Hunter Gatherer: A Collection Making Tool for the Web

m.c. schraefel
Dept. of Computer Science, U of Toronto
Toronto, ON, Canada
mc@dgp.toronto.edu

Yuxiang Zhu
Dept. of Computer Science, U of Toronto
Toronto, ON, Canada
yuxiang@dgp.toronto.edu

Abstract

Task analysis of how users collect information from within Web pages indicates that while capturing information within-Web-page is a common task, it is not a frequent one. Tool support for this interaction is poor: users must move between browsers for copying and editors for pasting content They must also name the components captured and remember to copy and add the URL from the source. These subtasks force users away from their primary focus of information gathering and into information management. Hunter Gatherer is a browser-based tool designed to address the specific problems of forced divided attention in information gathering smaller-than-Web-page sized components.

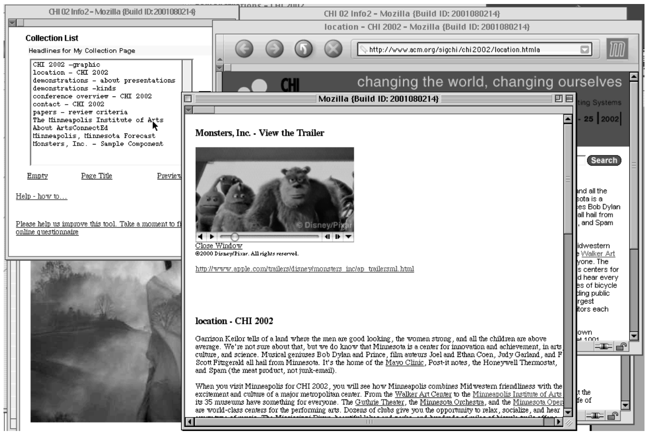
Keywords: Collections, Interaction Design, Hypertext, URLs

INTRODUCTION

Task analysis carried out in a user study reported in [2] indicates that users regularly need to deal with information components from *within* Web pages. The studies found two things: (1) that Web users want to be able to make collections of information found from within Web pages, but that (2) users only infrequently make such collections, in large part because of poor interaction support for this activity. For instance, bookmarks, referencing entire pages, often capture more than the desired data; this forces users first to load and then to sift through multiple pages to attempt to find the desired material. Text editors cause users to shift attention between the information gathering task in the browser and information management task with the editor. In this process, users often forget or neglect to label the collected component with the title or URL of the source page, which degrades the value of the collection over time.

The result of our work to date is Hunter Gatherer (HG) (Figure 1). Hunter Gatherer blends the transparency of bookmark capture for component selection, with the support of an editor for revising collections. The tool also automates the inclusion of a contextual, editable header for each component, and grabs the URL of the source page for that component so that users can return to the source document at any time. By this process, Hunter Gatherer lets users, rather than tools, determine which information activity they wish to focus on: gathering, management or contemplation of the collection.

CHI 2002, April 20-25, 2002, Minneapolis, Minnesota, USA.
ACM 1-58113-454-1/02/0004.


CHI 02 Info2 – Mozilla {Build ID: 2001080214}
CHI 02 Info2 – Mozilla {Build ID: 2001080214}
location – CHI 2002 – Mozilla {Build ID: 2001080214}
http://www.acm.org/sigchi/chi2002/location.htmla
CHI
changing the world, changing ourselves
Mozilla {Build ID: 2001080214}
Search
© Disney/Pixar

Collection List

Headlines for My Collection Page

```
CHI 2002 -graphic
location - CHI 2002
demonstrations - about presentations
demonstrations -kinds
conference overview - CHI 2002
contact - CHI 2002
papers - review criteria
The Minneapolis Institute of Arts
About ArtsConnectEd
Minneapolis, Minnesota Forecast
Monsters, Inc. - Sample Component
```

Empty Page Title Preview

Help - how to...

Please help us improve this tool. Take a moment to fi
online questionnaire

Monsters, Inc. - View the Trailer

Close Window
©2000 Disney/Pixar. All rights reserved.

http://www.apple.com/trailers/disney/monsters_inc/ap_trailersml.html

location - CHI 2002

Garrison Keilor tells of a land where the men are good looking, the women strong, and all the children are above average. We're not sure about that, but we do know that Minnesota is a center for innovation and achievement, in arts, culture, and science. Musical geniuses Bob Dylan and Prince, film auteurs Joel and Ethan Coen, Judy Garland, and F Scott Fitzgerald all hail from Minnesota. It's the home of the Mayo Clinic, Post-it notes, the Honeywell Thermostat, and Spam (the meat product, not junk-email).

When you visit Minneapolis for CHI 2002, you will see how Minneapolis combines Midwestern friendliness with the excitement and culture of a major metropolitan center. From the Walker Art Center to the Minneapolis Institute of Arts its 35 museums have something for everyone. The Guthrie Theater, the Minnesota Orchestra, and the Minnesota Oper are world-class centers for the performing arts. Dozens of clubs give you the opportunity to relax, socialize, and hear

Figure 1. Hunter Gatherer collection (foreground) with text and video components; List/Edit view for monitoring and editing the collection in the background (upper left); surrounding Web pages from which components have been collected.

DESCRIPTION OF THE TOOL

First, Hunter Gatherer is a browser-based, server-side application. By integrating HG with the browser we are able to minimize the forced divided attention introduced by shifting between browser and editor. As a server-side application, HG does not require users to download additional software to access the tool. This approach also lets us support multiple operating systems and browsers simultaneously. Also, our interest is in the potential impact of supporting within-Web-page collection making on Web information practices. Multiple OS support lets us deploy the tool over the widest possible user space.

Second, Hunter Gatherer does not copy data into a collection; it creates references, based on pageURL+LocationWithin ThePage for the components instead. We call this address an Aggregated URL (AURL). Thus, a collection is built as a string of AURLs. Which makes collections highly transportable: an AURL can be treated like a URL, to be bookmarked or emailed. For instance, `<http://[proxy server]/examples/servlet/Collection_b?aurl=http%3a%2f%2fwww%2eacm%2eorg%2fsigchi%2fchi2002%2fpapers%2ehtml%23P%231%231%23papers%20%20CHI%202002%7chttp%3a%2f%2fwww%2eacm%2eorg%2fsig-chi%2fchi2002%2flocation%2ehtml%23P%233%231%23loca-tion%20%20CHI%202002&pagetitle=Chi%20Info>` is a 2 component AURL.

Each user can view and non-destructively edit the collection, since editing only changes an AURL, and one user's changes to an AURL has no impact on someone else's copy. In this respect, Hunter Gatherer embodies a version of Nelson's Transclusions [1]: documents created by references to other documents. By the same referencing mechanism of the AURL, versions of collections can be readily shared while working on a project, for instance by publishing the AURLs in a collaborative Web page (like a WIKI), or sending them via email to the appropriate parties. A more detailed discussion of AURLs is in [3].

Interaction Technique

Selection Interaction

There are two steps to collect a component within a Web page in Hunter Gatherer: (1) select the component to be collected (hold the option key, click and drag over the area to be collected) and (2) with that component selected, press the "a" key. The component is then added to the collection. The user can continue to add components in this manner. Any component that can be displayed in a Web page can be added to a collection, from images to applets.

The selection and add process is relatively transparent. It does not require the user, after adding a component, to shift attention from the browser to an editor application, paste content into that application's file, go back to the browser, copy the URL, go back to the editor, paste the URL, add a note to contextualize the component, save the file, go back to the browser and refocus on hunting for the next component. The user simply identifies a component to be added; the system automatically adds the URL for the source page of the component, and creates a title for each component based on keywords from the component and the title of the source page. In this way, users can focus attention on a task until *they decide* to shift that focus to a different task.

Collection Interaction

On adding a component to a collection, a small window, the List view, opens. This window displays a list of the components in the collection letting the user monitor the growth of a collection. Figure 1 shows the List view partially visible behind the main browser window, making the state of the current collection peripherally available. At any time, the user can bring the List window to the foreground to focus on the collection.

If the user wishes to move task focus from adding components, to the collection, to dealing with the collection itself, they have two ways of doing this: from the List view they can rename a component, sort components in the collection, delete components from the list, give the collection a title and create an AURL to share the collection. Double clicking a title in the List view will also render just that single component. By choosing "preview" in the List view, users can render the collection and edit it directly in that page.

On choosing preview, a new browser window opens, displaying each of the components represented by the List in the order in which they are displayed in the List. Each component when rendered in the Collection view contains the title representing it from the List view and also contains the URL for the original page. At any time, the user can click that link to open the source page for that component, regaining the original context of that component. Likewise, any links within the captured component act like regular Web links: click on them and they will do what they were programmed to do, whether to open a new page or download a tune.

Gradations of Interaction: Focus

Throughout the collection making process with Hunter Gatherer, users can move from information gathering to information management and information sharing. The user can return to the collection at any point either for reference or to edit it further by loading its AURL into a browser's Location field. In other words, the user determines which part of the collection making activity they wish to foreground or keep in the background or have peripherally available at all times.

EVALUATION OVERVIEW

As reported in [2], we evaluated Hunter Gatherer in both a lab experiment and field study. The experiment showed that Hunter Gatherer was significantly more efficient than Microsoft Word for selecting, adding and sorting components into collections. The field study indicated that participants found the tool useful, and that they wished to continue to use the tool after the 4 week trial. Participants in both cases reported ways they would like to see the tool improved – such as being able to remove parts of collected components from within the Collection view – and we have implemented these revisions.

CONCLUSIONS AND FUTURE WORK

The success and ongoing improvements of the tool give us two benefits. First, we have a model for a novel way of interacting transparently with Web pages. Second, the tool that can be widely distributed to a heterogeneous user community in order to evaluate the larger question of affect of tool use on information practice. We plan to deploy Hunter Gatherer in a longitudinal study next to let us consider how being able to treat Web content as smaller-than-page-sized units affects interaction expectations with information resources.

REFERENCES

1. Nelson, T. H.. A Literary Structure with Two Fundamentally Different Means of Connection. http://www.sfc.keio.ac.jp/%7Eted/XUsurvey/xuDation.html.
2. schraefel, m.c. Modjeska, David, Wigdor, Daniel and Yuxiang Zhu. Hunter Gatherer: Interaction Support for Within-Web-page Collection Making. Tech Report, CSRG-437, DCS, U of Toronto, 2001.
3. Zhu, Yuxiang. Web-based Interaction Design and Support for Within-Page Collections. Masters Thesis, Dept. of Computer Science, U of Toronto, 2001.