A Heatmap-Based Visualization for Navigation Within Large Web Pages

Richard Atterer
University of Munich, Media Informatics Group
Amalienstr. 17, 80333 Munich, Germany
richard.atterer@ifi.lmu.de

Philip Lorenzi
University of Munich, Media Informatics Group
Amalienstr. 17, 80333 Munich, Germany
lorenzi@cip.ifi.lmu.de

ABSTRACT

When a web browser is used to read documents which consist of multiple long pages, such as technical documentation, today's browsers only offer inadequate support for users to orient themselves within these pages. Even if a table of contents is present, no information about the size of individual sections is available. Furthermore, when jumping to different parts of a document, there is no way to find sections that have been visited earlier – the browser's history functionality only works on the level of URLs, not within pages. In this paper, we introduce a tool that increases users' awareness of the organization of long HTML pages and visualizes their own navigation movements within these pages. Our JavaScript prototype uses a simple user interface concept which concentrates on automatic collection of information. It visualizes section sizes in a table of contents which is generated for all pages. Additionally, a heatmap highlights those parts of each page which have been viewed for extended amounts of time. In a user study, our concept is compared to the predominant existing in-page navigation aid, a fixed table of contents at the top of the page.

Keywords

WWW, navigation, long web pages, heatmap, history visualization

Categories and Subject Descriptors

H.5.4 [**Information Systems**]: Information Interfaces and Presentation—*Hypertext/Hypermedia*; H.5.2 [**Information Systems**]: Information Interfaces and Presentation—*User Interfaces*

1. INTRODUCTION

To this day, many users still prefer printing out long documents to reading them on their computer. Mostly, this is attributed to the better readability of text on printed paper, but printed documents also have other advantages, such as being easier to annotate. Furthermore, the physical copy allows the users to orient themselves more easily in the large amount of text: The number of sheets of paper per section reveals the approximate size of the section. If the reader has looked for information in several parts of a specification and then wants to go back to one of them, he will be able to find it relatively quickly (even in the absence of a bookmark) by browsing through the pages and recalling the placement of the information on its page (e.g. "right column near bottom, above a picture").

On the web, many of these advantages are not present. Authors of technical documentation and other long texts typically split their work into sections and create one web page per section in an attempt to avoid confusingly large web pages as well as an excessive number of pages. Still, this results in long pages which require a lot of scrolling. Unfortunately, current browsers only provide navigation and orientation support at the URL level (via bookmarks and the history, or with multiple tabs), so the in-page navigation must be done manually. It is possible to approximately remember the scroll offset of information on pages, but this is likely to become more difficult the longer the pages get. Furthermore, not only the scroll offset must be remembered, but also the page it applies to.

In this work, we present a concept to help the readers of a web document with long pages to find relevant information on the pages more quickly. Rather than requiring explicit actions (e.g. creation of bookmarks) from the users, we rely on the visualization of data that is created implicitly by them or the document authors. Our prototype is called WebpageMap and consists of two major UI components which make navigation within pages easier:

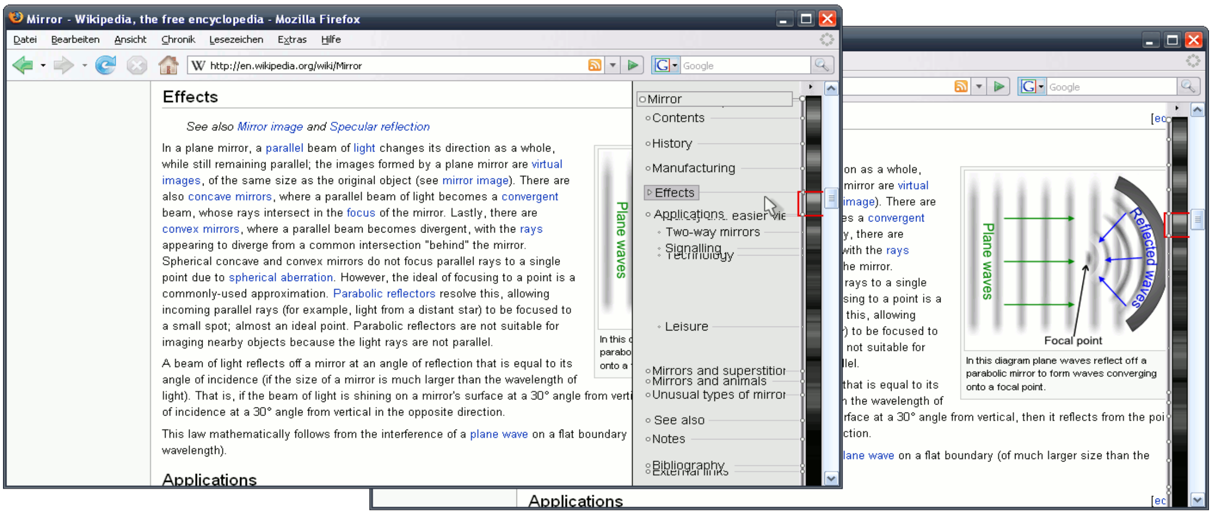
- Heatmap for visited parts of a page: To help the user identify which parts of a long page they have read before, the time spent at each scroll offset is visualized in a heatmap which is displayed beside the scroll bar. Clicks in the heatmap cause the browser to jump directly to the respective part of the page.
- Automatically generated table of contents: On a long page, users can display a table of contents which is generated from heading elements <h1> to <h3> on the page. The headings are distributed vertically along the scroll bar, which permits users to estimate the size of sections as well as gain insights about the content around their current scroll position.

This paper is organized as follows: After a look at related work in section 2, a detailed description of the features and UI concept of the prototype is presented in section 3. The implementation (a JavaScript-based script for the Greasemonkey extension of Firefox) is described in section 4. The evaluation in section 5 compares user performance for a long Wikipedia article with and without our tool. It is followed by the conclusion in section 6.

2. RELATED WORK

Several other tools try to solve similar problems. The Reader's Helper [3] was written in 1999, but many of its results are still valid today. A statement that inspired our work was that readers do not

NordiCHI 2008: Using Bridges, 18-22 October, Lund, Sweden



Mirror - Wikipedia, the free encyclopedia - Mozilla Firefox

http://en.wikipedia.org/wiki/Mirror

Effects

See also Mirror image and Specular reflection

In a plane mirror, a parallel beam of light changes its direction as a whole, while still remaining parallel; the images formed by a plane mirror are virtual images, of the same size as the original object (see mirror image). There are also concave mirrors, where a parallel beam of light becomes a convergent beam, whose rays intersect in the focus of the mirror. Lastly, there are convex mirrors, where a parallel beam becomes divergent, with the rays appearing to diverge from a common intersection "behind" the mirror. Spherical concave and convex mirrors do not focus parallel rays to a single point due to spherical aberration. However, the ideal of focusing to a point is a commonly-used approximation. Parabolic reflectors resolve this, allowing incoming parallel rays (for example, light from a distant star) to be focused to a small spot; almost an ideal point. Parabolic reflectors are not suitable for imaging nearby objects because the light rays are not parallel.

A beam of light reflects off a mirror at an angle of reflection that is equal to its angle of incidence (if the size of a mirror is much larger than the wavelength of light). That is, if the beam of light is shining on a mirror's surface at a 30° angle from verti
of incidence at a 30° angle from vertical in the opposite direction.

This law mathematically follows from the interference of a plane wave on a flat boundary
wavelength).

In this diagram plane waves reflect off a parabolic mirror to form waves converging onto a focal point.

Applications

Figure 1: The WebpageMap uses a heatmap to show the amount of time spent reading the different parts of the page. A table of contents is displayed when the mouse moves over the heatmap (top window). A red rectangle indicates the current scroll position.

necessarily read documents from the beginning to the end. Instead, they only skim through the text to extract relevant information. The additional UI elements of the described tool are quite complex and take up a significant portion of the available screen space. A thumbnail of the document is used as one of the main navigation aids.

In general, thumbnails are a popular way of providing a document overview. Space-Filling Thumbnails [2] display all pages of a document on the screen, allowing the users to employ their spatial memory to find previously visited pages. However, details may become too small to discern for very long documents, and the large amount of downscaled information may overwhelm the user. Thus, our approach only extracts and visualizes those types of information which are most likely to help during a search for information.

SnapShoot [5] is a web-based note taking system, it is one example for a system which supports the task of annotating documents. It contains interesting ideas on how to support active reading, which is also addressed in [6]. While annotations can be useful in the case that a user wants to study and summarize a document that is read from beginning to end, our tool is intended for the more common situation that a web document is scanned quickly in order to locate information. Taking notes should not be necessary in this case. Thus, our tool provides functionality that "automatically takes notes" according to the user's actions to support his workflow.

Different ways to support reading of electronic documents are compared in [4]. One of the findings is that a combination of an overview and the original text deepens understanding, but might increase the time needed to solve text-related tasks. As a result, when designing our interface we decided to develop a simpler overview, hence reducing the amount of information presented to avoid confusing the user. Donald Byrd [1] presents occurrences of search queries in documents by highlighting parts of the scrollbar.

3. ANNOTATED HEATMAP FOR NAVIGATION ON LONG PAGES

This section describes the different parts of the WebpageMap UI and illustrates how they can be used for various purposes. Once it is enabled, the tool's code executes on all pages visited by the browser, and displays a narrow vertical stripe (see figure 1) on pages which are much taller than the browser window itself, so that scrolling will be necessary to read all content. The stripe's position in the window is fixed, it does not scroll together with the page content. A small button in the upper right corner of the browser window can be used to toggle on or off whether it is displayed.

In the visualization, the height of the stripe corresponds to the height of the entire web page. A red rectangle is shown over the segment for the currently viewed part of the page content. The close vicinity of the WebpageMap to the scrollbar is intentional – in many ways, the look and feel of the two is similar.

3.1 Heatmap – "Have I Been Here Before?"

The WebpageMap stripe is subdivided vertically: The larger right part (19 pixels wide) shows a heatmap which will initially be blank (completely black) when a page is visited. As the user continues to look at the page, scroll around and read parts of its content, the tool's code runs in the background, taking note of how much time is spent at what scroll offsets. The gathered data is displayed in the heatmap stripe, scaled down to the height of the stripe.

The heatmap grows successively lighter in those areas which have been looked at for some time, and can turn white at spots which correspond to the most intensively viewed parts of the content. Usually, a first effect, i.e. parts turning from black into a dark shade of grey, can be observed after about 30 seconds. However, the code recognizes inactivity, e.g. when the user leaves the computer or switches to another window/tab. Occasional activity, such as mouse movements, is required for the heatmap to be updated.

While a page is viewed, the heatmap slowly begins to display a visual summary of the user's behaviour: Parts which were read several times or more slowly will appear lighter, whereas skipped content remains dark. This information is saved and redisplayed if the user leaves the page and visits it again at a later time.

With the heatmap, the user is given access to information which can be useful in a number of ways: First, if the map is not blank upon visiting a page, this is an unobtrusive hint that it was visited before – a fact the user might not be aware of while browsing through a document with many large pages. Next, the average intensity of the heatmap conveys information on the dwell time for the page. If only darker shades of grey are visible, the content was only skimmed, whereas lighter areas indicate more intensive reading. Finally, the map shows exactly which parts of the page were viewed. If the user has earlier read a few paragraphs on a long page and now wants to revisit one of them after returning to it, only a look at the content behind the different "hot spots" in the heatmap is required to find the wanted paragraph. With a small additional feature, the heatmap makes it easy to go to those parts of the page: When the mouse is clicked inside the map, the current page is scrolled directly to the offset which corresponds to the clicked-on part of the heatmap – it is not necessary to click on the scrollbar and drag it to the right position.

3.2 Heading Visualization – "How is the Page Content Organized?"

The smaller, left part of the WebpageMap stripe (7 pixels wide) shows a line on which dots of different sizes are placed. Each dot corresponds to one heading element on the web page, the dot size depends on the importance of the heading (the <h1> dots are the largest, the <h2> dots somewhat smaller, etc.). The vertical position of each dot is determined by the vertical offset of the respective heading on the page. Headings are automatically extracted from the page content during loading.

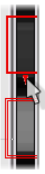
With this simple visualization, a visitor can quickly get a general idea about the page content: Is it organized into hierarchical sections, how many sections does it contain, and which sections or subsections occupy more space than others? Later on, when he is reading the text on the page, he can also quickly see how much text is left in the current section, even in the case that the next section heading is not yet visible at the current scroll position (which is indicated by the red rectangle).

The dots are only the summary of a full table of contents which is also made available by the tool: When the mouse pointer moves over the WebpageMap stripe, the right part of the browser window temporarily displays a list of headings, each of them connected to its dot by a horizontal line (figure 1, top window). The hierarchy of sections and subsections is visualized by displaying appropriately sized dots to the left of each heading label, and by indenting lower-level headings. Parts of the page are concealed by the list – to make them visible again, the pointer must be moved to the left until it hovers over the main page content again.

The table of contents has several advantages compared to the "Contents" section that is commonly placed at the top of long HTML pages by its authors. Like such a section, it allows the user to see the content structure, but it also shows the size of individual sections. Without scrolling up to the preceding heading, the user can see which section the current subsection belongs to. Furthermore, navigating to sections is very easy: The user can click on the label to go to any section. Unlike classical "Contents" sections, the concept supports jumps to the middle of sections rather than just their beginnings, in case the user knows the exact location of information he is interested in. Such a jump can be achieved with a click on the heatmap as described above, or with a click in the space between section labels. Another advantage of the automatically generated list is that it is present even on pages which do not include a table of contents at all. Finally, the pop-up list is very convenient when reading long specifications because it does not require scrolling back to the beginning of the page to look at the table of contents. Doing so without our tool means that the user needs to memorize the current scroll position in case she wants to continue reading there after a look at the "Contents" section.

Due to the way the table of contents is displayed, heading labels may sometimes overlap. To alleviate the resulting readability problems, labels are brought to the top and highlighted whenever the mouse pointer is over them.

Even with the pop-up table of contents, a common navigation pattern within web pages is that the user takes a quick look at another part of the page and then wants to go back to the previous scroll offset. This is supported by our tool using a "within-page back button": Whenever the WebpageMap is used to jump to another part of the page, a small red arrow appears above or below the red rectangle which visualizes the current scroll offset (see figure to right). Upon hovering over the arrow, a second rectangle indicates the previous scroll offset, and a click on the arrow restores it.



4. IMPLEMENTATION

The WebpageMap is written in JavaScript and designed to run as a user script in the Greasemonkey extension for Firefox, which executes it on all pages that the browser visits. However, it could be adapted for inclusion as a regular script running on a particular website. When used in this way, it would be available to all visitors of the site without the necessity to install the Firefox extension.

When a page loads, the headings (up to <h3>) are indexed together with their y offset on the page. This data is used to compute the position and determine the layout of the markers and headings in the table of contents. Dwell time information for the heatmap is recorded in an array with 3000 entries. This is big enough to allow accurate tracking of the scroll offset for large pages, but also larger than common browser window sizes, so that no inaccurracies occur when downscaling to the final heatmap size. The heatmap itself consists of many <div> elements, each of which displays three heatmap pixel rows via its upper border, background color and lower border. The scale of the heatmap intensity starts being adjusted as soon as one part of the heatmap has become completely white, so that the most intensively viewed part always corresponds to white and less viewed parts become darker over time.

The heatmap is updated on every keystroke, mouse movement and scroll action. Whenever this happens, the number of seconds that have elapsed since the last update is added to the appropriate entries of the dwell time array. To address the issue of idle times, there is an upper limit of 30 seconds that can be added by a single action. When the page is unloaded, the array is saved on the local machine via a mechanism provided by Greasemonkey.

When the mouse pointer enters any WebpageMap component, the heading corresponding to the pointer's y coordinate is highlighted in the table of contents by raising the entry's zIndex and changing its color and border.

5. EVALUATION

Most online reading activities can be divided into three steps: Identifying pages that contain relevant information, locating the information within the page and working through the content. A user study was conducted to evaluate how well these three steps are supported by our tool. There were twenty participants, thirteen male and seven female, aged 18 to 52, among them undergraduate students with technical and nontechnical background, a designer and a marketing assistant. All subjects use the Internet at least once a week or on a daily basis, and are familiar with online reading.

5.1 Tasks

The tool was introduced to the subjects by explaining the provided functionality using the German Wikipedia article about Africa as an example. The subjects had time to explore the features themselves after the introduction. The tasks that were used to evaluate the tool were performed on the German Wikipedia article about the USA. As mentioned before, we decided to divide

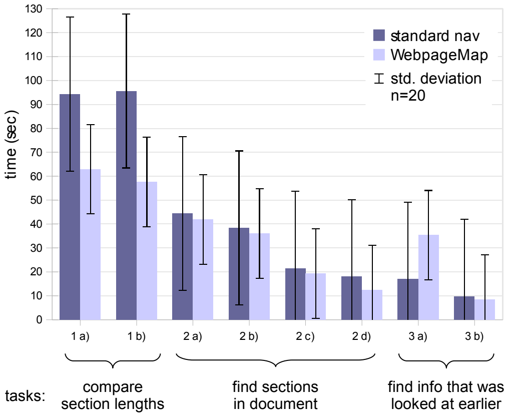


Figure 2: User performance was mostly improved with WebpageMap. Test users' subjective opinions were very positive.

the reading process into three steps. Accordingly, the subjects had to perform three groups of tasks related to these steps. Each such group (of two or four tasks) was followed by a questionnaire.

The easiest way to assess the relevance of a document is to determine the length of the sections associated with the matter of interest. In the first group of tasks, the subjects were asked to compare the lengths of four sections in the document. In the second group of tasks, they had to find several sections within the document. After finding a section, the test persons were asked to study its content for further questioning. This was done to simulate genuine interest in the content, which is needed to collect useful heatmap data. For the last group of tasks, the subjects were asked to find information that was located in sections they had visited earlier.

All subjects performed all tasks. Half of the tasks in a group used the heatmap, the other Wikipedia's navigation. To eliminate learning effects and the effects of possible minor differences between the tasks within a group, we systematically altered from subject to subject both the order of tasks within their group and whether the heatmap was used. The time needed for each task was measured.

5.2 Results

Our hypothesis was that the tool makes online reading easier and reduces the time needed to perform the tasks.

The first part of the hypothesis can be verified quite clearly by looking at the opinions that the test users submitted via 5-point Likert scales in their questionnaires: Most users preferred solving the tasks with the WebpageMap instead of the standard navigation as used by Wikipedia (average 1.4, where 1 is strong agreement, 5 complete disagreement). They also strongly agreed that the WebpageMap makes the given tasks easier for each of the three tasks of comparing section sizes (average 1.45), finding content (average 1.55) and revisiting previously seen content (average 1.55). When comparing the utility of the WebpageMap with the standard Wikipedia navigation in two separate questionnaire questions, a non-parametric Wilcoxon test showed that the difference in favour of the WebpageMap is highly significant for the first group of tasks ($T = 0$, $p < .001$, $r = -.61$) and significant for the second one ($T = 30$, $p < .05$, $r = -.39$). These results coincide with the observation that most subjects stated on more than one occasion that they felt lost in the text and would prefer using the tool even though the current task had to be performed without it. During the test, one feature that proved to be very popular was the possibility to jump directly to a passage of a text without having to scroll.

The results concerning the second part of the hypothesis differ depending on the kind of task. On average, most tasks (7 of 8) were solved faster using the WebpageMap than without it (as shown in the diagram in figure 2). With the first group of tasks, the difference is highly significant ($p < .003$), whereas the null hypothesis cannot be rejected for the other two ($p = .54$, $p = .50$). This difference is surprising at first glance. However, based on observation of the test procedure, it can probably be explained as follows: The second and third groups of tasks are strongly linked to deeper understanding of the content and the users' personal browsing habits, which can differ a lot. Participants' diligence or patience may have contributed more to the measured response times than the fact that the WebpageMap was present. For instance, some users always carefully checked all facts in the text before indicating that they had solved the task. It is also possible that the results would improve once the users become more familiar with our tool, since they would develop more efficient browsing strategies over time.

6. CONCLUSION

In this paper, we have presented a tool that makes it easier for web users to navigate in large web documents, to find information and to revisit previously seen parts of the document. This is done with a heatmap which provides an overview of the personal browsing history on a page, and by allowing quick access to an automatically generated table of contents. Compared to related efforts, our focus is on a simple user interface which provides as much useful information in as little screen space as possible. In an evaluation, our WebpageMap prototype performed well for all but one of the eight tasks. Even more than the measured performance improvements, the positive subjective feedback from users suggests that they appreciate the better document overview and the straightforward navigation possibilities of the tool.

Future work is possible in a number of areas – for instance, highlighting of search terms similar to the system in [1] would be feasible. However, extensions to the UI concept should be done carefully to avoid that it becomes crowded and confusing.

7. REFERENCES

[1] D. Byrd: A scrollbar-based visualization for document navigation. In *Proceedings of the 4th ACM Conference on Digital Libraries*, Berkeley, CA, USA, August 1999, ACM

[2] A. Cockburn, C. Gutwin and J. Alexander: Faster document navigation with space-filling thumbnails. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI2006)*, Montréal, Canada, pp. 1–10, April 2006

[3] J. Graham: The Reader's Helper: A Personalized Document Reading Environment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI1999)*, Pittsburgh, PA, USA, pages 481–488, May 1999

[4] K. Hornbæk and E. Frøkjær: Reading patterns and usability in visualizations of electronic documents. *ACM Trans. Comput.-Hum. Interact. 10*, 2 (Jun. 2003), pages 119–149

[5] S. Iga and M. Shinnishi: SnapShoot: Integrating Semantic Analysis and Visualization Techniques for Web-Based Note Taking System. In *Proceedings of the 2006 Asia-Pacific Symposium on information Visualisation*, Tokyo, Japan, pages 161–167, 2006

[6] B. N. Schilit, G. Golovchinsky and M. N. Price: Beyond Paper: Supporting Active Reading With Free Form Digital Ink Annotations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI1998)*, Los Angeles, CA, USA, pages 249–256, April 1998