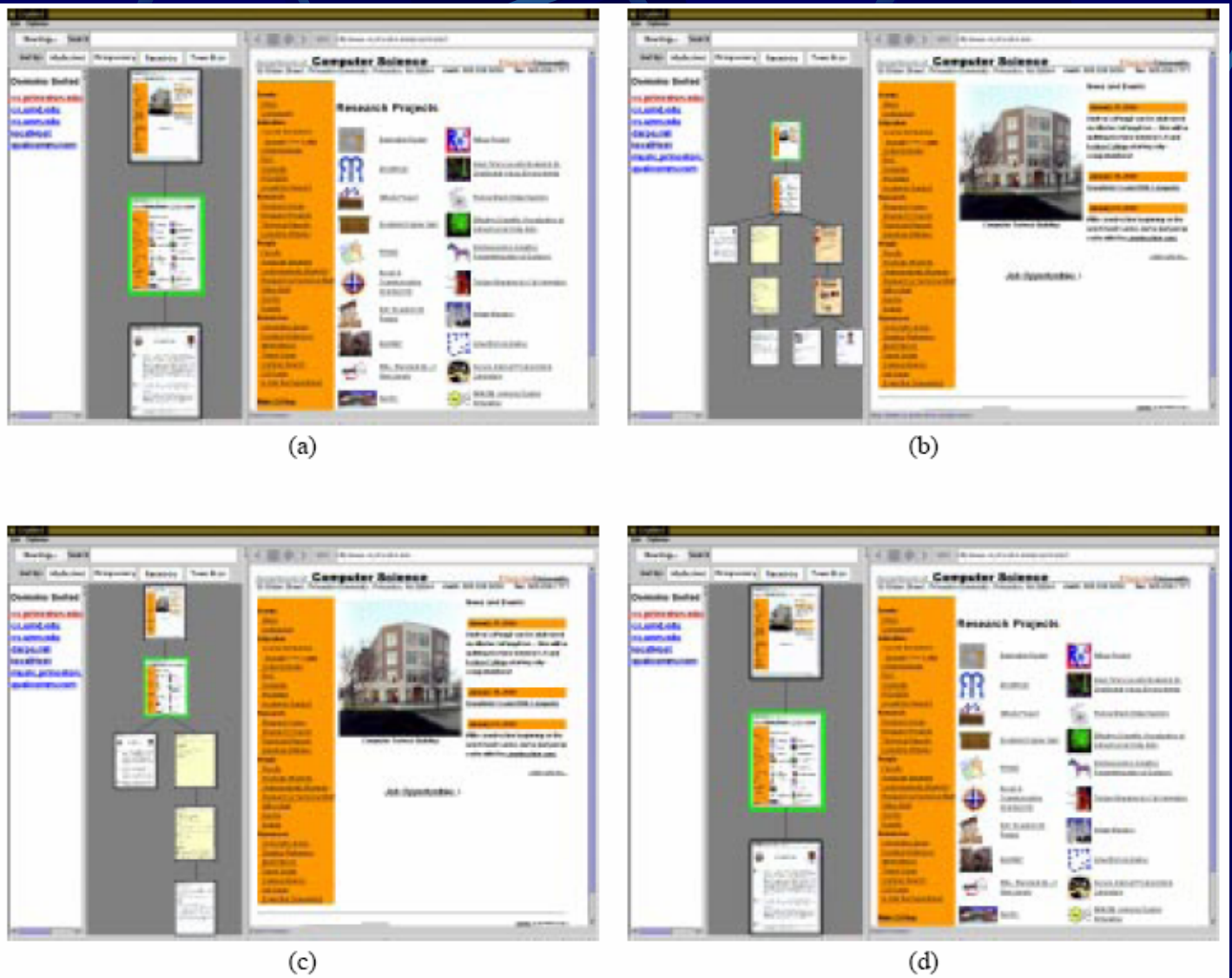


Figure 3. Pruning along with Zooming and Centering. (a) a screen shot of DTB. (b) two subtrees added to the center node of the tree in (a). (c) the resulting tree when the rightmost subtree of the center node is pruned. (d) the rightmost subtree of the center node is again pruned to give us the tree in (a).

Implementation

- Domain Tree Browser is implemented in Java Swing Package and Jazz
- The list of visited domains is maintained using a hashtable
- The tree panel is a Zcanvas (a subclass of Jcomponent in Jazz), which provides zooming and panning capabilities
- The thumbnails are generated by continuously taking the screen grabs of the web browser window

Conclusions

- Organizing URLs by domains and visualize each visited domain is an effective way to visualize history
- The usability study shows that the users took less time with DTB browser to revisit already visited pages
- This was a preliminary study, the utility of DTB need to be enhanced (related to design and interface)

Critique

Strong points

- Zooming and centering is an effective technique for displaying the tree
- Thumbnails provide effective cues for users to find a particular page

Week points

- Scalable? It's hard to find a node if the tree is large
- Lose the relationships among domains