

# Browsing Icons: A Task-Based Approach for a Visual Web History

Matthias Mayer

University of Hamburg  
Computer Science Department  
Vogt-Koelln-Strasse 30  
22527 Hamburg, Germany  
Tel: +49 (40) 428 83-2309  
E-mail: mayer@informatik.uni-hamburg.de  
Visiting Researcher at HCIL

Benjamin B. Bederson

University of Maryland  
Human Computer Interaction Laboratory (HCIL)  
Computer Science Department  
Institute for Advanced Computer Studies  
3171 A.V. Williams Building  
College Park, MD 20742, USA  
Tel: +1 (301) 405-2764  
E-mail: bederson@cs.umd.edu

## ABSTRACT

We have implemented a task and session based visual web history tool called Browsing Icons that dynamically draws animated graphs of the user's paths through the web. Using a proxy, it can be attached to any common web browser. Every web session builds an individual Browsing Graph with a characteristic shape. The graphs are organized in a hierarchy of user-defined tasks. Users can interrupt tasks and continue later using the graphs that provide access to all the web pages they have visited so far. The graphs can be reused for similar or recurrent tasks. By clustering the history hierarchically according to tasks, we try to cope with scale and to provide a powerful concept for easy revisitation. The visualizations have been implemented using the Jazz framework for zoomable user interfaces (ZUIs) in Java.

A usability study showed that users like the system very much and appreciate its support of revisiting web pages. Ten users revisited web pages after a few minutes and other web pages after one to six days using both Netscape with Browsing Icons and Netscape alone. With the tool they completed the short-term revisits in 84% of the time required when using Netscape alone. The revisits after one to six days users could accomplish in 57.8% of the time they needed when using Netscape, thereby visiting only 53.8% as many pages. Users were significantly more satisfied with Browsing Icons than with pure Netscape.

**KEYWORDS:** Web History Visualization, Task Based Information Organization, Information Visualization, Web Browser Usability, Jazz.

## A Browsing Icons Scenario

The following scenario describes how we expect our tool to support people's work on the web. Figure 1 shows the current prototype.

Imagine a university professor, named Susan, preparing one of her summer classes in information visualization. A lot of work for this task will involve the web. In the early phase she

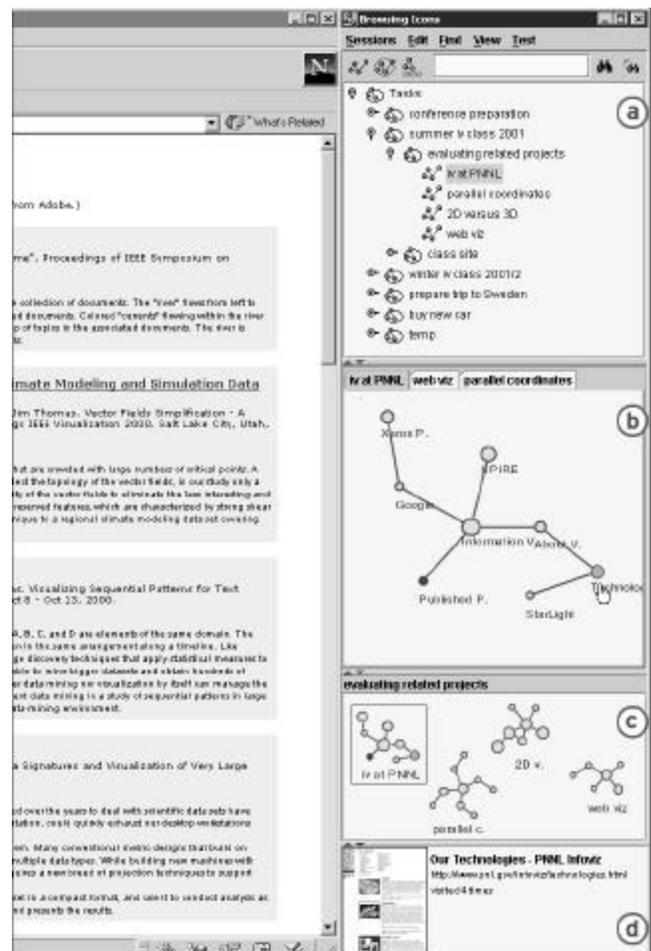


Figure 1: The Browsing Icons user interface placed on the right side of a common web browser (current prototype):

- Tasks and sessions are organized hierarchically in the Task Chooser.
- The Session View shows an interactive graph of the current web session. Layout is animated.
- The Task View shows miniature versions of all session graphs that belong to the current task.
- Hovering over a node shows a thumbnail and further information in the Detail View.

will look for new projects on web pages of her colleagues. She might also look for demos of certain visualizations and order videos of some “classics”. As the class approaches she will prepare the class web site with all the relevant materials. During the semester she will look repeatedly at the web pages the students set up to document their evolving projects. The whole process is often interrupted and is just one of the many tasks in Susan’s job. For example, the professor is also preparing a conference, which includes visiting and revisiting the pages of the keynote speakers. Finally, next winter she gives another class on the same topic. She will have to revisit a lot of material visited while preparing the first class.

Using current web technologies this scenario might work as follows: In the early phase she uses search engines, institutes’ home pages and her existing bookmarks to visit and revisit the web pages of her colleagues. Often, using a search engine seems faster than looking for the appropriate bookmark in her overflowing archive. Sometimes she is not sure if she bookmarked the page at all. For visiting her students’ pages she bookmarks the class’ web page and clicks her way through two of three levels of hierarchy each time. Finally, after half a year she starts the searches over from scratch because she forgot to bookmark some of the important pages when she visited them the first time.

When using Browsing Icons, the tasks could be treated in a more organized and efficient manner. Knowing about the task to come, the professor would create a new task “summer IV class 2001” in the early phase. She would add a subtask “evaluating related projects” and immediately start surfing. Each page she visits is visualized as a node in the session graph displayed in the Session View (Figure 1b). Her path is shown as edges between the nodes. The graph’s layout is automated and animated smoothly.

The subtask about related projects would contain the graphs for all related sessions – all her efforts to find related information on the web would be saved here. The Task View (Figure 1c) shows interactive miniature graphs of all these sessions. In some cases she might decide to continue working on an existing session and to extend its graph. After visiting a lot of sites she uses the Task View to revisit the publication pages of different authors in order to make her readings recommendation list.

For preparing the class’ web site she defines a new subtask “class site” under “summer iv class 2001”. As she creates web pages and navigates them often during this process she creates session graphs that she uses whenever she wants to navigate to one of these pages.

When working on other tasks, such as finding keynote speakers, she simply switches tasks using the Task Chooser. In this way she always has direct access to exactly those references she needs and no more. To view the old data provides contextual information to remember what she had done last and what she wanted to do next. This might also be useful in interruption rich environments.

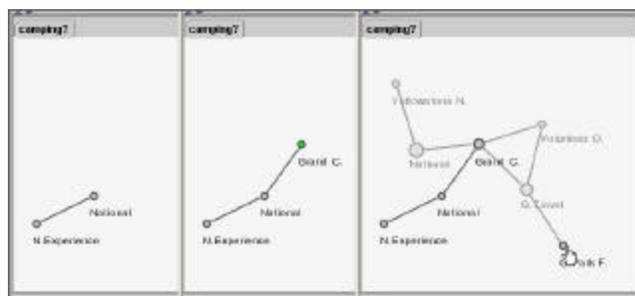


Figure 2: Browse Ahead Graphs: Older graphs appear automatically when browsing in the same “neighborhood”. This could speed up revisits dramatically.

Even half a year later while preparing her new class she can make good use of the tasks previously created. For example, she uses them to have a quick look at her colleagues’ pages again. Therefore, she copies the “evaluating related projects” subtask into her new task. Broken links will be marked automatically. She extends the graphs by following new links or visiting unrelated pages using search engines or the URL entry field.

Imagine our professor would browse across the home page of one of her colleagues in a different context but follow some links exactly as she did when preparing the summer course. After a certain number of identical links the old graph would appear semi-transparently under the current one providing instant access to the pages she visited in the old context. This would work automatically even if she forgot that she ever was there. We call this concept Browse Ahead Graphs (Fig 2.). We think it might be very useful, especially if a certain “fuzziness” factor was applied.

This scenario, while hypothetical, is achievable today with Browsing Icons. Everything described here, except detection of broken links and Browse Ahead Graphs is implemented in our current prototype.

## INTRODUCTION TO WEB HISTORY NAVIGATION

Revisiting web pages is a very common but surprisingly difficult task. In a study done with 23 subjects over six weeks in 1996, Tauscher and Greenberg found that 58% of an individual’s pages are revisits [31]. A more recent study of Cockburn and McKenzie in 2000 with 17 subjects over four months found this number to be 80% ([11]). In the 10<sup>th</sup> GVU WWW User Survey, 16.6% of 3,291 users mentioned not being able to return to an already visited page as one of the biggest problems of web usage [19]. Another 27.6% mentioned not being able to efficiently organize the retrieved information to be one of the biggest problems. In 1996, 37% of 322 users did not organize their bookmarks at all – they just stay in the order they were created [1]. These data provide strong evidence that additional work in supporting revisitation of web pages is needed.

The three primary mechanisms for revisitation in common web browsers are Back Button, Bookmarks or Favorites and the History List. The Back Button provides access to pages just visited while Bookmarks and History Lists can provide

access to more distant pages. Catledge and Pitkow measured a high inequality in usage of the different tools [7]: while embedded links were responsible for 51.9% of all browser interactions, the Back Button still was responsible for 40.6%. XMosaic's Hotlist (like Bookmarks) was used for 2% and the history for just 0.1%. Tauscher and Greenberg found similar numbers in 1997 [31].

All revisitation tools mentioned show serious problems: The stack based behavior of the Back Button makes short term revisits difficult: If you browse from A to B to C back to B to D you cannot revisit C again because C is popped when D is pushed [10]. Improvements such as recency ordered or hierarchical lists have been discussed and studied, showing promising results ([31], [18]). Bookmarks or Favorites for long term revisits require explicit action as well as the ability to know in advance that a certain page will be of interest in the future. Since bookmark lists grow with a fairly constant rate of about one page every five days [1], they are hard to maintain. [11] show that bookmarks contain about 5% duplicates and 25% invalid references. Finally, textual history lists are hard to scan and filter and in the case of Netscape even difficult to access.

Any minor inefficiency in revisitation support will result in a massive productivity loss when multiplied across millions of web users [11].

### **Advanced Web History Visualizations**

Visual web histories can be categorized according to their main focus or contribution. Several projects use 2D visualizations of the spanning tree of the graph built by browsing. Examples are WebMap [13], PadPrints [21] and the Domain Tree Browser [17]. This technique provides immediate visual feedback and the ability to directly revisit recent pages. However, the tree structure might be misleading because it reflects the user's path and not the hierarchy of the visited web sites. Therefore, a homepage can be hidden within a tree while the tree root represents the first page visited which may be less important. Further, a single tree might work for the current session but may grow too large to allow efficient navigation. The Domain Tree Browser tried to solve this problem by providing one tree per server-level domain. This approach introduced another problem when navigation crossed borders between different servers: The new page is attached to the last one visited in this domain's tree even if there is no meaningful relationship between both pages at all. Still both PadPrints and the DTB showed good results in user studies.

Other projects used 3D space in order to layout history information visually. The WebBook introduced spatial bookshelves to organize an individual library of books containing collections of favorite web pages [6], VISVIP visualizes the user's path as a 3D graph [12]. The Cyber-space Visualization Project mapped visited web pages in space according to their servers' directory hierarchies and

showed other users as they accessed the same pages [2]. WebPath introduced meaningful axes to map for example the creation date of a page or the number of outgoing links onto one of three dimensions [16].

Data Mountain [27] combines 2D and 3D approaches. It allows users to position thumbnails of favorite documents on an inclined plane textured with passive landmarks. It follows an interesting approach that takes advantage of human spatial memory skills.

Nestor Navigator [15] and WebNet [10] provide a 2D graph of the user's path. Both systems require manually saving and organizing these maps. They can be reloaded if necessary. Nestor introduces more features like annotations and support for constructing a collaborative space.

The WebView system [18] provides the user with two automatically generated views of thumbnails. One view shows them organized hierarchically according to the storage structure of the related web pages. The other view shows a recency list with duplicates removed. Hovering over them enlarges the thumbnails. They are history-enriched objects [22] in that a small dogear becomes darker as the number of visits increases. Explicitly set red dogears have the function of bookmarks. This approach has been continued where they removed the hierarchical view, but added dynamic query options to the recency list [24]. This enables users to easily view pages with a high visit frequency or even just the explicitly bookmarked ones [3].

A last group of projects focus on history mechanisms for groups of users. Recer [8] recommends certain paths to follow according to a history generated by other users. Similarly, Footprints [34] provides the visitor of a web site with information about other users who also visited the current document and where they came from. A more comprehensive bibliography about web histories and related topics can be found at [26].

### **BROWSING ICONS – SESSIONS AND TASKS**

We implemented a tool called Browsing Icons to aid web history navigation. It is a Java 2 application that can be used with any web browser that supports proxies (Figure 1). Browsing Icons differs from other systems by focusing on graphs of web sessions as basic visual elements and on a hierarchy of tasks on which the user works.

Web sessions are the meaningful units in which somebody uses the web with a more or less specific goal in mind. Our system allows representing and manipulating these units as distinct visual elements. They are highly interactive so that the user can directly revisit any page of the current session. A layout algorithm provides aesthetically pleasing views of the graphs that the user can further modify. We also developed algorithms that partially automate the creation and management of different concurrent sessions.

The task-based organization of sessions is the second main contribution of our approach. We aim at a group of users that both is able to define their tasks and willing to spend some effort to organize these tasks hierarchically. Similar to behavior regarding bookmarks, we expect one group of users that is not frightened by this overhead and therefore is able to gain the advantages of the system and another group that does not spend this effort. Our system is aimed at the former, but will still provide some help to the latter.

A major insight of earlier work is that there is a potential benefit in integrating the functionality of bookmarks, history list, and Back Button in order to reduce the user's need to learn and use the current variety of different systems [11], [30]. Our approach follows this advice and provides one consistent interface for all revisitation and organization aspects. In order to distinguish different kinds of navigational support we defined nine different contexts in which a user can navigate [25].

### Sessions – and Their Recognizable Graphs

When the user launches the web browser and begins to browse the web, a new session is created and visually displayed as a graph in the Session View (Figure 1b). As new web pages are loaded into the browser they are added as nodes to the graph. They are connected to the last visited page by an edge – which represent paths, not necessarily embedded links. A label shows a short version of the page title. The graph is drawn using a spring-based layout algorithm. Whenever changes in the graph structure occur (by a new page or manual reshape) the graph is smoothly animated to its new layout where the underlying physical spring system reaches a minimum of forces (Figure 3). This process results in the graph changing, in a pleasing way and users can easily track the changes of its shape during the animation phase. The graph visualizations are designed to be recognizable by their specific shape. For this reason we used a layout algorithm that tries to minimize edge crossings and reflects symmetry in the graph structure. Manual changes to the graph's shape can create even more unique and therefore more recognizable shapes.

The Session View allows users to directly revisit any pages visited before in this session. Tauscher & Greenberg [31] showed the importance of this ability. With a probability of 39%, the next URL loaded has already been visited within the last 6 pages. Thus, the current Session View will be sufficient for the majority of revisits.

The main visual clue to finding a specific node in order to revisit the corresponding web page should be given by its position within the graph and its graphical properties. However, additional information is provided by the Detail View (Figure 1d) if the user hovers over a node, including the URL, number of visits, and a thumbnail image.

An underlying heuristic decides when to begin a new session. By default, it does so whenever a new browser window is opened manually. We check the HTML source to see if the source itself causes a new window to open in which

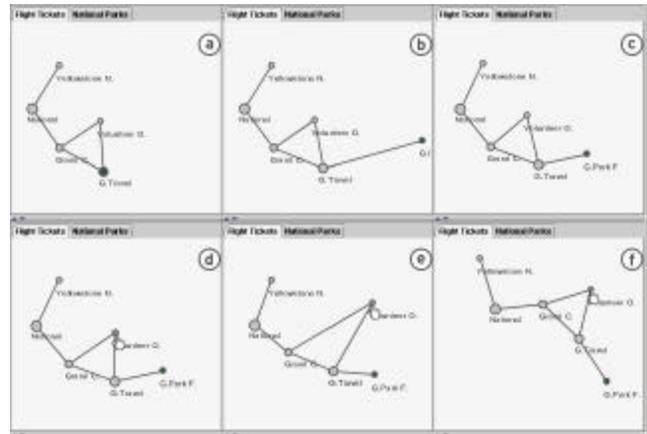


Figure 3: Smooth animation of the graphs using a spring-based layout algorithm.

a-c. When a new web page is visited, its representation flows in from the right.

d-f. The user can reshape the graph manually.

case the old session would be continued. If there is no interaction for 5 minutes, a new session is started. Of course, the user can always start a new session manually.

We define a session to be a series of visits to web pages for accomplishing a specific task. This definition is similar to the notion of “episodes” introduced by [5] for navigation in hypertext systems. The session graph, however, can be re-used in future sessions. The user can choose an existing graph and decide to use it as the basis for the current session, thereby making the session graph (or a copy of the original one) grow.

### Tasks – Modeling the User's Working Context

Our tool focuses on people who use the web primarily for their work and who are able to define certain tasks they want to accomplish. These tasks can be organized hierarchically using the Task Chooser (Figure 1a) by dragging and dropping. A user might use the web several times for accomplishing one task. This would result in several sessions – each of them represented by a visual graph– attached as child nodes to the task they belong to. Tasks can contain subtasks and sessions. Sessions are the leaf nodes in the task tree. All nodes can be renamed manually. The Task View (Figure 1c) provides small versions of the graphs of all sessions that belong to this task. This view shows just the session graphs of the current level, not those of sub- or supertasks.

By organizing web history by tasks, we hope that the user should just see those items that are of current interest. When tasks are switched, the visible and easily accessible items should also change.

This approach fits in to the notion of role management [28]. Shneiderman and Plaisant describe a kind of user interfaces as being role-centered. Their notion of Personal Role Managers focuses on the users' tasks rather than on their documents. The user should be able to enter role specific

working environments. Within each role would be a specific hierarchy of tasks as well as other components – a vision statement, a set of people and a schedule. Current interfaces are still far from being role centered. However, we believe that providing the concept of different tasks a step in the right direction.

We are aware that defining and organizing tasks and sessions add overhead to the user’s work and some users may be unwilling to spend any extra time during web browsing – even if it were to improve their future access. However, we believe that some users are willing to spend time organizing their sessions and tasks (probably the same ones who currently organize their bookmarks), and it is to those users that we are targeting this software. Abrams et al. [1] found that about 23% organize a bookmark at creation-time and 7% at the end of a session – about half of the users organize them sporadically, the rest never. Considering the huge number of internet users (350 million worldwide according to Telcordia™ Internet Sizer, Jan. 2001), this is still a fairly big target group.

### Visual Encoding and Interaction

The visual design rationale for Browsing Icons was to use colors and other visual distinctions that are as subtle as possible, but still clear and effective. This principle is described by Edward Tufte as “The Smallest Effective Difference” [32 p.73]. Color is used to mark the currently viewed page (or more if more browser windows are open), selected pages (which can be deleted or copied) and to give feedback if the mouse hovers above a node. The size of the nodes grows logarithmically with the number of visits to this page. The labels are generated by extracting substrings from the page titles that are as short as possible. The algorithm we developed tries to find unique and meaningful titles. All views are tightly coupled so that interaction with an element will give visual feedback in all views. New sessions and tasks can be generated manually by using the menu- or toolbar or key shortcuts. A search function provides string-based search over all items and all fields.

### Recognizing Sessions - Stellar Constellations Metaphor

We expect the characteristic shape of whole session graphs to allow users to recognize even single nodes fast according to their specific position within the shape. Our visual approach was inspired by people’s ability to read and memorize stellar constellations. We are able to immediately recognize Ursa Major (the big bear) or even locate one of its stars. This would provide big advantages on the Task View level where no labels are visible unless hovered above a node. The user study reported in the next section confirms this expectation.

We expect the uniqueness of shape to be both useful for short term revisits within the current session and for long term revisits after days, weeks or months. In these cases we expect those graphs, which are seen many times to be especially recognizable. This could be the case for example if the user accesses an online API and tutorials during a programming job and visits the same few pages over and

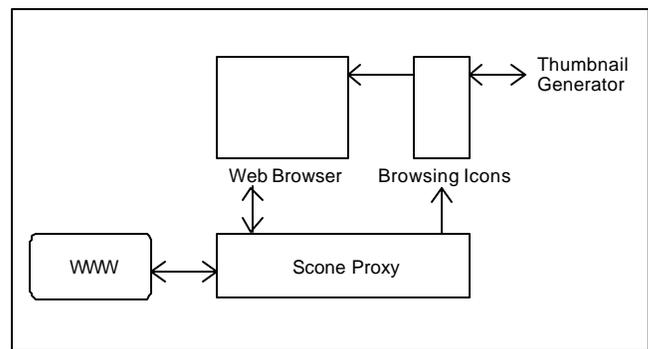


Figure 4: Browsing Icons architecture.

over again. Or if a user builds his own web pages and views them repeatedly during editing. This pattern was observed by [31]. When a task requires several sessions we expect the Task View to be a powerful visualization to provide direct access to a large number of nodes at once.

### IMPLEMENTATION OF THE PROTOTYPE

Browsing Icons is implemented as a Java 2 application using the Jazz framework for zoomable user interfaces [4]. The main architectural parts are the web browser, the Browsing Icons application (BIA), and the proxy.

#### Architecture

Figure 4 shows the high level structure of Browsing Icons. The web browser (any browser that allows to use a proxy) is responsible for displaying the web pages. Whenever a page is loaded, the proxy sends an event to the BIA that updates the visualizations accordingly. When these visualizations are used to load a web page, a special Java class is used to tell the web browser to load this page into the frontmost window. The advantage of the proxy solution is that we can use any web browser we want and that we don’t have to modify it and become dependent on a certain version. The drawback, however, is that we do not get as detailed data as we would like. For example there is no easy way to tell if the user used a bookmark, the URL entry field or a link to load a certain page. Right now we use a heuristic to guess which interaction style was used. Second, we cannot determine the specific window in which a page should be loaded which would be very helpful in an environment with multiple browser windows open. For implementing the thumbnail images we use the HTMLWindow classes [20].

*The Proxy.* We use a modification of the WBI proxy developed by IBM [23]. This proxy toolkit, called Scone, was modified especially for the purpose of providing additional web navigation support [33]. The WBI supports monitors that watch the traffic to and from the web browser. It also allows for defining editors to modify the content of the web page before it is displayed. This would allow us to add Java Script code in order to be able to reconstruct the nesting structure of pages containing frames and to treat them as a single visual element in the BIA.

Scone adds user-tracking functionality to WBI and a database to store all retrieved and extracted information. The

BIA itself is developed as a plugin to Scone which itself is a plugin to WBI.

### **The Use of the Jazz Java2D Graphics Framework**

All graph visualizations are done using Jazz, which is an extensible toolkit for zoomable Java2D Graphics [4]. It allows for manipulating graphical objects in Java on a higher level than the Java2D API. It also provides for panning and zooming functionality and multiple views.

*Layout and Animation.* We wanted to generate unique and recognizable views of sessions since it is important that the user is able to recognize a session even as a miniature in the Task View as fast as possible. The layouts further should be aesthetically pleasing – the numbers of edge crossings should be minimized and symmetries in a graph should be reflected in its shape. Furthermore, the layout process should be smoothly animated so that the user can easily recognize identical objects before and after the transitions.

To meet these goals, we chose an embedded spring algorithm to layout the graphs. It models a physical spring system by calculating forces along the edges between the nodes. This algorithm meets our needs to a high degree. The specific algorithm was taken from the GraphLayout demo that ships with the Java SDK. It is based on a spring-based algorithm as described by [14]. It interprets the graph's edges as springs and the nodes as rings connecting the springs. Starting at some initial configuration the system oscillates until it finds a stable configuration with minimal energy function.

*Persistence.* The BIA allows to save the whole hierarchy of tasks and sessions including the exact graphical shape and placement of all visual elements. For this purpose we use an XML-based file format. The default setting is to automatically save the current state when the application is closed and to reload this state when it is launched again. It is also possible to load and save different states.

### **USABILITY STUDY**

We performed a usability and performance study with ten subjects in April 2001 to learn about one of the two main characteristics of our tool – the visualization of sessions as recognizable graphs and their support for revisitation. This study did not evaluate the task concept since this was not possible in a controlled experiment of this small duration.

The study looked at how users perceive and use our tool compared to Netscape 4.7. In order to compare differences in short and medium term revisits, the subjects revisited pages immediately and then some days after they had visited the pages first.

Our hypotheses were that users would be both faster and visit less pages when revisiting a certain page using our tool as compared to Netscape alone. We also hypothesized that users would be more satisfied using our tool than using the history tools that are available in Netscape. Netscape 4.7 was chosen because it allowed us to easily manage and reload each subject's history for the second meeting.

### **Equipment**

We conducted the study on a 600 MHz Windows NT PC with 512MB RAM and a 21" monitor with 1280 x 1024 pixel resolution. A program presented the questions and also recorded statistics. We recorded the number of visited pages and the time spent on each question. During task execution, users could look on a paper version of each task so they didn't have to switch applications if they forgot a certain part of the task.

### **Stimuli**

Users had to visit and then revisit web pages on the National Park Service web site ([www.nps.gov](http://www.nps.gov)). This site contains pages for all national parks in the United States. They contain images, different sets of links and are organized in a hierarchy.

### **EXPERIMENT 1: SHORT TERM REVISITS**

#### **Method**

Each user had to answer two similar sets of tasks. For one set, they would use Netscape alone. For the other, they would use the Browsing Icons tool aside Netscape. Each set contained four pretasks where users had to visit pages two levels below the NPS homepage. After this they had to revisit these pages in four experimental tasks using the history mechanisms of either Netscape or Browsing Icons. A fifth question requested comparing three pages visited before which were one level below the NPS start page. Example tasks were "Go back to the 'Fort Washington' 'Activities' page. What phone number is listed at the bottom?" or "Compare the NPS home pages of the following three parks. Each of them shows a photo of the park in the upper right corner. How many of the photos show persons on them?". The tasks tried to precisely describe what to look for and how to get there so that the variability due to differences in finding a certain link were small. The first meeting took about one hour for each subject including introduction, training, execution of tasks and answering the questionnaire.

A 2x2 block design was used. This enabled us to collect more information from each subject than if we had used a between-subjects design and to reduce some of the between-subjects variability. The independent variables were whether Browsing Icons or Netscape alone was used first and which set of tasks was used first. Each subject was assigned randomly to one of four groups.

The dependent variables were average time of task completion, average number of pages visited until completion as well as several user satisfaction ratings. Satisfaction was measured using specific questions in the style of QUIS questions (Questionnaire for User Interaction Satisfaction). QUIS was developed at the University of Maryland [29]. Its scale ranges from 1 (negative extreme) to 9 (positive extreme).

Pretasks were not counted because they did not afford any revisits. All subjects answered all questions correctly so we could count all results.

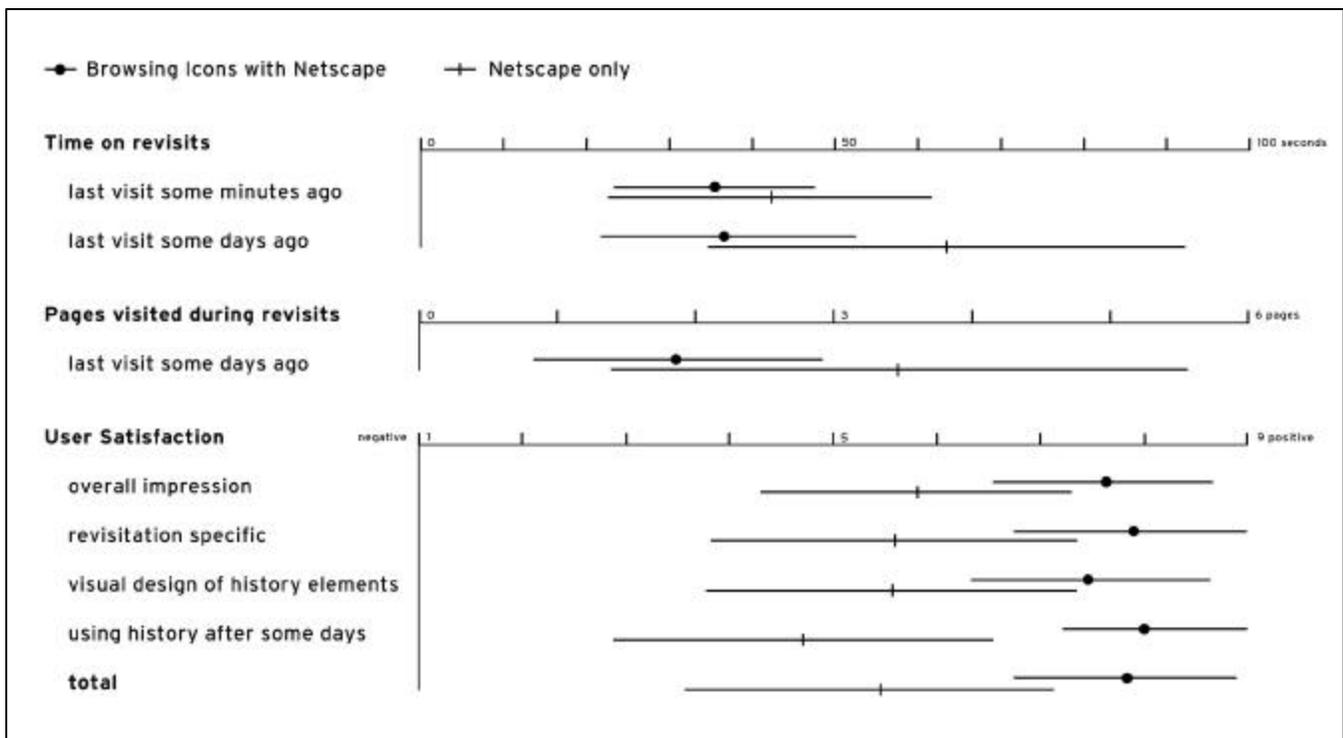


Figure 5: Results of the studies: Users were able to accomplish short term revisits with our tool in 84% of the time they needed when using Netscape alone. For revisits after some days they needed only 57.8% of the time needed when using Netscape. They visited only 53.8% of pages during a revisit compared to Netscape. The user satisfaction questions showed that they clearly preferred Browsing Icons over Netscape (all presented results are significant with  $p < 0.05$ , the graphs show the mean and the range mean-1sd to mean+1sd).

### Training

Each subject was introduced into both systems before executing the tasks. Introduction to Netscape included the demonstration of the Back and Forward Buttons, the manipulation of bookmarks, and the use of the history list. Introduction to Browsing Icons included the demonstration of a short demo session so that the visual behavior of graphs could be seen. A new session was generated and the interface was explained. The text search function was demonstrated shortly. The NPS homepage uses a dropdown menu to select the park of interest at the beginning. All subjects were introduced into the different possibilities to navigate within this list, for example by key shortcuts.

Before using Browsing Icons, subjects had to execute four training tasks in order to get a feeling for the system. They learned how to manipulate graphs, create a new session and how to load a page in the browser. All subjects had high experience with Netscape so we did not have them practice with Netscape alone.

For the Netscape tasks, subjects were motivated to use any history functionality they considered to be appropriate including bookmarks and history list. To keep settings comparable, we asked them to use just one browser window.

The Netscape history was empty to start with, and the bookmarks contained just the hierarchy that Netscape is

delivered with. The personal toolbar was enriched with one bookmark that directly lead to the NPS start page.

### Subjects

Ten subjects participated in this experiment. All were computer science students at the University of Maryland. Six were between 20 and 29 years old, three between 30 and 39, and one between 40 and 49. All have used Netscape before. Five subjects use both Internet Explorer and Netscape, two use Netscape, three Internet Explorer as their primary browser. Four subjects reported using the web between 10 and 19 hours per week, six subjects use it more than 20 hours. Nine subjects majored in computer science, one in information science. Four subjects were female, six male. All subjects were offered payment for their participation. One, however, refused the money.

### Results

We ran paired ttests on the means to calculate results (Figure 5). For all presented results,  $p$  was much lower than  $\alpha = 0.05$ , so all results are statistically significant. In Experiment 1, subjects showed a significantly faster task completion time when using Browsing Icons. The number of pages visited could not be used in this experiment due to an error produced by the proxy, which was fixed for the second experiment. User satisfaction was also significantly higher for Browsing Icons. To quantify user satisfaction, several questions were asked per topic and later compiled to a single

mean value. All results should be interpreted as a first trend since variances are high and the number of participants was relatively small. Variances in task completion times were amplified by changing delays on the network.

## **EXPERIMENT 2: REVISITS AFTER 1 TO 6 DAYS**

We wanted to compare the short-term revisit results to revisits after some days. According to the subjects' schedules, we could arrange a second meeting with each subject one to six days after the first meeting.

Two reasons made us expect different results than in the first experiment: Netscape would have emptied its Go-menu as well as the Back and Forward Buttons (as it does when the application is closed). Browsing Icons provided exactly the same information as before. We still expected a difference because some time has passed since the graphs were generated and seen last.

Each user's Netscape profile was saved after Experiment 1 as well as the Browsing Icons sessions and graphs. In the second meeting, the Netscape profile and Browsing Icons data were loaded again. To provide more realistic circumstances, we added about 150 entries to the Netscape history list, which would represent the pages visited in the meantime. According to [11] the mean number of new pages visited each day is about 42 – therefore this seemed to be an appropriate number. We didn't add entries to the Browsing Icons history since they would have been stored in different tasks within the hierarchy and therefore would make no difference when working again on the first tasks. Of course, this would have added some organizational overhead. Nevertheless, looking at the measured results we think that the total time spent for revisitation with Browsing Icons would still be far below the time using Netscape.

### **Method**

The same method as in experiment 1 was used – each user had to complete 3 revisit tasks and one compare task per system. This meeting took about 20 minutes.

The dependent variables again were the average time of task completion, the average number of pages visited until completion as well as one user satisfaction rating. No training was necessary since all subjects had used both systems before. We told them that we added some pages to their history list in Netscape.

### **Subjects**

Ten subjects participated in this experiment. All of them had participated in experiment 1 before.

### **Results**

The results found in this second experiment exceeded our expectations a great deal (Figure 5). Using Browsing Icons users were almost twice as fast and used only about half the pages during revisits. Regarding user satisfaction, the gap between our tool and Netscape increased. All ten users commented that they liked Browsing Icons more than Netscape for visits after some days.

We see two reasons for these results: first, Netscape lost its "short term memory" while Browsing Icons did not. Second, subjects seem to be able to recognize the session graphs even after some days. Seven users explicitly commented that they recognized the shapes of their graphs.

## **FUTURE WORK**

The conducted studies as well as our own experience provide valuable hints and high motivation for continuing and improving the Browsing Icons project.

Future work can be categorized into four groups: issues regarding further evaluation and improvements in visual design, interaction, and system architecture.

### **Further Evaluation**

We will evaluate the tool in a longitudinal study during summer 2001. This should give deeper insight in how Browsing Icons supports every day work in realistic environments. A small number of users will use the tool for several weeks for their day-to-day work. We will measure quantitative data such as the frequency of use of single history elements. Subjects will be interviewed and asked to answer a questionnaire after this period.

### **Improving the Visualizations**

The visual design has to be carefully reviewed. One issue is to improve the recognizability of the graphs. We are thinking about easy coloring of nodes (personal landmarks), about showing directions at the edges and about visually distinguishing specific node types such as hubs or search results. Another issue is to modify and refine the layout algorithm. A third issue is dynamic web content. For example, pages that generate new results every time they are used like weather forecasts or search engines should be distinguishable and possibly easier accessible than pages with lasting content. For the latter it might be useful to introduce an easy way to save them locally because the page might still be modified or just disappear.

The concept of Browse Ahead Graphs described in the scenario should be implemented soon since we expect it to increase effectiveness to a large degree.

Further visual challenges are improvements of page representations. Will it help to replace the circles by thumbnails or specific logos extracted from web pages? Would it help to show a background image that can be used to place nodes at specific positions? A good starting point for this discussion is provided by [9].

*Nutrient Solutions.* Inspired by the visual and physical metaphor we created the concept of digital nutrient solutions: A special window could act like a petri dish in biological experiments: when the user wants to start a certain search by example query on the internet he would open a specific petri dish window, select certain nodes in the Task or Session View and simply drag them into the dish. This would initiate a search in the background for similar items. Found items would start to grow around the seeds. The query could be modified by giving the nutrient solution specific

properties – for example in order to find just research pages and no commercial ones.

### Improving Interaction

Interaction techniques will be modified and extended. For example, it should be possible to copy or move nodes from one session to another by drag and drop or to generate new sessions by dragging and dropping. Another interesting issue would be to find out if user control of zooming increases the effectiveness.

### Modifying the System Architecture

Regarding the architecture, we are considering eliminating the proxy approach and to implement a browser specific plugin. This would make the distribution and installation of the tool much easier and therefore would help us to conduct the longitudinal study. The plugin approach would also allow getting more detailed data about specific events occurring within the web browser. Thus, collected data could be much more precise and illuminating.

### CONCLUSIONS

Browsing Icons are a new visual approach to web history navigation that is based on fluidly animated and recognizable graphs of web sessions and on a hierarchy of tasks. We have shown that this approach of showing history graphs is highly effective. It reduces the time necessary to revisit a page a few minutes after visiting it and even more a few days after the last visit. It also reduces the number of accessed pages during revisits drastically. Users were able to revisit pages much easier and faster than with the history tools of Netscape. They also liked it more: The results from the questionnaire showed for different areas that users were significantly higher satisfied with Browsing Icons than with Netscape only.

Besides these quantitative results, many subjects commented that they enjoyed the tool a lot. They appreciated its immediate visual feedback for navigation. They mentioned that the graphs' shapes were helpful for revisiting pages directly and that they recognized their individual session graphs even after some days. Also the smooth animation when the graphs' shapes change was said to be helpful. The organization of web related work into single sessions as well as embedding them in a hierarchy of tasks was positively mentioned. Some subjects wanted to keep using the tool for their own browsing.

Browsing Icons is a promising approach to integrate the functionality of Back Button, bookmarks and history list. Its visual interface provides easy and pleasing ways to revisit web pages of interest. The characteristic shape of specific session graphs makes it easy to relocate single web pages in the visualizations. The overview of many web pages in the Task View allows to directly visit all of them using very limited screen space. Chunking web related work into sessions and organizing them hierarchically in different tasks seems to be a good way to scale up with the ever-growing amount of history information over time.

By reducing the amount of cognitive efforts spent on navigation in history Browsing Icons allow individuals to spend more time for their actual work and less for humble-some navigation. Considering the millions of potential users on the web this may lead to dramatic reductions of time and thereby costs.

### ACKNOWLEDGEMENTS

We acknowledge generous support both from the Computer Science Department of the University of Hamburg as well as the Hansische Universitätsstiftung. This work was also supported by the HCIL at the University of Maryland.

We further acknowledge the vital collaboration with our colleagues working on the Scone proxy at the Computer Science Department of the University of Hamburg, Harald Weinreich and Volkert Jürgens as well as the professional and immediate Jazz support provided by Jesse Grosjean and Lance Good at the University of Maryland.

### REFERENCES

- [1] Abrams, D., Baecker, R., & Chignell, M. (1998). Information Archiving With Bookmarks: Personal Web Space Construction and Organization. *In Proceedings of Human Factors in Computing Systems (CHI 98)* ACM Press, pp. 41-48.
- [2] Adler, S., Heise, J., Mayer, M., & Scheffe, P. (1998). Spatial Representations of WWW-Structures to Enhance Navigation, Presentation and Communication. *In Proceedings of Soft Computing, Multimedia and Image Processing (WAC 98) : Vol. 8.* TSI Press, pp. 379-384.
- [3] Ahlberg, C., Williamson, C., & Shneiderman, B. (1992). Dynamic Queries for Information Exploration: An Implementation and Evaluation. *In Proc. of Human Factors in Computing Systems (CHI 92)* ACM Press, pp. 619-626.
- [4] Bederson, B. B., Meyer, J., & Good, L. (2000). Jazz: An Extensible Zoomable User Interface Graphics Toolkit in Java. *In Proceedings of User Interface and Software Technology (UIST 2000)* ACM Press, pp. 171-180.
- [5] Bernstein, M. (1999). Patterns of Hypertext. *In Proc. of Ninth ACM Conference on Hypertext and Hypermedia* ACM Press, pp. 21-29.
- [6] Card, S. K., Robertson, G. G., & York, W. (1996). The Webbook and the Web Forager: An Information Workspace for the World Wide Web. *In Proceedings of Human Factors in Computing Systems (CHI 96)* ACM Press, pp. 111-117.
- [7] Catledge, L. D., & Pitkow, J. E. (1995). Characterizing Browsing Strategies in the World-Wide Web. *(Third Int'l Conference on the World Wide Web)*
- [8] Chalmers, M. (2000). When Cookies Aren't Enough: Tracking and Enriching Web Activity. *Knowledge Politics on the Web* (pp. 99-102). Jan van Eyck Academie Editions.

- [9] Cockburn, A., & Greenberg, S. (1999). Issues of Page Representation and Organisation in Web Browser's Revisitation Tools. *In Proceedings of Australian Conference on Human Computer Interaction (OZCHI 99)*
- [10] Cockburn, A., & Jones, S. (1996). Which Way Now? Analysing and Easing Inadequacies in WWW Navigation. *International Journal of Human-Computer Studies*, 45(1), pp. 105-129.
- [11] Cockburn, A., & McKenzie, B. (2001). What Do Web Users Do? An Empirical Analysis of Web Use. *International Journal of Human-Computer Studies*, (in press).
- [12] Cugini, J., & Scholtz, J. (1999). VISVIP: 3D Visualization of Paths Through Websites. *In Proc. of International Workshop on Web-Based Information Visualization (WebVis 99)*
- [13] Doemel, P. (1994). WebMap - A Graphical Hypertext Navigation Tool. *In Proceedings of Second International WWW Conference*
- [14] Eades, P. (1984). A Heuristic for Graph Drawing. *In Proc. of Congressus Numerantium* pp. 149-160.
- [15] Eklund, J., Sawers, J., & Zeiliger, R. (1999). NESTOR Navigator: A Tool for the Collaborative Construction of Knowledge Through Constructive Navigation. *Proc. of Fifth Australian World Wide Web Conference (AusWeb 99)* Lismore: pp. 396-408.
- [16] Frécon, E., & Smith, G. (1998). WebPath: A Three-Dimensional Web History. *In Proceedings of Information Visualization Symposium (InfoVis 98)* New York: IEEE, pp. 3-10.
- [17] Gandhi, R., Kumar, G., Bederson, B. B., & Shneiderman, B. (2000). Domain Name Based Visualization of Web Histories in a Zoomable User Interface. *Proc. of Eleventh International Workshop on Database and Expert Systems Applications - Second International Workshop on Web-Based Information Visualization (WebVis 2000)* pp. 591-598.
- [18] Greenberg, S., & Cockburn, A. (1999). Getting Back to Back: Alternate Behaviors for a Web Browser's Back Button. (*Fifth Conference on Human factors and the Web*)
- [19] Gvu's WWW Surveying Team. (1998). *Gvu's 10th WWW User Survey: Problems Using the Web*. Georgia Tech, Atlanta, GA.
- [20] Heistermann, H. HTMLWindow (2001). <http://home.earthlink.net/~hheister>.
- [21] Hightower, R. R., Ring, L., Helfman, J., Bederson, B. B., & Hollan, J. D. (1998). Graphical Multiscale Web Histories: A Study of PadPrints. *In Proceedings of ACM Conference on Hypertext (Hypertext 98)* ACM Press, pp. 58-65.
- [22] Hill, W., & Hollan, J. (1993). History-Enriched Digital Objects. *In Proceedings of Computers, Freedom and Privacy (CFP 93)*
- [23] IBM. Web Intermediaries (2001). <http://www.almaden.ibm.com/cs/wbi>.
- [24] Kaasten, S., & Greenberg, S. (2001). Integrating Back, History and Bookmarks in Web Browsers. *In Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 2001)* ACM Press, pp. 379-380.
- [25] Mayer, M. (2000). Kontextvisualisierung: Browsing-Icons und BrowsingGraphs zur Verbesserung der Orientierung und Navigation im World Wide Web. *In Proceedings of 6. Tagung der Deutschen Sektion der Internationalen Gesellschaft für Wissensorganisation Hamburg, Germany*. Similar English version at <http://asi-www.informatik.uni-hamburg.de/personen/mayer/>
- [26] Mayer, M., & Komlodi, A. A Bibliography of Information-Seeking Histories, History Navigation and Visualization (2001). <http://asi-www.informatik.uni-hamburg.de/personen/mayer/historybib/>.
- [27] Robertson, G., Czerwinski, M., Larson, K., Robbins, D. C., Thiel, D., & van Dantzich, M. (1998). Data Mountain: Using Spatial Memory for Document Management. *Proc. of User Interface and Software Technology (UIST 98)* ACM Press, pp. 153-162.
- [28] Shneiderman, B., & Plaisant, C. (1994). The Future of Graphic User Interfaces: Personal Role Managers. *Proceedings of People and Computers IX (HCI94)* Cambridge, UK: CU Press, pp. 3-8.
- [29] Slaughter, L. A., Harper, B. D., & Norman, K. L. (1994). Assessing the Equivalence of Paper and On-Line Versions of the QUI5 5.5. *In Proceedings of 2nd Annual Mid-Atlantic Human Factors Conference* pp. 87-91.
- [30] Tauscher, L., & Greenberg, S. (1996). Design Guidelines for Effective WWW History Mechanisms. *In Proc. of Workshop on Designing for the Web: Empirical Studies* Microsoft Corporation.
- [31] Tauscher, L., & Greenberg, S. (1997). How People Revisit Webpages: Empirical Findings and Implications for the Design of History Systems. *International Journal of Human-Computer Studies*, 47, pp. 97-137.
- [32] Tufte, E. (1997). *Visual Explanations*. Cheshire, CT: Graphics Press.
- [33] Weinreich, H. Scone-Proxy (2001). <http://www.scone.de>.
- [34] Wexelblat, A., & Maes, P. (1999). Footprints: History-Rich Tools for Information Foraging. *In Proceedings of Human Factors in Computing Systems (CHI 99)* ACM Press, pp. 270-277.