

WebAdapter: A prototype of a WWW-browser with new special needs adaptations

Dirk Hermsdorf, Henrike Gappa, Michael Pieper

GMD - German National Research Center for Information Technology
Schloß Birlinghoven, D-53754 Sankt Augustin

Tel.: +49 2241 142018

Fax: +49 2241 142065

dirk.hermsdorf@gmd.de, henrike.gappa@gmd.de, michael.pieper@gmd.de

<http://zeus.gmd.de/projects/tedis.html/>

Abstract. This paper presents a prototypical WWW (World Wide Web)-browser called "WebAdapter", which provides new special needs adaptations for physically handicapped, blind and visually impaired end-users. These adaptations include near miss tolerances, implementation of sophisticated HTML-guidelines and advanced speech output. For evaluation purposes a usability test was conducted proving the suitability of the implemented special needs adaptations. The future goal of this work is a user interface for all (UI4All) for a standard Webbrowser. With regard to this perspective, the WebAdapter is still an reactive approach in that it only reacts to shortcomings of common Webrowsers instead of proactively integrating a standardized software layer between the user front-end and underlying applications by which the I/O-Interface can easily and universally be adapted to a variety of different personal needs of handicapped as well as able-bodied end-users. Thus, the WebAdapter only illustrates some new special needs adaptations which may be included into future versions of common WWW-browsers in order to move another step towards a UI4All. Further inevitable improvements of the user interface of WWW-browsers for disabled and elderly people beyond the scope of this project are discussed as well.

1. INTRODUCTION

Computer technology has become a very important part of our everyday life. It can be found in offices, libraries, homes and many other places. Nowadays, computers are a necessary tool to handle the mass of existing data. So it is very important that everyone including people with special needs can access them otherwise some people will be excluded from sharing in the social, cultural and economic activities of their communities. Ultimately, this problem can be solved by designing a user interface for all (UI4All).

A UI4All in general should be ...

- designed according to established *software ergonomic principles* like consistency, reliability, usefulness and clarity.
- *variable* in order to meet the diversity of tasks: text display, manipulation of audiovisual materials, virtual reality and so on [22].
- *modality independent*, that means the system should be able to present information to the user in different ways [22]. A blind user e.g. needs speech output rather than monitor output. This kind of information can be stored modal independent or modal redundant. Modal independent indicates that information can be offered in different ways like ASCII

text (visual, acoustic, haptic). Modal redundant means that information is stored in different formats, which enables the system to present it in different ways, e.g. a picture and a text description of the picture.

Furthermore, the system should provide interfaces to common assistive technologies like braille displays and speech synthesizers.

- *adaptable* which refers to user defined settings of preferences like determination of colors, fonts and so on [22].
- usable by people with and without special needs (*dual use*).
- *straight-forward* and *easy to learn* [22].

Especially for people like physically handicapped and blind persons, who are often directly or indirectly affected by mobility restrictions, the Internet and in particular the WWW is a very important source of information, possibly even allowing for access to working opportunities via telework. It provides a lot of different, widespread information in a familiar and ubiquitous environment. Thus the future goal of this work is to supply a common WWW-browser with a UI4All.

Information in the WWW is obtained by so called WWW-browsers. Recently released versions of common WWW-browsers like Microsoft Internet Explorer 4.01 and Netscape Navigator 4.0 sought to improve their user interface (UI) for people with special needs. Here one of the most important improvements is to see in the complete access by keyboard for the benefit of physically impaired users, who do not depend on using the mouse for navigation purposes in this environment anymore. The WebAdapter introduces other special needs adaptations which are not integrated in any other WWW-browser so far and can be included in future versions of common WWW-browsers in order to move another step towards a UI4All.

The WebAdapter is part of a master thesis, which was written within the scope of GMD's research project "TEDIS" (TEleworking for DISabled people) [9]. Thus, the WebAdapter by no means could have followed a proactive UI4All design-approach. Integrating a standardized software layer between the user front-end and underlying applications by which the I/O-Interface can easily and universally be adapted to a variety of different personal needs of handicapped as well as able-bodied end-users is beyond the horizon of a master thesis. Thus, the WebAdapter only illustrates some new special needs adaptations, which may be included into future versions of common WWW-browsers in order to move another step towards a UI4All. As well as standard Webrowsers, which up to now only incrementally and still insufficiently take into account the variety of personalized Human Computer Interaction, the WebAdapter is an reactive UI4All design-approach in that it only reacts to adaptive shortcomings of common Webrowsers

2. The WebAdapter

The WebAdapter provides common functionality like storing bookmarks, a context sensitive help function and additional adaptations for blind, visually impaired and physically handicapped people. The intention of this prototype is to illustrate the possibility to include adaptations for people with special needs in a WWW-browser without affecting the UI of non-handicapped users (Dual Use).

Due to restricted resources it was only possible to implement a prototype of a WWW-browser. The WebAdapter was implemented with Microsoft Visual Basic 4.0 and NetMasters FastNet ActiveX 6.02 [6]. FastNet ActiveX is a tool for retrieving and parsing HTML-documents.

2.1. Near miss tolerance

The near miss tolerance, which is integrated in the WebAdapter, enables the user to select buttons from the toolbar even when the mouse pointer is close but not directly touching the button for the benefit of physically impaired users. The closest button to the mouse pointer is always highlighted in case this feature is active serving as a visual feedback so that the user knows which button is selected when the right mouse button is hit. Links, widgets and so on can be selected with the left mouse button as usual.

The near miss tolerances are so far only implemented for the toolbar. An extension for links, image maps and so on would be useful, but could not be implemented due to interface problems with the used programming environment.

The implementation of a near miss tolerance is not only suitable for WWW-browsers, but for all mouse based applications.

2.2. Automatic application of HTML-guidelines

An other approach realizes the implementation of some HTML(Hypertext Markup Language)-guidelines [2, 3, 4, 5, 7, 16, 19, 20, 21, 23, 26]. These guidelines provide specific instructions for web designers to program a more accessible web side for people with special needs. Applying these guidelines should nevertheless not affect the aesthetics of the HTML-document so that non-handicapped users are not disturbed by the adaptations. Some of the HTML-guidelines which are implemented in the WebAdapter can be applied automatically.

A lot of the implemented guidelines realize a better presentation for screen readers. Screen readers are software programs, which read the textual screen contents to the user.

Altogether seven HTML-guidelines were found which could be implemented in the WebAdapter. The user can specify in the WebAdapter which adaptations should be applied to the current HTML- document. From a technical point of view, a guideline is applied by parsing and changing the HTML-code of the document.

The following adaptations are provided by the WebAdapter:

1 adaptation for physically impaired users:

- It is possible to specify a minimum size for clickable images. This is good for physically impaired users, who can enlarge the image and hit it more easy.

1 adaptation for visually impaired users:

- Turn off background image. It is hard to read text when there is a low-contrast to the background picture.

5 adaptations for blind users:

- Replace acronyms. Some screen readers have problems with letters like \$, &, % and so on. This feature enables the user to replace acronyms by other expressions like "dollar", "and", "percent" and so on.
- Insert a dot at the end of list elements, so that the screen reader will lower its voice at the end of each list item pointing out its end.

- Sequential presentation of tables.

Screen readers usually read tables line by line which totally confuses the listener. Additionally some screen readers do not pause between different cell items.

Sometimes it is possible to find out if there are headers in the first row or/ and in the first column of the table. That is for example the case if the HTML-tag <TH> (table header) is used or if the cell entries in the first column are all in bold. If the WebAdapter finds headers, he transforms the table in a sequential form: *"header1: cell item11, header2: cell item12, ..."*

The table browsing feature described in chapter 2.4 might be a more useful approach to access tables.



Screenshot: Original table presentation in the WebAdapter



Screenshot: Sequential tables in the WebAdapter

- Insert dots in abbreviations like EURO, GMD etc. so that a screen reader will pause between each letter and will not try to read the abbreviation as one word. An abbreviation is defined as two capitals following each other immediately. A wrong interpretation of abbreviations can not be excluded - for example the letters in a header might be all in capitals.
- Label lists (size of a list, number of sublists). Blind end users can get a quicker overview about the list structure when lists are labeled.



Screenshot: List labels in the WebAdapter

2.3. The project WAB (Web Access for the Blind)

The approach of modifying the HTML-layout has already been realized in the WAB (Web Access for the Blind [24]) project of the university of Zuerich. If you specify the proxy server as "http://ea.ethz.ch:8080" in a common WWW-browser, the HTML-layout will change according to the three following adaptations. Nevertheless WAB provides no modular

selection of the adaptations and blocks the selection of a proxy. The modular selection of the adaptations is implemented in the WebAdapter. The following adaptations are concerned:

- Provide a list of all links in the HTML-document. Sighted users can get a quick comprehensive overview of a document by just looking at it. Blind users do not have this ability. This feature will enable blind users to get a quick overview or impression of the document.



Screenshot: List of all links in the WebAdapter

- Provide a list of all headers in a HTML-document. Like the list of all links, the list of all headers provides a quicker overview of the document, even for sighted users. It is possible to click on the headers. That way the user jumps to the corresponding header in the document.
- Keywords like 'Image', 'Radiobutton', 'Checkbox', 'Editfield', 'Button', 'Combobox', 'refto' ('refto' stands for 'reference to', that means for a link which points at a position within the same web document) and 'Link' are inserted in the corresponding places in the web document. So the blind end user is informed about the object type when he is using a screen reader or a braille display. Furthermore, he can search for keywords which allows him e.g. to jump from link to link.

2.4. Integration of a screen reader on HTML-level

The synthetic speech output Speak&Win from ETeX, Frankfurt is integrated in the WebAdapter. The same synthetic speech is integrated in the "pwWebSpeak" browser from Productivity Works [18].

Using the integrated speech synthetic, the WebAdapter reads a HTML-document on HTML-level. This reveals an advantage in contrast to ordinary screen readers, which only can read the screen contents. The process of visualizing HTML-code turns explicit structure information which are specified in the HTML-code into implicit structure information. It is hard and often even impossible to rediscover this structure information [8].

The WebAdapter provides an advanced speech output interface. With this option it is possible to interact with the WebAdapter when the monitor is turned off. The only input device can be the keyboard. Almost all objects are assigned to shortcuts. Additionally the user can tab through all objects within an active window of the WebAdapter interface. This should make the system accessible for blind users.

Moreover it is possible to assign a pitch, speed and volume value to each HTML-tag. You can e.g. assign a fast and low voice to a header, a high voice to a list item and so on. In this way the structure of the HTML-document can be presented acoustically.

During the reading process, the last read link can be selected by pressing "a" ("a" for anchor). Optionally the system reads the current link and asks if this link really should be selected.

Additionally it is possible to jump between the different HTML- structure elements. Therefore the keys "o" (read the current element once again), "b" (backward with a specified step range) and "f" (forward with a specified step range) are provided. The step range in combination with the keys "b" and "f" can be specified by the user. A step range can be a fix jump value, words, sentences, paragraphs, links, headers, list elements and table elements. This selection can be performed in the general preferences of the WebAdapter or with the key "s" (set step range). The key "j" (jump) is used to set the reading position immediately after the list of all links and headers and, when "j" is pressed once again, to jump back at the beginning of the document. Within tables it is possible to navigate with "r" (right), "l" (left), "u" (up), "d" (down), "h" (read row header: if cell (x,y) is the current cell, the row header is (1,y)) and "c" (read column header: if cell (x,y) is the current cell, the row header is (x,1)) between the different table cells. Optionally the current row and column number is spoken as well, which only changes when the keys "r", "l", "u" or "d" are pressed. The acoustic navigation in tables starts in the first column and the first row when one of these keys is pressed and can only be aborted with the key "e" (end). Then the speech output jumps to the end of table. The acoustic navigation in hypertext systems is implemented in a similar way in other research projects [1, 14], but without a table navigation. Asakawa et al. [1] implemented additionally a fast forward function. The integration of a fast-forward and rewind-function might be a useful extension of the current system.

The adaptations do not need to be visualized. This shows the "Dual Use" aspect of the WebAdapter. While a visually impaired user looks at the unchanged version of the current web document, the blind user who sits at the same computer listens to the adapted version of the document (using headphones).

With the key "F2" it is possible to ask the WebAdapter which control element got the focus right now. If the HTML-document form is active, the program says the loading or reading status. This function is useful e.g. in order to enable blind users to start again quickly after a rest.

During the loading process, the WebAdapter says "Working" in order to inform the user that the system has not crashed, but is busy. The user can specify the interval of this speech feedback.

3. Evaluation of the WebAdapter

After the implementation of the WebAdapter was accomplished the usability of the system was investigated by the following test scenario:

- Pretest of the speech interface with five non-disabled end users in Sankt-Augustin (four computer professionals, one frequent computer user).
The monitor was turned off during the pretest in order to simulate the circumstances under which blind people have to navigate through a computer system.
- Test of the near miss tolerance with one physically impaired end user in Dortmund (very frequent computer user).
- Test of the speech interface with two blind and one visually impaired end user in Muenich (computer professionals).

The goal of the evaluation was to gain qualitative results in order to assess the implemented special needs adaptations in terms of their suitability as an extension for the user interface of a common WWW-browser as well as to research necessary enhancements. For these purposes, the rather small number of subjects was sufficient.

Each test scenario followed the same structure and took established test principles into account like the WebAdapter is to be tested, not the user [15]. After a brief description of the procedure of the usability test, an interview was performed in order to determine the computer skills of the subjects. Afterwards, each subject was asked to perform a predefined standard task representing typical navigation procedures while "surfing" through the web: navigate from the GMD homepage to the TEDIS homepage, find a certain information within the TEDIS homepage and open a local file. An observer noted remarks of the subjects while working on the standard task.

Finally an interview was conducted with the participants of the usability test consisting of open as well as closed formats of questions. The items of the questionnaire were selected according to the software-ergonomic principles as they are suggested in the EVADIS II-guideline developed for evaluating dialogue systems according to the ISO-Norm 9241 [17]. The dialogue principles of particular interest for redesigning the user interface as implemented in the WebAdapter are suitability for the task, controllability, clearness and self-descriptiveness. The open questions mainly served to ask for possible enhancements of the presented user interface. All subjects have been encouraged to express constructive criticism.

The main goal of the pretest was to verify the standard task, the interview guideline and the correct running of the WebAdapter itself. It revealed some bugs in the WebAdapter and the need to change some phrasings of the standard task as well as the test items on the questionnaire. The pretest of the speech interface with the five sighted end users showed differing results from the usability test with the blind and visually impaired participants as it was to be expected. First of all, the sighted users had much more problems to handle the non-visual feedback. It was also more easy for the non-sighted users to remind shortcuts. However, some suggestions for improvements like the possibility to determine different user profiles, a rewind and fast forward for the "read" function and more tangible information about the size of a loaded document (number of sites, approximate reading time; compare with "page summary" in chapter 4) were mentioned from both user groups.

In summary the evaluation showed the following results:

- The WebAdapter is easy to use
- The WebAdapter was suitable to perform the standard tasks
- A list of links is very useful for blind users while they navigate through different web sites
- Structural elements like anchors, headers and list items could be recognized in the "read"-function with an adjusted voice
- Good control of the speech interface (acoustic navigation)
- The near miss tolerance clearly facilitated the selection of buttons from the toolbar for the physically impaired end user

Furthermore, the subjects expressed improvements and extensions. Some of the suggestions of the pretest like the key "j" for the acoustic navigation, the key "F2" for the status information (compare with "Where am I?" in chapter 4) and the "Working" feedback were already implemented. The other suggestions are:

- Acoustic navigation should provide a step range "row"
- User defined shortcuts
- Fast forward and rewind for the "read" function
- More tangible information about the size of a loaded document (number of sites, approximate reading time; compare with "page summary" in chapter 4).
Some test users had difficulties to handle cryptic information like "this document contains 38765 bytes".
- Near miss tolerance for all selectable objects (clustering of objects into links, widgets, toolbar and so on)
- Different user profiles (e.g. novice, expert) for the speech interface and the HTML-adaptation
The WebAdapter allows the user to store a lot of different parameters like which HTML-adaptation is active and voices for a lot of different HTML-elements, thus some subjects were missing an option to store user profiles with different combinations of these parameters.
- Paragraphs could not be recognized in the "read" - function
- Number links (like in the WWW-browser Lynx)

- An extended pause during the reading process of two immediately succeeding links in order to distinguish between them more easily

The suggested improvements will be considered in the redesign process of the WebAdapter.

4. Other improvements for WWW-browser user interfaces

However further improvements of the user interface of WWW-browsers besides the adaptations in the WebAdapter have to be accomplished in order to achieve a UI4All. One important improvement is to supply WWW-browsers with sufficient interfaces to common assistive technologies.

The next list proposes further improvements of the user interface of WWW-browsers [10, 11, 12, 13, 25], which have not been implemented in common WWW-browsers like Microsoft Internet Explorer 4.01 and Netscape Navigator 4.0. Many of the proposed improvements can also be applied to other applications, not only WWW-browsers:

Alternative format for information objects:

- Presentation of the description field of PNG graphics
Blind users always need a textual description of multimedia-elements.
- No fading of ToolTips
When the ToolTip fades, it can not be detected by screen readers.
- Different "views" for the same document. Examples for different "views": table of contents (list of all headers), text version, sequential tables, bigger fonts and so on.
Using different "views" makes it possible to process a HTML-document according to the needs of the user.

Tables:

- Navigation with arrow keys through tables (compare with chapter 2.4).
- Presentation of textual information of tables (AXES and AXIS attributes) in order to distinguish the current row and column.
This position information facilitates the orientation in tables for blind end users.

Frames:

- Navigation through frame objects combined with a feedback in which frame the user is
It is a software-ergonomic requirement that every program can be controlled entirely by keyboard. Furthermore it is important for blind users to be provided with an exact information about the current status of the system which they are using (which element has the focus, error messages, accomplishments of processes and so on).

Java:

- Presentation of an alternative description for java applets (APPLET ALT).
Blind users need alternative text descriptions. A consistent use of the ALT attribute for all multimedia elements including applets might be a useful extension for HTML 4.0.
Nevertheless, HTML 4.0 provides a lot of new useful improvements for people with special needs. HTML 4.0 was designed by the W3C's HTML4 Working Group in co-operation with the Web Accessibility Initiative's (WAI) HTML & CSS Review Working Group. The GMD as a W3C member also participates in the WAI, which was launched to the occasion of the WWW6 conference in Santa Clara, CA in April 1997.

Orientation:

- A "Where am I?" function. This function should provide information about the current position of the user within the document like " 34% of the document have been read, the cursor is in the second frame, in the third header, in the fifth list element of a ten element list .. ".
Such a "Where am I?" function would facilitate the orientation within a HTML- document for blind users.
- A page summary like "this web document contains X images, Y frames, Z headers, ..".
Such a page summary function enables blind users to achieve a quicker overview about the structure of the HTML-document. The WWW-browser "pwWebSpeak" provides a page summary [18].

Navigation:

- Navigation per keyboard through headers, paragraphs, ..
This feature enables users and especially those users who can not use the mouse like blind people to navigate quicker through a HTML-document.
- Determination of a special keyboard command in order to tab through links in a selectable step range.
If a blind end user wants to select a link, he has to tab through all the preceding links sequentially. If the structure of the HTML-document is already well known to the user, this sequential tabbing is really time-consuming and can be shortened by defining a link step range.

Feedback:

- Visual and acoustical feedback for user interactions and occurring events (advanced speech output interface in the WebAdapter).
An additional feedback for occurring events can be provided when acoustical signals (speech, artificial sounds like "beep" or real sounds like a paper rifle) are used, which is especially useful for blind end users.
- Specification of the button size and the button color.
Visually impaired end users could select the most appropriate button color individually. Physically impaired end users could specify the most appropriate button size in order to hit the button more easy.

Presentation (for visually impaired end users):

- A high contrast mode (font sans serif, background white, normal text black, visited links violet, other links blue).
- A zoom modus.

Interaction:

- Recognition of words after only some letters have been typed (definition of macros, dictionary).
Physically impaired end users might perform keyboard input slowly. Especially in this case a reduction of the input length is necessary. Netscape 4.0 and Internet Explorer 4.0 provide this feature in the "Open URL" text field.
- Selection of links by typing the first letter of the link label (like the selection of icons in Windows). After the input of the first letter the corresponding link label should be highlighted. If the same letter is pressed again, the next corresponding link label is highlighted, if there is another link label with the same first letter. The final selection of a link could be performed by the "Enter" key.
A reduction of the input length can be especially useful for physically impaired end users.
- Integration of a scanning mode. In this mode, selectable objects and object groups are highlighted one after another. The user can specify the time how long the same object remains highlighted. The highlighted object can be selected by pressing any key enabling especially physically impaired users to choose objects with only one (big) key.
- A searched word is highlighted when it is found in a HTML-document. Furthermore this word should get the focus, thus it will be detected by a screen reader. If this word belongs to the label of a link, this link should be selectable with the "Enter" key.
Such a "search-function" would enable blind users a more straight forward interaction with a WWW-browser. The WWW-browser "pwWebSpeak" [18] provides such a "search-function".

5. Conclusion

This work introduced the prototypical WWW-browser WebAdapter with its new special needs adaptations near miss tolerance, implementation of sophisticated HTML-guidelines and advanced speech output. The evaluation showed that these new adaptations are useful for people with special needs. Nevertheless these adaptations have to be improved and enhanced by other improvements like a scanning mode, different "views", a "Where am I?" function and the like (see chapter 3 and 4).

In continuation of the presented project a new research project is planned at GMD aiming at implementing the mentioned improvements by using the APIs (application programming interfaces) of a common WWW-browser, in order to proactively - or at least *release-independently* - move towards a corresponding UI4All. Such an extended standard Webbrowser will be a valuable benefit for users with special needs, enabling them to access the already available and rapidly evolving internet-based applications for learning, working,

telecare or social purposes. Considering the growing number of applications offered through the internet for the management of everyday life and the rising number of elderly and disabled people throughout Europe, the need for assuring access to this medium becomes apparent. In a future scenario WWW-browsers will be supplied with an ubiquitous interface enabling e.g. a blind person to access his or her individually tailored system at a railway station. A personalized homogenous user interface will be available then at any place.

However, despite all efforts to supply an accessible WWW-browser, this will not be sufficient in order to achieve a "WWW for All". It is just as important to advocate for the design of more accessible web pages. For this purpose the mentioned HTML-guidelines (see chapter 2.2) were created, but only a few of them can be applied automatically. Unfortunately the most important guidelines like an alternative description for multimedia elements can not be automated. They can be integrated though in web authoring tools as it is proposed now by WAI (Web Accessibility Initiative).

REFERENCES

- [1] Chieko Asakawa, Takashi Itoh, "User interfaces of a home page reader", IBM Japan, in: ASSETS '98, the third international ACM conference on assistive technologies, April 15-17, 1998, Marina del Rey, CA, ACM press, ACM ISBN: 1-58113-020-1
- [2] Adaptive Technology Research Center, University of Toronto, Jan Richards, "Guide to Writing Accessible HTML", 1995, <http://www.utoronto.ca/atrc/rd/html/html.html>
- [3] Adaptive Technology Research Center, University of Toronto, Mike Paciello, "Making the Web Accessible for the Blind and Visually Impaired", 1996, <http://www.yuri.org/webable/mp-blxax.html>
- [4] DO - IT (Disabilities, Opportunities, Internetworking & Technology), "DO - IT HTML Guidelines", 1996, <http://weber.u.washington.edu/~doit/Resources/design.html>
- [5] Paul Fontaine, Center for Information Technology Accommodation, General Services Administration, Washington, DC. USA, 1995, "Writing Accessible HTML Documents", <http://www.yuri.org/webable/htmlcode.html>
- [6] NetMasters FastNet ActiveX, <http://www.netmastersllc.com/fnax.html>
- [7] InfoUse, Center for Accessible Technology (CforAT), "The User Interaction Design Guidelines Document", 28.12.1995, Version .955, compare <http://www.infouse.com/disabilitydata/guidelines.html>
- [8] Frankie James, "AHA: Audio HTML Access", in "The sixth international world wide web conference - proceedings", April 7-11 1997, Santa Clara Convention Center, California, USA, ISBN 0-9657614-0-1, <http://proceedings.www6conf.org/HyperNews/get/PAPER296.html>
- [9] Henrike Gappa, Stefanie Mermet, Michael Pieper, "Teleworking for Disabled People - Pitfalls and ongoing Challenges", in "Computers and Assitive Technology, ICCHP '98", Editors: Alistair D.N. Edwards, Andras Arato, Wolfgang L. Zagler, page 438 - 444, ISBN 3-85403-118-1
- [10] Jon Gunderson, "World Wide Web Browser Access Recommendations", <http://www.staff.uiuc.edu/~jongund/access-browsers.html>
- [11] Jon Gunderson, Gregg Vanderheiden, D. Shumila et al., "Suggestions for Browsers", http://trace.wisc.edu/docs/browser_sug_blind/browsers.htm

- [12] Dirk Hermsdorf, Michael Pieper, "BSCW for Disabled Teleworkers - Usability Evaluation and Interface Adaptation of an internet - based Cooperation Environment", in: Michael Genesereth and Anna Patterson (ed.), Proceedings for the Sixth International World Wide Web Conference, Santa Clara, CA, USA, 7-11.4 1997, ISBN 0-9657614-0-1, <http://proceedings.www6conf.org/HyperNews/get/PAPER161.html>
- [13] Dirk Hermsdorf, "WWW Design fuer Alle", Master thesis, University of Bonn, August 1998
- [14] Sarah Morley, Helen Petrie, Anne-Marie O'Neill, Peter McNally, "Auditory navigation in hyperspace: Design and evaluation of a non-visual hypermedia system for blind users", in: ASSETS '98, the third international ACM conference on assistive technologies, April 15-17, 1998, Marina del Rey, CA, ACM press, ACM ISBN: 1-58113-020-1
- [15] Jakob Nielsen, "Usability Engineering", Academic Press Limited, 1993, ISBN 0-12-518406-9
- [16] Jakob Nielson, SunSoft Distinguished Engineer, "Accessible Design for Users With Disabilities", October 1996, <http://www.sun.com/columns/alertbox/9610.html>
- [17] R. Oppermann, B. Murchner, H. Reiterer, M. Koch, Mensch Computer Kommunikation 5/2, Software - ergonomische Evaluation, Der Leitfaden EVADIS II, 2., neu bearbeitet und erweiterte Auflage, 1992, Walter de Gruyter & Co., ISBN 3 11 013506 X
- [18] The Productivity Works, The pwWebSpeak Project, <http://www.prodworks.com/pwwwebspk.htm>
- [19] A. Savidis, C. Stephanidis, Developing Dual User Interfaces for Integrating Blind and Sighted Users: the HOMER UIMS, 1995 ACM Conference on Human Factors in Computing Systems (CHI '95), Denver, USA, 7-11 May 1995, pp. 106-113.
- [20] A. Savidis, G. Vernardos, C. Stephanidis, Embedding Scanning Techniques Accessible to Motor-Impaired Users in the WINDOWS Object Library, HCI International '97, San Francisco, USA, 24-29 August 1997, pp. 429-432.
- [21] SIH Laeromedel Datapedagogen, Christer Johansson, "Designing information for WWW - Tips and guidelines for increasing accessibility for people with disabilities", <http://www.sih.se/engelsk/brosch/wwwcont.htm>
- [22] C. Stephanidis, A. Paramythis, C. Karagiannidis, A. Savidis, Supporting Interface Adaptation: The AVANTI Web-Browser, 3rd ERCIM Workshop on User Interfaces for All, Obernai, France, 3-4 November 1997 [20] Clas Thoren, "Nordic guidelines for computer accessibility", ISBN 91-86954-15-6
- [23] Jutta Treviranus, Chris Serflek, University of Toronto, "Alternative Access to the World Wide Web", <http://www.utoronto.ca/atrc/rd/library/papers/WWW.html>
- [24] Gregg C. Vanderheiden, "Anywhere, Anytime (+Anyone) Access to the Next - Generation WWW", in "The sixth international world wide web conference - proceedings", April 7-11 1997, Santa Clara Convention Center, California, USA, ISBN 0-9657614-0-1, <http://proceedings.www6conf.org/HyperNews/get/PAPER253.html>
- [25] Gregg Vanderheiden (Trace Research and Development), Wendy Chisholm (Trace Research and Development), Ian Jacobs (W3C), "WAI Accessibility Guidelines: Page Authoring", W3C Working Draft, 14-Apr-1998, <http://www.w3.org/WAI/GL/WD-WAI-PAGEAUTH-0414.html>

- [26] Projekt "Web Access for Blind Users", Institut fuer Informationssysteme, ETH Zuerich,
<http://www.inf.ethz.ch/department/IS/ea/blinds/>
- [27] WAI Accessibility Guidelines: "User Agent", W3C Working Draft, 18-June-1998,
<http://www.w3.org/TR/WD-WAI-USERAGENT>
- [28] Adlai Waksman, "Making your web pages Lynx friendly",
<http://www.cfar.umd.edu/~waksman/lynx/making.htm>