

An Interactive Visualization for Tabbed Browsing Behavior Analysis

Daniel Cernea^{1,2}, Igor Truderung¹, Andreas Kerren², and Achim Ebert¹

¹Computer Graphics and HCI Group, University of Kaiserslautern,
P.O. Box 3049, D-67653 Kaiserslautern, Germany

²Computer Science Department, ISOVIS Group, Linnaeus University,
Vejdeles Plats 7, SE-35195 Växjö, Sweden

{cernea, i_truder, ebert}@cs.uni-kl.de, andreas.kerren@lnu.se

Abstract. Web browsers are at the core of online user experience, enabling a wide range of Web applications, like communication, games, entertainment, development, etc. Additionally, given the variety and complexity of online-supported tasks, users have started parallelizing and organizing their online browser sessions by employing multiple browser windows and tabs. However, there are few solutions that support analysts and casual users in detecting and extracting patterns from these parallel browsing histories. In this paper we introduce *WebComets*, an interactive visualization for exploring multi-session multi-user parallel browsing logs. After highlighting visual and functional aspects of the system, we introduce a motif-based contextual search for enabling the filtering and comparison of user navigation patterns. We further highlight the functionality of *WebComets* with a use case. Our investigations suggest that parallel browser history visualization can offer better insight into user tabbed browsing behavior and support the recognition of online navigation patterns.

Keywords: Tabbed browsing behavior visualization, parallel browsing history, time series, glyph-based techniques.

1 Introduction

From the beginnings of the Internet, the most important client-side software that users employed to navigate online was the Web browser. However, in the last decades the tasks that users could execute on the Web have greatly increased in complexity, thus influencing the range of features that today's Web browsers incorporate. As a result, users now employ the browser in extremely diverse scenarios, ranging from checking e-mails, chatting and streaming media, to playing games, managing schedules and even developing software [17].

To further support this development many browsers started implementing *tabs* to support parallel browsing. Tabs allow users to access and explore multiple Web pages simultaneously. The importance of tab-based operations can be further supported by the work of Miyata et al. [19], where the presence of foreground and background tasks and their interconnection in the user's mind are

In Computer Vision, Imaging and Computer Graphics - Theory and Applications, volume 458 of Communications in Computer and Information Science (CCIS), pages 69-84, Springer, 2014.

© Springer, 2014. This is the authors' version of the work. It is posted here by permission of Springer for your personal use. Not for redistribution. The original publication is available at www.springerlink.com. Springer Verlag.

emphasized from the perspective of cognitive psychology. As such, Web browser tabs are specifically designed to follow this principle and allow users to distribute their attention based on this model. At the same time, the logging features of Web browsers, called *browser histories*, have the ability to reflect user interests and activity. However, most current browser histories do not consider the additional dimensions introduced by tabbed browsing behavior and fail to capture and adequately represent this parallel navigation information.

In this paper, we address the problem of designing an interactive visualization tool for Internet browser histories that supports intuitive search operations based on content and context information, and that allows the tool user to more quickly find, compare and analyze parallel navigation behavior based on a set of existing—e.g., those described in [14]—and novel metrics. This work extends the research presented in [6].

The remainder of the paper is organized as follows: First, we focus on research that is relevant to the topic at hand and continue with a requirement analysis. This is followed by a detailed discussion of the design decisions and the featured interactions of our proposed visualization. This is further complemented by a use case highlighting how WebComets works in a particular scenario. In order to validate our approach we then describe an evaluation of our tool, and lastly we offer our conclusions.

2 Related Research

Maybe the most common approach for visually encoding browsing histories are tree representations. Tools like MosaicG [11], PadPrints [13], Organic Bookmark Management [21], WebMap [9] and Domain Tree Browser [12] use one or multiple vertical or horizontal 2D trees to represent the domain-structure of the navigated Web sites. In some cases, these tree views are coupled with additional list views that highlight the temporal order of visit, as the tree representations do not reflect the temporal succession of events. Additionally, in many cases screenshots of the Web pages are used as thumbnails embedded in the nodes to support the recognition process [11, 13, 21]. Still, all these approaches represent a Web page only once in the tree, even if it is visited multiple times.

An alternative 2D graph representation focuses on capturing and visualizing the branching events in the navigation path [26]. These visualizations manage to capture the sequential aspect of the browsing process, as each accessed page is drawn as an additional node in the graph. If the user navigates back and accesses a different Web site, the resulting branch will be accordingly represented in the visualization.

A slightly different 2D space-filling solution is offered by the Trails plugin [28] that supports a hierarchical, chronological and group-based representation of the visited pages. Furthermore, it offers a statistical overview of the most often visited Web sites. Another method for representing browser histories is highlighted by solutions that employ one [15] or multiple [7] interconnected linear views that are enhanced by graphical elements (e.g., thumbnails).

Besides 1D and 2D solutions, Web browser histories that employ multiple dimensions or intuitive metaphors have been developed. VISVIP [8] is a 3D representation of a navigation log, where two dimensions are used for drawing the Web site structure, while the third one encodes the temporal information. On the other hand, the combo WebBook and WebForager [5] use the concept of a book to give an overview of the Web sites as well as offer an intuitive information-space for the user.

A special class of browser histories is represented by the statistical summary histories. Tools like SlifeWeb [2], RescueTime [1] or Eyebrowse [17] are mainly focused on time management and analytics, and allow users to generate their own statistic view about how they—or others—navigate the Internet.

However, browser histories are not the only type of data revolving around complex, interconnected temporal events. Other time-series visualizations employing similar visual concepts to our approach include World Lines [25], a visualization technique for exploring the alternative paths of heterogeneous simulation runs, and LeadLine [10], a visual analytics tool for identifying and representing meaningful events in news and social media data.

While diverse and functional, none of these methods focuses on the complex parallel browsing habits of today, where tabs and windows have become means for the user to organize his thoughts, actions and accessed information [14]. The importance of a tool for visualizing, analyzing and comparing parallel browser behavior is further highlighted in [4], since users tend to use multiple windows and tabs as means for backtracking (e.g., users abandon the use of in-browser back operations in favor of opening new tabs and switching between them) and multitasking (e.g., users interact with one tab while Web pages are being loaded and processed in others). Similarly, findings from [22, 27] suggest that users often employ parallel browsing in Web search tasks for reasons like comparing search results, executing multiple queries, interacting with a page while others are being loaded, etc.

3 Requirement Analysis

To support flexible search and analysis efforts, all control and interaction elements that users employ to organize complex and parallel browsing sessions—such as windows / tabs and back-forward operations—need to be recorded and graphically represented in a first step (cp. Subsection 4.1). The captured data organized into a set of different user profiles will be at the core of the later visualization. It has to embed rich meta-information that could be of interest for the user of WebComets, i.e., for researchers in information retrieval, behavioral sciences and related fields (called tool user or analyst in this paper). Following the study described in [14], typical research questions for a better understanding of parallel browsing behavior on the Web are for example: *When and to what extent are users parallel browsing on the Web? What affects parallel browsing behavior during interaction with Web search results?*

Another possible research question is to identify reasons why users revisit pages—for example, because of monitoring pages [3, 16]—and how this is typically done in a multi-tab browser environment. For supporting such studies and for finding answers to such questions, a visualization tool has to offer specific functionalities. We have composed a list of requirements that need to be satisfied by a browser history visualization based on preliminary studies, user feedback about browser histories and their limitations, and information about the nature of parallel browsing behavior from the previously referenced publications. Fundamental requirements for the visualization of the captured data are:

- The visualization should offer an *overview* of the loaded data and *support detailed investigations*. This can be achieved by means of tailored interactions and methods like detail-on-demand.
- The *temporal flow* of the visualized navigation sessions needs to be clearly distinguishable. While most Web browsers only display a chronologically sorted list of the accessed Web pages, it is important that the temporal dependencies are visually and relationally highlighted.
- For each visited Web page, *additional meta-data* needs to be captured like the duration of each visit or the duration of interacting with the Web page. While many solutions already count the number of executed accesses to each Web page, it is important to visualize the temporal sequence of events related to any sequence of navigated pages because documents may be inspected more often and for longer periods.
- *Web site categories*, like search engines or news Web sites, should be introduced and represented. Current browsers support bookmarking of Web pages, which implies saving the Web pages address while at the same time tagging it with the help of keywords or categories, which. These operations increase the retrievability of stored bookmarks and should also be offered by our system.
- The visualization should clearly represent which *navigation path*—or sequence of visited Web pages—the users have followed during their browsing sessions. This includes information about what browser windows and tabs have been opened and closed. Thus, it will be possible to reconstruct the steps that lead to the opening of a particular Web page—a missing feature in many related solutions.

Requirements that improve scalability (with respect to log size and number of user profiles) and analysis possibilities:

- *Visualizing multiple browsing histories* at the same time should be supported in order to allow comparison and analysis operations (e.g., detect if multiple users have similar interests).
- *Connections between similar Web sites* should be emphasized as these might be relevant alternatives in search and analysis tasks.
- Our tool users should be able to *search for particular Web pages* based on content (e.g., title or category) and context information (e.g., pages accessed prior to the one in question). While content-based search is present in all history lists, a search for the context is not supported in most cases.

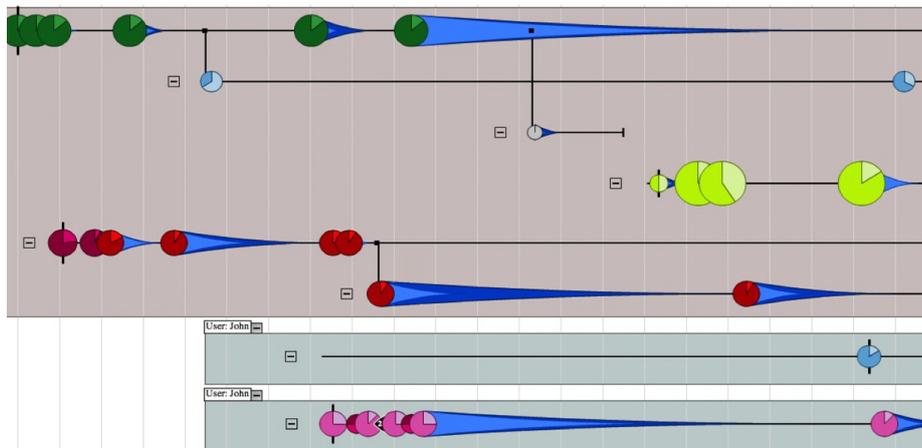


Fig. 1. WebComets visualization of the parallel browsing histories of two users (light-brown and light-grey background). Each horizontal line represents the timeline of a tab that the user has opened, while vertical branches highlight new tabs that have been created by clicking a hyperlink in the parent tab. The comet glyphs encode loaded Web sites and their color coding represents topics. Their position on the time axis depends on the moment when they were accessed.

- Equally important for the analysis of parallel browsing behavior is the *search for navigation patterns*, i.e., finding specific structures (motifs) in the navigation graph which results from branching out from a linear navigation behavior by using tabs or additional browser windows.

4 WebComets

WebComets is a system for the interactive visualization of extended, tab-based browser histories. It was implemented in Adobe Flash ActionScript, with both online and standalone capabilities. The representation and interaction metaphors it incorporates satisfy the requirements highlighted in the previous section. Fig. 1 shows a screenshot of our tool.

To achieve the required functionality and analysis capabilities, WebComets cannot solely rely on information gathered by standard logging systems. For example, browsers like Mozilla Firefox or Internet Explorer do not record the duration for which a user has actively interacted with a Web page. More importantly, browsers do not focus on capturing the parent-child relationships between accessed Web pages and even less the connections between opened tabs or windows. Other researchers have also encountered this difficulty when investigating the parallel browsing behavior of Internet users [14]. The unavailable information included, among others, missing source tabs for branching operations and no information on how a tab or window was created (new tab, hyperlink click).

To address this, we developed a Mozilla Firefox browser add-on with the help of Javascript and libraries like jQuery and Kinetic. The add-on incorporates the

ability of recording and saving an Extended Browser History (EBH) inside an SQLite database. The information collected in this manner can be subsequently accessed and visualized for multiple browsing sessions and users. For any current user profile, the extension saves the navigated URLs together with relevant additional information, representing a combination of a subset of the metrics employed in [14] and a set of additional metrics relevant to the analysis and comparison of parallel browsing habits. Additional details about the EBH data are highlighted in our related publication [6].

4.1 Design

In order to satisfy the requirements highlighted in Section 3, the WebComets visualization has to consider a variety of aspects. Probably the most important is the representation of the temporal dimension and the mapping of the Web pages to a time axis. In order to use the larger width to height ratio of modern screens (widescreen), a visualization concept was devised that maps the timeline to the horizontal axis, from left to right. In this representation, each accessed Web page is displayed as a circle and gets assigned its corresponding position on the timeline (x-axis), cf. Fig. 2.

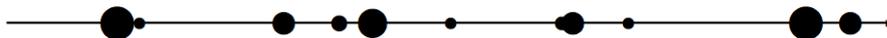


Fig. 2. Conceptual representation of a tab-line that maps the flow of time from the left to the right. The circles on the line represent visited Web pages. Their diameter may represent different attributes.

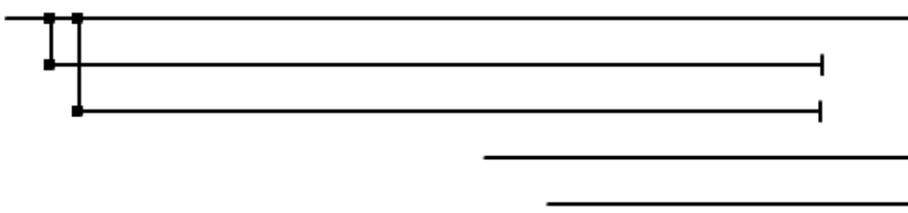


Fig. 3. Conceptual representation of the tab hierarchy. Each horizontal line represents a single tab. Tabs that are connected through a vertical line suggest a parent-child relationship between the tabs and, respectively, between the Web pages loaded in the tabs at that point in time.

The encoding of the parallel navigation that the users are involved in by using multiple browser windows and tabs is another important aspect of the visualization. WebComets represents each browser tab as a separate horizontal

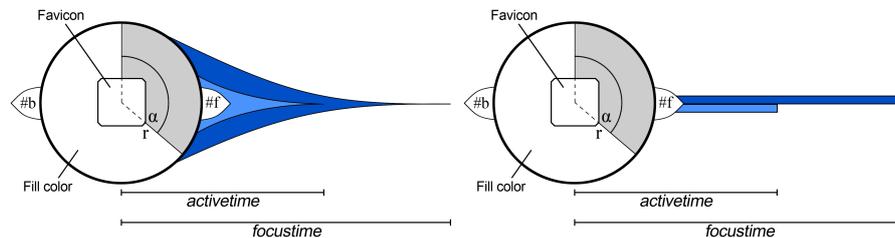


Fig. 4. Circular glyph representation of a visited Web page. The figure highlights two versions of representing the focus and active times of the visited Web pages: as a comet tail (left) or as beams (right). Active time is represented in light-blue, whereas focus time is in dark-blue.

line segment that is parallel to the time axis (Fig. 3). This combination of patches and parallel segments is similar to the representation of a parallel browsing session in [14], as well as to [18] where multiple time-series are visualized through a comparable solution.

As tabs can be opened manually or by clicking a link in another tab, this can result in a tree-like structure that also suggests connections in terms of hyperlinks, but possibly also themes between various Web sites. This parent-child relationship is represented in the visualization as two horizontal lines connected by a vertical one (Fig. 3). At the same time, if the user opens a tab manually, there is no clear way of connecting the first Web page of this tab to any other already opened pages. Therefore, a new tab line is shown as disconnected from the rest of the tabs that were already loaded.

At the same time, multiple opened browser windows are visually encoded as framed rectangular areas, where each rectangle contains a tree-like structure of tabs that reflects the opened tabs in each window during the user session.

Comet-like Glyphs: The representation of a visited Web page has at its core a pie chart enriched with additional graphical elements encoding multiple EBH attributes as shown in Fig. 4. These circular glyphs are mapped on the horizontal axis to the moment in time when the corresponding Web page was loaded, while the vertical positioning identifies the tab in that the Web page was loaded. There are two different visual encodings for the accessed Web pages. While both are based on pie chart representation, there are differences that need to be highlighted.

Encoding 1: The first design focuses on representing the visit counts through the pie charts. More precisely, the radius of each circular representation encodes the number of visits (c_{site}) the user executed in the current session to a particular domain, e.g., *www.google.com*. To correctly visualize this, the maximum visit counts (c_{max}) are computed for each domain in the EBH. Also, tool users have the option of setting and storing the minimum and maximum size for the pie chart radii (r_{min} and r_{max}). Based on these values, the radius of each Web page

circle is computed as:

$$r = \frac{(r_{max} - r_{min})(c_{site} - c_{max})}{c_{max} - 1} + r_{max} \quad (1)$$

As a result, pages that are part of domains that are visited more often will have a larger pie chart than those that have domains that are visited rarely.

At the same time, the pie chart representation is divided into two sectors. The first sector to the right captures the ratio of visit counts for the current Web page compared to the overall visit count for the domain. For example, if the domain *google.com* has been accessed six times in total and the current Web page (*www.google.com/search?q=conference*) only two times, then the pie chart will encode a sector of 1/3. This is computed by the following formula where c_{link} represents the number of accesses executed to the current link:

$$\alpha = \frac{360 \cdot c_{link}}{c_{site}} \quad (2)$$

As mentioned in Subsection 4.1, the EBH includes three time intervals for every Web page: total, focus and active time. Note that the focus time stores the amount of time the Web page was in the foreground and active time captures the duration for which there the user was clearly present and interacting with the Web page, e.g., by mouse movements or key strokes. Therefore, the following relationship is valid for any visited Web page:

$$total_time \geq focus_time \geq active_time \quad (3)$$

In all visual encodings, the total time for a Web page is represented by the horizontal segment between its pie chart representation and the position of the pie chart for the following Web page on the same tab. Because of Eq. 3, the focus and active times are visualized as subsegments of the total time. This is achieved by two representations that resemble a comet tail and a beam that follow the pie chart on the right side (Fig. 4). Both intervals have their origin at the loading time of the current Web page. The length along the timeline is computed by Eq. 4, where t_{enter} and t_{quit} are the timestamps for the begin and end of the session, $x_{rightEdge}$ is the rightmost x position and Δt is the focus time and active time, respectively.

$$l = \frac{x_{rightEdge} \cdot \Delta t}{t_{quit} - t_{enter}} \quad (4)$$



Fig. 5. Example of a Web page glyph employing the first encoding.

Encoding 2: While this encoding is similar to the previously highlighted one, there are two important differences: complexity and attribute mapping. This representation is simpler as not all attributes are encoded in the glyph, and attributes like active time and visit count are neglected. At the same time, the visual elements now encode other attributes of the EBH: the radius of the pie chart is now proportional to the total time of the Web page. This is achieved by using Formula 5 for the radii r , where r_{min} and r_{max} are selected by the tool user, Δt is the total time for the current Web page, and Δt_{min} and Δt_{max} represent the time interval for the shortest and the longest Web page visit.

$$r = \frac{(r_{max} - r_{min})(\Delta t - \Delta t_{max})}{\Delta t_{max} - \Delta t_{min}} + r_{max} \quad (5)$$

Additionally, the focus time is encoded as a sector of the Web page pie chart, where:

$$\alpha = \frac{360 \cdot focustime}{\Delta t} \quad (6)$$

This approach eliminates the need for a comet tail or beam representation, thus freeing up screen space and simplifying the visualization at the cost of removing information. The tool users have the possibility to seamlessly switch between the two encodings in the WebComets visualization.

Further glyph features, like category colors, favicons and back-forward operations, have been extensively highlighted in [6] and will not be detailed in this paper.

4.2 Interaction

Besides the already presented features of the approach, tool users have additional possibilities to customize and interact with WebComets. The current configuration can be stored inside an option window and includes—but is not limited to—the following: switching between comet tail and beam representations, customizing min and max values for the pie chart radii, enabling or disabling elements like favicons or back-forward arrows, and selecting the EBH attributes to be displayed.

Visual Exploration: A set of interaction metaphors complement the visualization’s abilities by addressing topics like flexibility, scalability or detail-on-demand. The WebComets interface supports pan and zoom operations, similar to modern interactive maps. While the panning operation is self-evident, there are two zooming approaches implemented in the tool: one is a regular 2D zoom that allows tool users to inspect the details in a certain area, while the other is a 1D horizontal zoom along the timeline that stretches the horizontal axis.

When a Web page is selected, WebComets searches the browser history to check for similar Web pages. By default, only pages with the same domain are considered to be similar, but more complicated rules for interconnection can be generated, e.g. pages containing a keyword, pages that have an active time of

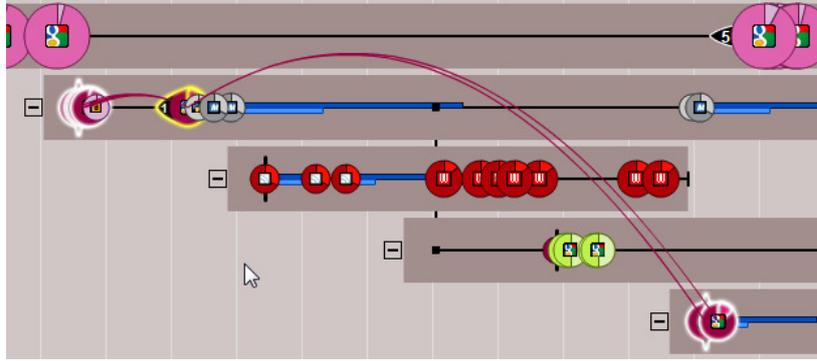


Fig. 6. Connections between the selected Web pages and other glyphs highlighted through continuous curved lines.

similar length, etc. These are highlighted by adding links between each selected element and its counterpart. The curved lines (Fig. 6) are used for showing the presence of similar Web pages to the selected ones, possibly in areas of the visualization that are not currently visible. Curves can be easily perceived as they contrast with the overall orthogonal representation of our approach. To avoid clutter, curves are drawn in such a way that the probabilities of intersecting curves and a curve intersecting a glyph are reduced.

It might occur that multiple glyphs are partially or almost totally overlapping. Even if the tool user has the possibility to execute timeline zoom commands to compensate for this and clearly separate the overlapping glyphs, this is a vital scalability issue. To deal with this, WebComets displays partially overlapping pie charts by positioning the glyph of the Web page that has been accessed later over the top of the previous one. To further compensate, the tool user can move the mouse pointer over a set of densely grouped circles. By doing so, the glyph with the center closest to the pointer will be moved to the foreground.

Content and Motif-based Filtering: Highlighting different elements in a browser history is closely coupled with searching for Web pages or navigation patterns. Most Web browsers support a text-based search of their records that limits their ability to detect context information. This functionality is also supported by WebComets. Tool users can search for terms and keyword combinations (e.g., *apple+pc*), strict phrases (by using quotes, e.g., *"apple pc"*) or even exclude words from their query (by using the minus sign, e.g., *apple-pc*).

Besides the possibility to execute a text-based search of the extended browser history's informational content, supporting the detection, analysis and comparison process of temporal patterns in the navigation graph requires a different approach. Thus, WebComets offers a motif search window (Fig. 7) that allows users to define, store and search for custom information and patterns of navigation. Inspired by the building blocks concept in [20, 23], motifs in WebComets are predefined subgraph structures that can be used to filter the current history.

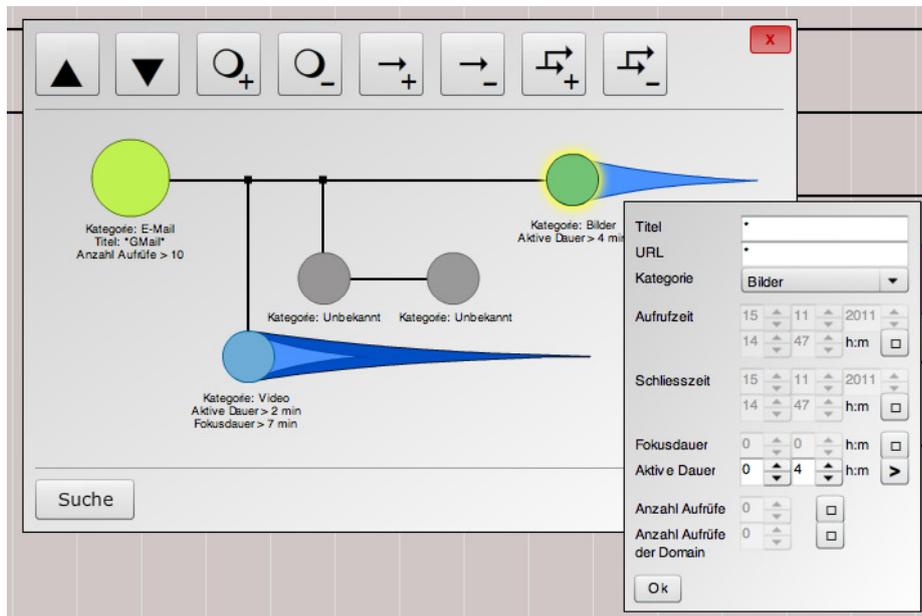


Fig. 7. The motif window that helps users construct, save and search for custom motifs based on Web page attributes and context information.

Compared to other approaches, the WebComets motif search has the advantages of allowing logical combinations of sought patterns (e.g. find all node groups that satisfy motif A and do not satisfy motif B), as well as the possibility to model a large set of node and internode attributes.

These subgraphs can be generated in two ways: either by mining substructures from the currently opened history log files or by manually defining a motif and its corresponding rules. In the first case, the tool user would look through the visualized browser histories and select any subset of glyphs that could be at the core of a parallel browsing behavior. When all relevant elements are selected, the generated motif can be edited further in the motif window to generalize or particularize the final structural pattern. Contrary to this, in the second approach the analyst would start generating a motif by directly building it in the motif window, adding node after node and customizing their attributes based on his experience and assumptions.

As such, tool users analyzing an EBH can not only look for topics of the visited pages, but also detect navigation motifs. For example, the motif search could detect that in 72% of the cases when an Internet user accesses his e-mail account, he also opens a Web page from the category video in a new tab originating from his e-mail page. This might suggest that he receives much e-mail with links to video content. Thus, filtering the browser history based on structural aspects can have many applications, for example, detecting a Web page where the tool user knows some attributes of the originating site or investigating similar interest and patterns of navigation between multiple users.

It is also possible to filter out numerical and temporal values by giving exact numbers, suggesting min or max thresholds, or defining intervals. These rules are then incorporated in the motif and displayed under the corresponding pages. In terms of structure, complex motifs can be built by adding multiple Web pages and highlighting existing relationships between them, be it on the same tab/window or on different navigation branches. Once the motif specification is finished, the analyst executes the search operation and the sum of all the rules will be used for filtering the history. Finally, the nodes that fit the query will be highlighted as already described.

5 Use Case Scenario

In the following, we will highlight a possible use case for the WebComets tool focused mainly on multi-session single-user tabbed browsing behavior analysis. Thomas is a freelance programmer who develops Web-based applications. As such, he is involved not only with programming, but also with quality management and even customer support for some highly specialized modules. His attention is currently focused on an application he is developing in a programming team as well as answering written queries from former customers. For all these operations, Thomas uses mainly online tools: online chat rooms, online documentation, and even an online PHP editor inside his Web browser. Additionally, in order to have a close collaboration with his team, he spends some time on *fonie.de*, where he has periodical online conferences about implementation ideas and concept ambiguities.

After some workdays, Thomas notices that while he spends more than his daily eight hours in front of the computer, he still does not manage to finalize his tasks during the designated timeframe. He is aware that he takes short breaks now and then from writing code in order to inspect documentation, read e-mails, or simply to disconnect for a few minutes. Still, he does not believe that these interruptions are the sole reason for his reduced efficiency. Thus, he decides to use WebComets to examine the extended browser history for the last nine hours he was working (Fig. 8).

Initially, Thomas is trying to obtain an overview and recognizes that he has been distracted multiple times on the day in question. At the beginning of the day, he opened the online PHP editor and accessed his project files to continue coding. Immediately after, a conference call interrupted his work and lasted over an hour, much longer than initially intended (Fig. 9). Furthermore, Thomas notices that he read multiple PHP articles before noon that also resulted in a couple of visits to news Web sites. After the lunch break, he accessed his project files again and continued programming. This time, a piece of information about PHP distracted him and he ended up watching a set of YouTube videos related to coding practices and later entertainment. Soon afterwards, he had to participate in an online conference with a customer. Towards the end of the day, he inspected a set of Web pages with PHP articles and documentation.

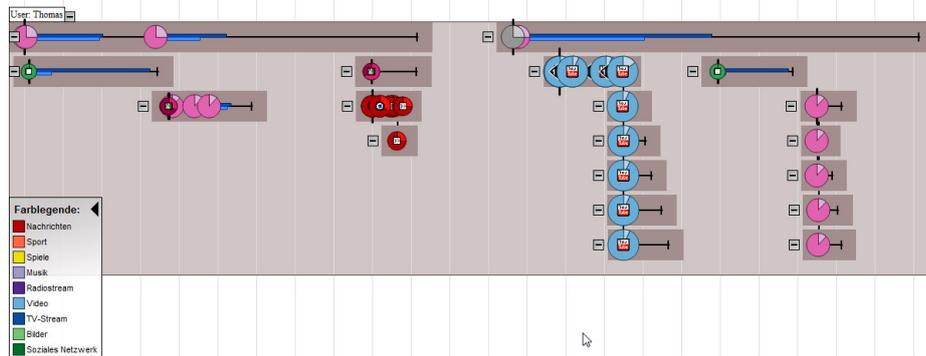


Fig. 8. WebComets representation of Thomas' browser history for today.

Thomas wants to find out exactly how his navigation time is distributed over the various Web sites, in terms of actual time investment. Also, he would like to know how much percent of his working time actually flowed into the project development. From the times of the navigated Web pages, he is only interested in the focus time, because only during these periods the pages were in the foreground. To find out how much time he spent on particular Web pages, he selects the corresponding glyphs. He can inspect an information box for additional overview or detail information. By executing a mouse over operation in the left-top corner of the box, he finds the convoluted values for the focus times on each domain (Fig. 10 (left)).

Thomas notices that he has worked on the project only 48% of the eight hours he was supposed to. Instead, approximately one quarter of the time he has spent speaking with colleagues or customers. Furthermore, he has spent 13% of his time watching videos and another 10% on inspecting PHP documentation. To illustrate the result again, Thomas selects all pages by executing a search without search terms. He then sorts the list of all pages presented in the information box based on the focus time (Fig. 10 (right)). He now can see a clear distribution of the times spent on various Web sites and Web pages.

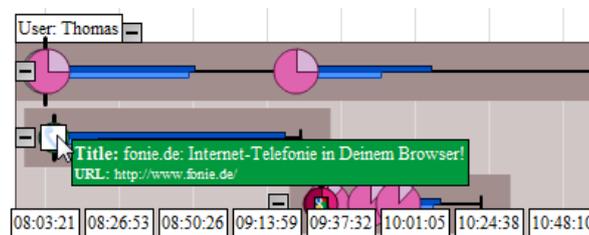


Fig. 9. Conference call executed by Thomas in the morning.

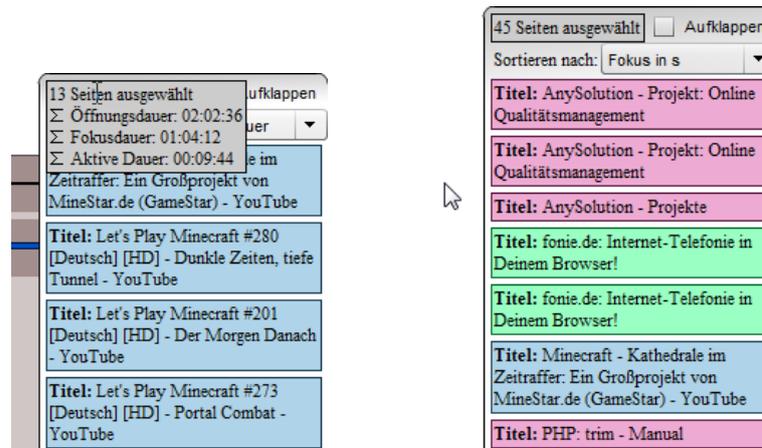


Fig. 10. Information box with the summed durations (open, focus and active time) for every accessed site (left). List of accessed Web pages sorted by focus time (right).

6 Evaluation

A brief evaluation of the WebComets visualization approach has been executed in order to capture advantages and disadvantages. The aim of the study was to compare the performance and accuracy of tool users when inspecting and comparing patterns in multiple parallel browsing histories. For this purpose, the participants would interact with the same EBH log files by two different approaches: the WebComets visualization and a list-based browser history.

The evaluation involved 20 participants with experience in knowledge exploration as well as with extensive background in using diverse Web browsers and accessing a variety of online applications. An initial step in the evaluation process was to randomly divide the participants into two groups and attribute a task to them. Each member of the first group would have to solve the task using the WebComets visualization, while the members of the second group used a list-based history.

The scenario involved the analysis of two browsing sessions from different users. An initial assumption was made that the users have participated at an online conference call where they suggested relevant Web sites to each other. This collaborative browsing approach is frequently used in cases where one party tries to highlight some information to the other. A simple example for this would be the collaboration between two students that are preparing for an exam at remote locations. As a result, both users would access multiple similar or identical Web pages in the same time interval. The test persons had to determine if the initial assumption of collaboration is supported by the browser histories, and if so, what Web pages might have been involved and in what time interval this collaboration took place.

The results were evaluated by inspecting the total time each participant took to find a solution and the time frame he/she reported as part of the conference

session. On average, users managed to find a solution more than twice as fast with the WebComets visualization than with the text-based representation of the navigation history. Further, the subjects that used the WebComets tool have identified the correct time frame in almost 100% of the cases, while the group using the list-based browser history has reported a lower success rate with only a 71% average overlap of the detected collaboration time frame. Based on these findings and the results of the post-task questionnaire given to the participants, WebComets seems to address the issues raised by our requirement analysis. Additional details of the evaluation are presented in [6].

7 Conclusions

In this paper we presented WebComets, an interactive visualization for tabbed browser histories. After discussing related work and functional requirements, we focused on the interactive features and filtering capabilities of our tool. WebComets allows its users to more efficiently search for patterns in parallel browsing sessions by means of motif-based filtering as well as compare and analyze the tabbed browsing behavior of Internet users. This functionality is further highlighted by a use case. Finally, an evaluation confirmed that our approach has met the initial requirements, and our users were able to quickly filter navigational information and detect patterns in the online sessions they were exploring.

References

1. Rescuetime. <http://www.rescuetime.com> (2013)
2. Slife labs time management software. <http://www.slifeweb.com> (2013)
3. Adar, E., Teevan, J., Dumais, S.T.: Large Scale Analysis of Web Revisitation Patterns. In: 26th Annual SIGCHI Conference on Human Factors in Computing Systems, CHI 08, pp. 1197–1206, New York, NY, USA, ACM (2008)
4. Aula, A., Jhaveri, N., Kaki, M.: Information Search and Reaccess Strategies of Experienced Web Users. In: 14th International Conference on World Wide Web (WWW 05), pp. 583–592 (2005)
5. Card, S.K., Robertson, G.G., York, W.: The Webbook and the Web Forager: An Information Workspace for the World Wide Web. In: Human Factors in Computing Systems (CHI 96), pp. 111–117 (1996)
6. Cernea, D., Truderung, I., Kerren, A., Ebert, A.: WebComets: A Tab-Oriented Approach for Browser History Visualization. In: Conference on Information Visualization Theory and Applications (IVAPP), pp. 439–450, SciTePress (2013)
7. Cockburn, A., Greenberg, S., Jones, S., McKenzie, B., Moyle, M.: Improving Web Page Revisitation: Analysis, Design and Evaluation. In: IT & Society 1(3), pp. 159–183 (2003)
8. Cugini, J., Scholtz, J.: VISVIP: 3D Visualization of Paths through Websites. In: International Workshop on Web-Based Information Visualization (WebVis 99), Florence, Italy, pp. 259–263 (1999)
9. Doemel, P.: Webmap - A Graphical Hypertext Navigation Tool. In: 2nd International World Wide Web Conference, vol. 28, issue 1–2, pp. 85–97 (1995)

10. Dou, W., Wang, X., Skau, D., Ribarsky, W., Zhou, M.X.: LeadLine: Interactive Visual Analysis of Text Data through Event Identification and Exploration. In: IEEE Conf. on Visual Analytics Science and Technology (VAST), pp. 93–102 (2012)
11. Eric, M., Ayers, E.Z., Stasko, J.T.: Using Graphic History in Browsing the World Wide Web. In: International WWW Conference, pp. 1–7 (1995)
12. Gandhi, R., Girish, K., Bederson, B.B., Shneiderman, B.: Domain Name Based Visualization of Web Histories in a Zoomable User Interface. In: International Workshop on Database and Expert Systems Applications (DEXA), pp. 591–598 (2000)
13. Hightower, R.R., Ring, L.T., Helfman, J.I., Bederson, B.B., Hollan, J.D.: Graphical Multiscale Web Histories: A Study of Padprints. In: 11th Annual ACM Symposium on User Interface Software and Technology, pp. 58–65 (1998)
14. Huang, J., White, R.W.: Parallel Browsing Behavior on the Web. In: 21st ACM Conference on Hypertext and Hypermedia (HT), pp. 13–18 (2010)
15. Kaasten, S., Greenberg, S.: Integrating Back, History and Bookmarks in Web Browsers. In: Extended Abstracts of ACM Conference of Human Factors in Computing Systems (CHI), ACM Press, pp. 1–2 (2000)
16. Kellar, M., Watters, C., Inkpen, K.M.: An Exploration of Web-based Monitoring: Implications for Design. In: SIGCHI Conference on Human Factors in Computing Systems, CHI 07, pp. 377–386, New York, NY, USA, ACM (2007)
17. Kleek, M.V., Xu, C., Moore, B., Karger, D.R.: Eyebrowse: Real-time Web Activity Sharing and Visualization. In: CHI 2010 Extended Abstracts on Human Factors in Computing Systems (CHI EA), New York, USA, pp. 3643–3648, ACM (2010)
18. Krstajic, M., Bertini, E., Keim, D.: Cloudlines: Compact Display of Event Episodes in Multiple Time-series. In: Visualization and Computer Graphics, IEEE Transactions on, vol. 17(12), pp. 2432–2439 (2011)
19. Miyata, Y., Norman, D.: Psychological Issues in Support of Multiple Activities. In: User Centered System Design, D.A. Norman and S.W. Draper (Eds.), pp. 265–284 (1986)
20. Reeder, J., Reeder, J., Giegerich, R.: Locomotif: From Graphical Motif Description to RNA Motif Search in Bioinformatics. In: Bioinf., vol. 23(13), pp. 392–400 (2007)
21. Shen, S.T., Prior, S.D., Chen, K.M.: A Solution to Revisitation using Organic Bookmark Management. In: Design, User Experience, and Usability. Theory, Methods, Tools and Practice: First International Conference, pp. 46–52 (2011)
22. Spink, A., Park, M., Jansen, B.J., Pedersen, J.: Multitasking during Web Search Sessions. In: Inf. Process. Manage., vol. 42(1), pp. 264–275 (2006)
23. von Landesberger, T., Rehner, R., Gorner, M., Schreck, T.: A System for Interactive Visual Analysis of Large Graphs using Motifs in Graph Editing and Aggregation. In: Vision Modeling Visualization Workshop (VMV), pp. 331–340 (2009)
24. Ware, C.: Information Visualization (second edition), Perception for Design. Elsevier Inc. (2004)
25. Waser, J., Fuchs, R., Ribicic, H., Schindler, B., Blschl, G., Grrler, E.: World Lines. In: Visualization and Computer Graphics, IEEE Transactions on, vol. 16(6), pp. 1458–1467 (2010)
26. Waterson, S., Hong, J.I., Sohn, T., Heer, J., Matthews, T., Landay, J.: What Did They Do? Understanding Clickstreams with the Webquilt Visualization System. In: Advanced Visual Interfaces, pp. 94–102 (2002)
27. Weinreich, H., Obendorf, H., Herder, E., Mayer, M.: Off the Beaten Tracks: Exploring Three Aspects of Web Navigation. In: 15th International Conference on World Wide Web (WWW 06), pp. 133–142 (2006)
28. Yu, W., Ingalls, T.: Trails - An Interactive Web History Visualization and Tagging Tool. In: HCI (10), pp. 77–86 (2011)