

# Web Accessibility: A Broader View

John T. Richards  
IBM T. J. Watson Research Center  
19 Skyline Drive  
Hawthorne, NY 10532  
+1 914-784-7731  
jtr@watson.ibm.com

Vicki L. Hanson  
IBM T. J. Watson Research Center  
19 Skyline Drive  
Hawthorne, NY 10532  
+1 914-784-6603  
vh@watson.ibm.com

## ABSTRACT

Web accessibility is an important goal. However, most approaches to its attainment are based on unrealistic economic models in which Web content developers spend too much and receive too little. We believe this situation is due, in part, to the overly narrow definitions given both to those who stand to benefit from enhanced access to the Web and what is meant by this enhanced access. In this paper, we take a broader view, discussing an approach that costs developers less and provides greater advantages to a larger community of users. While we have quite specific aims in our technical work, we hope it can also serve as an example of how the technical conversation regarding Web accessibility can move beyond the narrow confines of limited adaptations for small populations.

## Categories and Subject Descriptors

H.5.2 **Information Interfaces and Presentation:** User Interfaces – *Input devices and strategies*. K.4.2 **Computers and Society:** Social Issues – *Assistive technologies for persons with disabilities*. K.4.2 **Computers and Society:** Social Issues – *Handicapped persons / special needs*. K.5.2 **Legal Aspects of Computing:** Governmental Issues – *Regulation*.

## General Terms

Human Factors, Standardization, Legal Aspects.

## Keywords

Web Accessibility, User Interface, Standards.

## 1. INTRODUCTION

There is a growing, worldwide recognition that users with disabilities have the same right as others to access information technologies. This recognition is manifested in the enactment of legislation like that in the United States whose aim is to make the Web and other information technologies accessible to users with disabilities [18]. Such legislation has led to the creation of standards, guidelines, and checklists for accessibility. The goal is straightforward: To develop a common understanding of what is needed to make Web pages accessible thereby enabling Web designers and developers to meet these requirements [18], [23]. To help Web authors determine whether pages they create do, indeed, meet these guidelines, Web accessibility validation tools [5], and professional Web site accessibility audits e.g., [9] are available to Web authors and Web site designers.

Consider, however, the economic “equation” presented to Web content developers by this approach to accessibility. First, the *cost* of creating accessible Web content is substantial. On some sites, hundreds or thousands of pages of “legacy” Web content must be changed. Even minor improvements like the addition of ALT tags for images quickly add up when an entire site must be made accessible. Anyone who has had their site subjected to a proper accessibility audit will confess that missing ALT tags is but the tip of the iceberg.

Second, the *benefit* of creating accessible Web content is often not obvious. After all, another word for legacy is “old”. Few developers willingly rework old content when so much new content needs to be created. Moreover, the guidelines necessarily focus on those most in need of assistance. Small, narrowly defined populations do not translate into large economic benefits for those who respond to their unique needs. Guidelines and standards are just not enough to cause most Web content developers to create broadly accessible content for everyone who needs it.

So how might the economics influencing Web accessibility be changed? How might the costs be reduced and the benefits increased?

Technology of various sorts provides an obvious approach to cost reduction. We can move beyond Web site compliance checkers to semi-automatic Web site repair tools [2]. Better still, we can provide Web adaptation facilities that transform existing Web content “on the fly”. The technology discussed in this paper provides an example of one such on the fly transformational capability.

Increasing the benefits side of the economic equation requires a broader definition of *who* stands to gain from Web content adaptations. Older adults represent a large and growing population who gain much from a more accessible Web. Easier reading, less cognitive load, fewer visual distractions, and more efficient keyboard and mouse use all translate to direct advantages for this group of users. More effective use of the Web *by* this population (generally possessing both more free time and greater discretionary spending power) translates to a worthwhile return on investment for creators of accessible Web sites. In this paper, we discuss our work in making the Web more usable for older adults.

“More usable” is a key phrase here. Accessibility has a technical specification spelled out in regulations. Usability has a broader definition generally resistant to attempts at specification. But usability matters to nearly everyone. The appeal of enhanced

usability is thereby universal (universality being a strongly multiplicative factor on the benefits side of the equation).

It is also fair to say that the technical approach we discuss here actually offers benefits for nearly everyone, not just disabled persons and the growing population of older adults. Even among the young, poor lighting conditions, small screens, and plane rides with laptop screens at sub-optimal angles reduce our ability to easily read and navigate. For those with cognitive and attentional disabilities, and for those learning to read a non-native language, enhanced readability is particularly useful. We discuss some of these benefits near the end of the paper.

Finally, the motivation for enabling a larger group of users to make better use of the Web follows directly from the fact that the “Web” is not just about recreational surfing. Often persons with disabilities are largely housebound and the Internet and e-mail can be their lifeline to the world. Significant commerce and critical information access is mediated by the external Web now. Intranets host a growing proportion of internal business applications. “Web services” eventually will tie global commerce together. The desktop is itself becoming little more than a Web browser. The browser is itself an embeddable component in a growing set of other applications. Making the Web more accessible and usable in this broader sense is obviously valuable.

We would argue that the impact of guidelines, standards and accessibility authoring support is likely to remain modest and confined primarily to government and other publicly-controlled websites for which regulations are enforced. Those not required to make their site accessible will largely not be incented to adopt these guidelines until they see the economic advantage. We would propose, instead, an approach that directly addresses the issues and problems from the end-user perspective. The approach we have developed manipulates Web content by combining and applying a number of page transforms and adaptations according to user preferences without requiring Web designers and developers to rewrite their Web content. We have built this capability as a browser extension and refined it through feedback from a variety of user groups.

Ours is not the first attempt to provide adaptations to enable access to Web content for persons with disabilities. For blind users, talking browsers render Web content in speech [1], [25]. Some websites provide for low vision versions of their own content, e.g., [3], [10]. Other systems allow users to make adaptations to a wider range of Web content, although these are limited in function to specific visual transformations and require users to go through special gateway pages to load Web pages [4], [14]. Our approach differs from these in having the goal of providing adaptations for the full Web, not a subset of Web pages, dynamically implementing these adaptations as users access unmodified Web content. Moreover, our work targets multiple limitations.

Our approach is based on several observed user and situational characteristics. First, people tend to prefer a standard browser with the accessibility adaptations added rather than a specialized browser offering only a limited set of features (which would also tend to “mark” them as being disabled). Second, people possess multiple interacting and fluctuations of disabilities. They therefore require rather substantial flexibility in selecting and combining adaptations. Third, many people *desire* assistance of a form not addressed by current accessibility guidelines. A prime

example of this is the desire of visually impaired users to use the vision they *do* possess rather than use the speech output of a screen reader. Fourth, it is not possible to know at first encounter what a person’s required adaptations are. Thus, our solution needed to address the difficult bootstrapping problem of how to make the selection of adaptations itself accessible. Fifth, people are generally unable to map from an abstract characterization of a possible adaptation to its actual effect. Our solution therefore needed to make it easy to directly *try out* any adaptation on any Web page at any time. Sixth, people want to be able to access the entire Web. Our approach could not, therefore, simplify the problem by constraining content to a fixed set of pages (as is the case with approaches relying on special markup). Finally, people do not want to spend a lot of time adjusting Web adaptations. A few clicks, at most, are what people will tolerate when what they are trying to do, after all, is *use* the Web.

In this paper, we present an approach that serves a large number of users whose needs fall outside the current accessibility standards and assistive tools for the Web. In the next section, we highlight the challenges and the requirements posed by user needs, technology constraints, and legacy content. We argue the case for an approach that provides an effective solution for serving a large and non-homogeneous population. After this, we present our approach which involves adapting the behavior of the Web browser and input devices, transforming pages according to user preferences at the client-end, and using additional channels for the presentation of information. This presentation is followed by a discussion of user interface design requirements and the challenges and issues with implementing the approach. Then, we present user feedback and usage of the prototype by various user groups. We conclude with discussion of future directions for the work.

## 2. ENHANCED WEB ACCESSIBILITY AND USABILITY

We were motivated initially to provide a solution for older adults. The U.S. National Institute on Aging has conducted studies on the use of information technology and has compiled guidelines for designing Web sites specifically for elderly users [6], [15]. These guidelines go beyond the regulatory standards. Like other guidelines, these typically are not followed by Web content designers.

More recently, however, we have expanded our target population. Interestingly, it has become clear that our adaptations work well for a number of disabilities, as will be discussed later in this paper.

We took as our starting point the fact that older adults face specific challenges when using the Web. Sensory, physical, and cognitive changes occur with age that will impact most of us and can limit our ability to utilize the Web fully [7], [8], [16]. Visually, decreases in acuity, contrast discriminations, and color perception accompany aging. Physically, arthritis and tremors accompany aging and partial paralysis is not uncommon. Cognitively, memory impairments and attentional problems arise. This situation suggests that certain aspects of the visual presentation of Web content are difficult for older users. There are the obvious difficulties with small font sizes, font colors that make reading difficult (particularly in combination with certain background colors), and background images on Web pages that decrease legibility. The small size of scroll bars is also problematic for users with severe vision limitations. Visual

clutter, such as multiple columns, and animations, are all problematic.

A number of problems specific to keyboard and mouse usage also can be identified. Certain limitations with hand mobility can lead to repeated letters when typing and difficulty with input that requires simultaneous key presses (as when holding the Shift key and pressing another key to make a capital letter) [21]. Mouse problems include inability to click a small target, moving the cursor off the target in the process of clicking, timing problems when double-clicking is required, and inability to coordinate the actions required to use the mouse for dragging and scrolling.

## 2.1 Web Content Adaptations and Transformations

Our system goals and design are motivated by the necessity to accommodate the multiple requirements and restrictions posed by user population, technology, and legacy content. All must be addressed by an effective solution servicing a large and non-homogeneous population of disabled persons.

One set of content adaptations was designed to address the visual presentation of pages. Certain changes such as font enlargement, font style (sans serif), increased inter-letter spacing, and enhanced color contrast can increase legibility for this population [6]. For users with greater degrees of visual problems, we provide additional changes, augmenting the text with speech output and providing options for very large banner-text displays, enlargement of browser controls, and page magnification. As is typical of magnification techniques, critical information often scrolls off the screen, forcing horizontal scrolling in order to see all page content. Given that horizontal scrolling is difficult for everyone, this adaptation has the potential to make browsing more rather than less difficult for our users. To ameliorate this problem, we also provide the option for changing the page layout by linearizing it, reducing the multiple column format to a single column as shown in Figure 6.

Other adaptations to the visual presentation of Web pages address both cognitive and visual limitations. For example, hiding images, backgrounds, and animations as well as performing page linearization combine to reduce the visual clutter on pages that has both visual and cognitive impacts. The page linearization feature provides an excellent adaptation in such situations. Using this feature, multi-column pages are transformed into one column. In cases where the Web authors provide skip navigation links, our page linearization feature brings the user directly to the main page content. The original intent of skip navigation links was to enable screen readers to skip over navigational menus and lengthy lists of links to access the primary content more directly. For our purposes, these skip navigation links similarly allow users, without a screen reader, to immediately see the main content. The benefits of page linearization relate to users with low vision who want to reduce screen clutter, users with cognitive limitations who similarly want to reduce screen clutter, and users with limited hand use who wish to reduce horizontal scrolling requirements (particularly with magnified pages).

Adaptations were also developed to target limitations in hand usage. Some Web activities, such as writing e-mail, filling in forms, or completing login and registration information, require the use of a keyboard. Appropriate keyboard accessibility settings can significantly improve the accuracy of typing for some users,

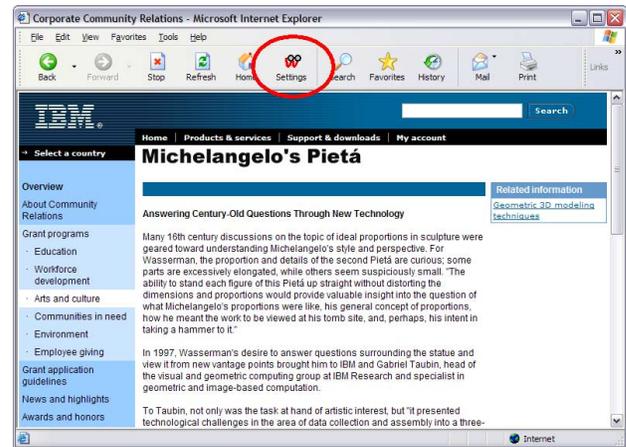


Figure 1. The 'Settings' button at the top brings up the settings panel.

yet these settings are difficult to learn and manipulate [21]. To address this problem, we provided users with easy ways to configure their keyboard settings.

## 2.2 User Interface Design Requirements

Our initial user population, older adults, posed several additional user requirements that motivated our interface design. First, many of these older adults were new to computing and very few were computer literate. Consequently, the specification of user preferences could not require prior computer experience.

Second, user preference specifications must be accessible to users. This is a particularly interesting challenge because we do not know, in advance, the needs of a particular user. Thus, the settings themselves must be fairly accessible to be usable by people with a variety of limitations.

Finally, users with multiple disabilities must be able to apply transformations in combination. In this regard, our unified approach to Web accessibility, rather than an approach based on a set of individual devices or applications, made this goal achievable.

These requirements, in concert, caused us to make it extremely simple for users to set preferences. To this end, users click a 'Settings' button on the browser toolbar to bring up a band that contains a series of control panels (see browser's otherwise standard toolbar buttons in Figure 1). The band appears at the bottom of the browser displaying the first of the control panels (see Figure 2 which shows the panel for setting 'text size'). To make the Web page text larger, the user simply clicks on the button showing the larger size, as shown in Figure 2.

For some adaptations, such as keyboard adjustments, banner text display, and speak text, the results of the adaptations are not immediately apparent on the Web page. Rather, the effect becomes apparent once the user types or points at text. For adaptations that effect the global page presentation, however, the requested changes are immediately applied to the current Web page. Clicking the larger text button shown in Figure 2 immediately increases the size of the text. If the user does not like the results of the change, they can easily adjust it.

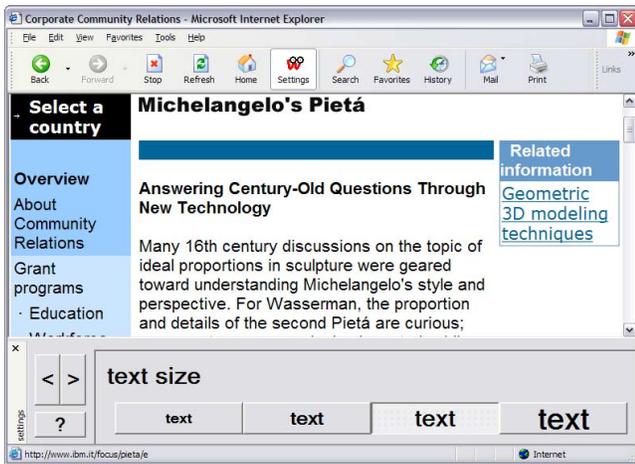


Figure 2. Web page showing panel for 'text size' set to a larger option.

The currently available adaptations, as presented by the settings panels, are the following:

- Speak text
- Magnify (pages)
- Text size
- Banner text
- Text style
- Line spacing
- Letter spacing
- Colors (text, background, links)
- Page layout (linearization)
- Images (enlarge and enhance)
- Hide (images, animations, background)
- Large browser (pointers, controls)
- Keyboard (one hand, key clicks, mouse keys, plus typing adjustments)

Access to each of these adaptations (except the typing adjustments, which will be discussed below in Section 2.3.2) is provided through a different interface panel. Users can cycle through the panels using the arrow buttons on the left of the band. Interactive help, keyed to the particular panel, is available by clicking on the '?' button. When the user is satisfied with a particular adaptation or a set of adaptations, users can close the band and browse the Web as usual. All subsequent Web pages that are viewed will have the selected adaptations automatically

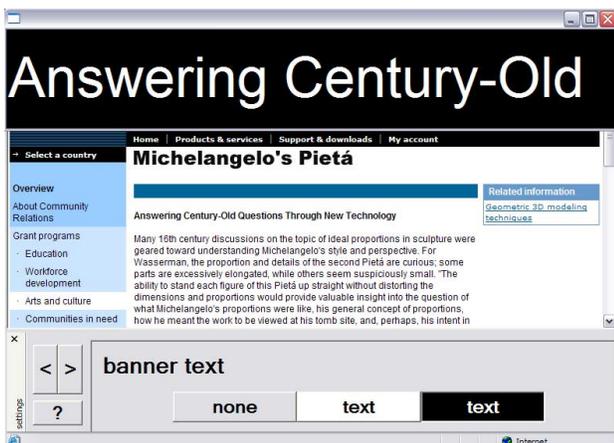


Figure 4. Web page showing banner text.

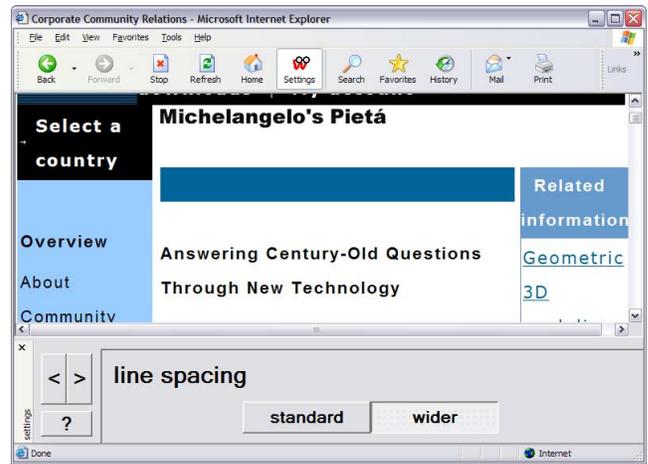


Figure 3. Web page showing adaptations for large font, letter spacing, and line spacing.

applied. Transformations are applied whether the band is present or not. Additional examples of transformations are shown in Figures 3 – 6.

The ease with which users can test different adaptations means that they can see which transformations work for them. This is particularly important in cases where users may not know in advance which adaptations will prove beneficial. In fact, users may discover useful transformations that they may not have predicted in advance of trying them out. Consider, for example, the font style adjustments and letter and line spacing provided by our software. None of our users requested this in our initial interviews. The literature, however, suggested that such adjustments would increase legibility [6]. When given the option to try out this adaptation, many users did indeed discover that it improved legibility. A design that assumed that users could pick out their needed adaptations from a checklist, for example, would likely not lead to this outcome.

In addition, the simple interface we employed offered the advantage of making the system's built-in accessibility options easily available to our users. As noted previously, some accessibility options are built into the operating system or browser. By and large, however, users were unaware of these options. Many were aware that font size could be set in the

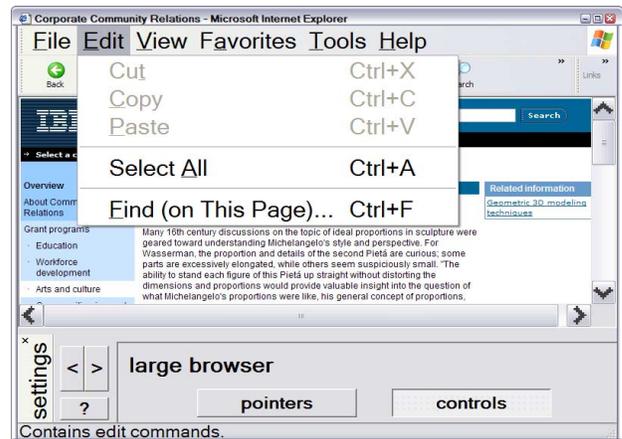


Figure 5. Web page showing enlarged browser controls.

browser, but they were unaware that there were other features that could also be set. These other options are often obscured by several levels of unintuitive menus and dialog boxes that additionally are difficult to click for users with motor difficulties.

For example, consider the steps that a user must go through in order to change the text and background colors on Web pages to white text on a black background. In Internet Explorer®, the user would have to progress through several steps involving selection of menu items, a series of dialog boxes, clicking small check boxes and buttons, dealing with terminology such as “visited link color”, and knowing to override system defaults. Font, background, visited link, unvisited link, and hover colors all must be individually set. Changing just the font and background colors requires a minimum of 16 different steps. The required skills for making a change of color include the ability to see and click small targets, and the cognitively complex text of dealing with multiply embedded menus and dialog boxes. Moreover, the color options that can be set may result in colors that, in combination with link colors, for example, render the page unreadable. For users new to computing, this task is made even more difficult by having to understand system settings and deal with unfamiliar terminology. The dual problems of user limitations and lack of computer expertise make this a daunting task. As shown in Figure 7, with our interface users simply click the button indicating their preferred text and background color and the browser adjusts to these colors including setting maximally-contrasting visited link, unvisited link, and hover colors.

## 2.3 Implementation

Our initial work in this area explored using a proxy server [11], [12]. Like other approaches that use Web intermediaries to transform Web pages e.g., [1], [4], [20], our proxy software had the advantage of not requiring any client software. In our proxy implementation, the only required client modification was the setting of the browser’s HTTP proxy, directing that requests pass through a specified Web address. Among the downsides of this proxy approach, however, was the difficulty in accurately rendering the Web pages of a substantial number of Web sites [11]. This approach required, in essence, the building of a full-functioned browser on the proxy server, capable of interpreting JavaScript, cascading style sheets, and a variety of proprietary software. An attempt to circumvent some ambiguity in this

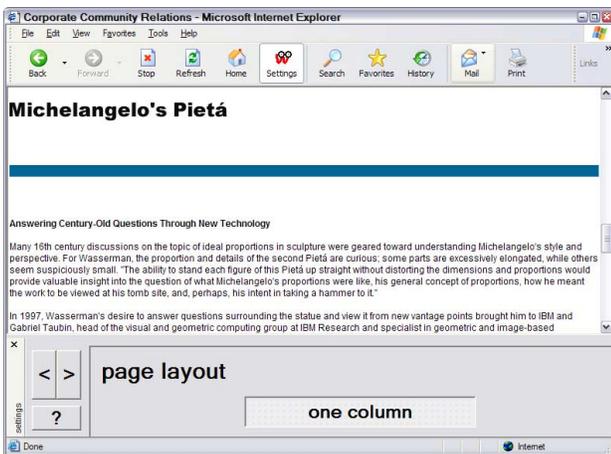


Figure 6. Web page showing linearization. This page also takes advantage of the skip-navigation tag to directly position the browser to the primary content.

intermediary transcoding approach has been to require that Web sites be annotated for presentation [2], [20]. As discussed above, however, most Web sites, particularly legacy sites, will not be annotated so will remain inaccessible.

Our current approach transforms the content of HTTP requests without a proxy server. These transformations are performed on the client machine using a combination of approaches involving DOM manipulations and utilization of browser and operating system features. No changes are made to the page source.

### 2.3.1 DOM Manipulations

Certain dynamic visual changes to Web pages are accomplished via manipulation of the DOM produced by the browser itself. Our Internet Explorer® implementation uses a Browser Helper Object (BHO) written in Java, which gives program access to the document object before it is rendered in the browser (but after any JavaScript or other dynamic changes to the document have been made). Existing DOM APIs support both document manipulation and handling of user interface events in the document via the BHO. As a result, we do not need to be concerned with how the HTML is parsed. We simply deal with it once the DOM is constructed.

Examples of DOM-based visual transformations include changing colors (text, background, and links), and preparing mechanisms for text selection for presentation of very large ‘banner text’ and speech rendering.

### 2.3.2 Utilization of Browser and Operating System Features

In some parts of our implementation, we exploit features built into the browser (e.g., font size) and operating system (e.g., StickyKeys, FilterKeys, and the size of browser apparatus) to make Web browsing easier. Sometimes this involves a straightforward setting of these features by users, as in the case of StickyKeys. In this case, there is a keyboard panel in the settings interface, with a button for StickyKeys labeled ‘one hand’ (since StickyKeys are useful for people who type with one hand).

In other cases the setting of operating system features is not quite so direct. Take the case of debounce time. Persons with tremors, for example, may depress one key multiple times in rapid

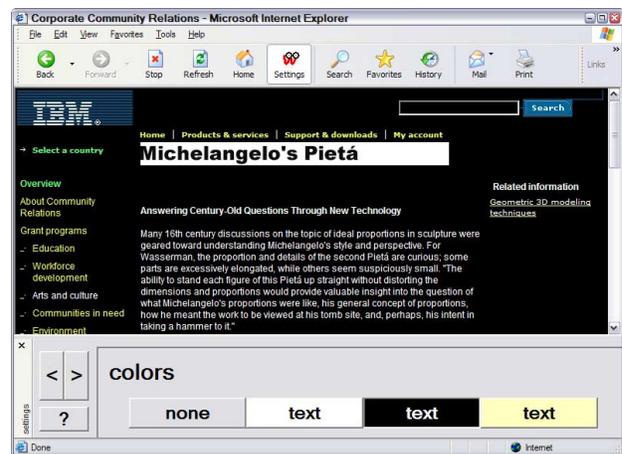


Figure 7. Web page showing panel for colors with the page colors transformed.

succession, causing repeated letters to appear when typing. Windows® allows users to set a ‘debounce’ parameter to control the length of a time period in which repeat keys will be filtered out. In order to set this, however, not only must users know that this feature exists, but they must go through a series of dialog boxes to get to the option. For readers who haven’t tried this feature, a minimum of 10 steps are required, several requiring clicking of a small button or an even smaller checkbox. Most daunting is that the user must be able to specify directly the required debounce time in milliseconds, having some sense of what this would translate into behaviorally.

In our software, user requirements for the most critical typing options are inferred from short samples of a user’s typing. Thus, as a user types, our program makes a determination of optimal keyboard settings referencing typing models tested with users having a variety of keyboarding difficulties due to limited hand coordination [17] [22]. Based on this analysis, the keyboard accessibility parameters are automatically adjusted to maximize typing accuracy for an individual user. While there are counter-arguments to the use of self-adapting systems [20], our pilot work indicated this to be the most effective approach for our users. Specifically, the notion of “correcting typing” was too difficult to convey to users in a situation where there was not in-person instruction. Users who experience no motor difficulties kept trying this option (before it was hidden) with the expectation that it would improve their spelling.

### 2.3.3 Saving User Preference Settings

From the user’s perspective, a Web session begins by logging in to a server. The logon screen is designed to be usable by persons with limited vision and keyboarding ability. Specifically, the screen completely fills a 640 x 480 display, with extremely large text indicating where people should enter their username and password. The typing adjustment software described above is active and with little typing input is capable of making adjustments to help users more accurately enter their information.

Upon successfully logging in, returning users have their settings preferences retrieved and applied. If they are still happy with those preferences, they use the Web as usual. Otherwise, they press the ‘Settings’ button to bring up the control band allowing their preferences to be modified. New users use the same control band to specify their initial preferences. At the close of each session, the user’s current settings preferences are sent back to the server and saved for the next session. Since their preferences are stored on a server, users are not tied to a particular machine. This is essential for shared machines typical of those found in classrooms and community centers.

## 3. USER FEEDBACK AND USAGE

We have been testing our software with a number of different user groups. The feedback has provided valuable insights into its utility and people’s usage patterns. It has led to many design improvements and functional enhancements.

The original deployment using the proxy server approach and a somewhat different UI [11] was deployed for six months with several older adults, mostly instructors at SeniorNet Learning Centers who taught computer courses to older adults [19]. Feedback from these users led us to abandon the proxy approach. Some of the UI design principles continued from this original work (e.g., the ability to see results of changes on a current Web

page), although the proliferation of new adaptations required abandoning the original UI in favor of the settings band at the bottom of the screen.

The current version of our project has been in use for nearly a year by a number of organizations serving seniors. By and large these older adults would not consider themselves disabled. They continue to remain active. For this population, the single most used feature is the speak text adaptation. The speak text feature employs IBM’s ViaVoice® text-to-speech capability. The implementation of this speak text feature differs from the kind of text-to-speech support used in screen readers for blind users [1], [24]. Our speak text works by having the user point the mouse at that portion of the page they want to have read aloud. As the mouse hovers over text, links, or images, the corresponding text is read aloud (for images, the ALT text that is present is spoken). This ability to point at the desired portion of the page makes usage much simpler than talking browsers designed for blind users in which navigation must be accomplished by keyboard commands.

We have found that approximately 40% of our users elect to turn on the speak text functionality. This is a very high proportion, especially when we consider that many desktop computers used by our population do not have audio support so not everyone could take advantage of this adaptation.

A more recent user group is comprised of persons who are developmentally disabled in terms of cognitive, social, and emotional development or who have some degree of mental retardation. Adults in basic literacy courses are another group currently using the software. While this software was not initially developed for these user groups, reports from organizations serving these persons indicate that several features of this technology have been especially beneficial. Many of the features for visual and cognitive difficulties (e.g., for reducing visual clutter) are used by these persons. In particular, larger font, bold sans serif fonts, increased letter and line spacing, and speak text have proven to be useful.

Reports from users with developmental disabilities and mild mental retardation have been particularly encouraging. Previously, these users have been frustrated in their attempt to deal with the text contained on Web pages. Not only was reading difficult, but vision difficulties often associated with these disabilities made viewing normal Web text difficult. For these users, the speak text feature has made it possible for them to follow along as the computer reads the text on the screen. This is beneficial for users who either cannot read well or who find it difficult to read for a long period of time. The text size and magnify features makes it possible to read without squinting. These users have been able to access the Web using the Web Adaptation features. Their reports indicate that for the first time they were able to use the Web independently. The UI was simple enough that they were able to set their own preferences. In some cases, these users were not able to read the text on the panel buttons. They understood, however, that they could click on the buttons to see the effect of the changes and, thus, were able to determine for themselves which settings were most beneficial. Interestingly, the login procedure seems to have created a sense of empowerment for those users and increased their satisfaction with their Web experience.

The one failure for ease of use appears to be the ability to access online help. This feature is utilized by clicking the ‘?’ button on the bottom left of each settings panel. Requesting this brings up

help for the current panel function. Very few users have discovered the help button on their own, although all easily discovered the < and > arrows for previous and next that appear just above the help button. We have since addressed this problem through a one page startup screen that appears the first time a user logs in (see Figure 8).

#### 4. RELATIONSHIP TO WEB ACCESSIBILITY GUIDELINES

Our approach is complementary to accessible Web page design as presented in guidelines. The majority of Web audits and checklists focus on the types of technical performance standards of Section 508 [5], [13], [18]. As mentioned earlier, these guidelines tend to address the most disabled users and provide guidelines for making Web pages capable of being rendered by assistive technology devices such as screen readers. It is useful to note that to the extent Web content providers meet these accessibility guidelines, our technology will provide better usability. For example, if no ALT tag is provided for an image, the user simply hears the word “image” when they mouse over it. However, if the image has been made accessible with an ALT tag, our software will read aloud the image description. Another example of this utilization of accessibility features is the skip-navigation feature mentioned above in relation to page linearization. These are examples in which conformance to standards makes a more usable page for others who do not rely on the assistive technology devices that the standard was designed to support. Thus, rather than obviating the need for accessible Web pages, our technology capitalizes on accessibility enhancements to give any person a more useable Web page.

It is worth noting that all the Web pages used as examples in this paper were fully compliant with accessibility guidelines. As shown here, even these pages can be difficult for persons with failing eyesight and hand control. As shown, our technology allows us to make a number of adaptations “on the fly” that greatly increase the usability of these pages for a large number of users. The types of adaptations made by our technology are consistent with guidelines that focus on functional performance standards for limited vision and input, the User Agent Accessibility Guidelines, and guidelines for Web design for older adults outlined by the National Institute on Aging [6], [18], [23]. Our software, for example, addresses the functional performance standard which directs that Web content must be viewable by a person with limited vision (defined as between 20/70 and 20/200

with correction) without relying on audio output. Screen readers do not satisfy this standard because of their reliance on audio.

One alternative to meeting this functional performance standard without software such as ours is to create shadow pages at different levels of resolution. This is generally too costly. It is also impractical in that a shadow page cannot adequately address the variability in user needs.

#### 5. FUTURE DIRECTIONS

Our software was originally designed for use in classrooms where instructors could work with new users. Moreover, the classroom setting made it so that individual users did not need to worry about issues related to software downloading and installation. As the adoption of our software has grown, usage has expanded to the home. Per our initial design consideration related to shared machines, our login procedures allow for portability such that a person could use the software initially in a classroom and then have their settings applied when home. Increasingly, however, our users never enter a class. We have learned that the software is easily used by persons at home, with the exception of the installation process. This remains a challenge as we move forward.

Having learned much about the abilities and needs of our users, we are continuing to refine our architecture and deploy a range of transformations and adaptations. The software has recently been translated into a number of other languages and we expect to learn additional requirements as we get feedback from users in other countries. We also are in the process of developing adaptations for users struggling to use the mouse.

#### 6. CONCLUDING REMARKS

With the Web becoming ever more central to commerce, business, and information delivery, the case for Web accessibility is compelling. We have argued, however, that a broadly accessible Web requires more than standards and guidelines, even those backed by regulations. Developers will be more inclined to make Web content accessible (especially legacy Web content) when it is in their economic interest to do so. Cost reduction through technology and market expansion through a broadened definition of who benefits both contribute to this.

The work discussed here addresses both the cost and benefit sides of this equation. With our technology, costs are low since most Web content needs no change. We can improve readability, reduce distraction, and filter noisy keyboard input with software running alongside existing browsers. This benefits not just those having visual and motor disabilities but anyone desiring a more usable Web experience. Older adults represent one such population. We have also seen benefits for people with limited language skills and for those having attentional and cognitive disabilities.

Standards and guidelines, backed by regulations, have produced a climate favorable to the creation of a more accessible Web. Technology can provide a cost effective means to attain not just improved accessibility, but enhanced Web usability for all.

#### 7. ACKNOWLEDGMENTS

We would like to acknowledge the contributions of our colleagues at IBM’s Watson Research Center: Jonathan Brezin, Susan



Figure 8. The Welcome screen.

Crayne, Peter Fairweather, Cal Swart, Beth Tibbitts, and Shari Trewin.

## 8. REFERENCES

- [1] Asakawa C., & Itoh, T. User interface of a Home Page Reader, in Proceedings of ASSETS'98 (Marina del Rey, CA, April, 1998), ACM Press, 149-156.
- [2] Asakawa C., & Takagi, H. Annotation-based transcoding for nonvisual Web access, in Proceedings of ASSETS'00 (Arlington, VA, November, 2000), ACM Press, 172-179.
- [3] Betsie. <http://www.bbc.co.uk/education/betsie/>
- [4] Brown, S. S., & Robinson, P. A World Wide Web mediator for users with low vision. Paper presented at the CHI'2001 Conference on Human Factors in Computing Systems Workshop No. 14. Seattle, WA. Available at <http://www.ics.forth.gr/proj/at-hci/chi2001/files/brown.pdf>
- [5] CAST Bobby. <http://www.cast.org/bobby/>
- [6] Creating Senior-Friendly Web Sites. Center for Medicare Education. Available at <http://www.medicareed.org/pdfs/ibv1n4.pdf>
- [7] Czaja, S. J. Aging and the acquisition of computer skills. In W. A. Rogers, A. D. Fisk, & N. Walker (Eds). Aging and skilled performance (pp 201 – 220). Mahwah, NJ: Lawrence Erlbaum, 1996.
- [8] Czaja, S. J. Computer technology and the older adult. In M. Helander, T. K. Landauer, & P. Prablu (Eds). Handbook of Human-Computer interaction, Second edition, (pp 797-812). Netherlands: Elsevier Science, B. V., 1997.
- [9] Digital Media Access Group. <http://www.computing.dundee.ac.uk/projects/dmag/>
- [10] EASI. <http://www.rit.edu/~easi/>
- [11] Fairweather, P. G., Richards, J. T., & Hanson, V. L. Distributed accessibility control points to help deliver a directly accessible Web. Universal Access in the Information Society, 2 (2002), 70 – 75.
- [12] Hanson, V. L., Richards, J. T., Fairweather, P. G., Brown, F., Crayne, S., Detweiler, S., Schwerdtfeger, R., & Tibbitts, B. Web accessibility for seniors. In C. Stephanidis (Ed). Universal Access in HCI: Towards an Information Society for All (pp. 663-666) Mahwah, NJ: Erlbaum. 2001.
- [13] IBM Web Accessibility Checklist. Available at <http://www-3.ibm.com/able/guidelines/web/accessweb.html>
- [14] Just Vanilla. <http://www.justvanilla.com>
- [15] Making Your Web Site Senior Friendly: A checklist. National Institute on Aging, Available at <http://www.nlm.nih.gov/pubs/checklist.pdf>
- [16] National Institute on Aging. Older Adults and Information Technology: A Compendium of Scientific Research and Web Site Accessibility Guidelines. Bethesda, MD: National Institute on Aging, 2002.
- [17] Richards, J. T., Hanson, S., & Trewin, S. Adapting the Web for Older Users. 2nd International Conference on Universal Access in Human-Computer Interaction (pp. 892 – 896), 2003.
- [18] Section 508: The Road to Accessibility. <http://www.section508.gov/>
- [19] SeniorNet. <http://www.seniornet.org>
- [20] Takagi, H., Asakawa, C., Fukuda, K., & Maeda, J. Site-wide annotation: reconstructing existing pages to be accessible, in Proceedings of ASSETS'02 (Edinburgh, Scotland, July, 2002), ACM Press, 81 – 88.
- [21] Trewin, S. Configuration agents, control and privacy, in Proceedings of CUU'00 (Arlington, VA, November, 2000), ACM Press, 9-16.
- [22] Trewin, S., & Pain, H. A model of keyboard configuration requirements, in Proceedings of ASSETS'98 (Marina del Rey, CA, April, 1998), ACM Press, 173 - 181.
- [23] User Agent Accessibility Guidelines. Available at <http://www.w3.org/TR/2002/PR-UAAG10-20021016/>
- [24] Zajicek, M. (2001). Supporting older adults at the interface. In C. Stephanidis (Ed.). Universal Access in HCI: Towards an Information Society for All (pp. 454 – 458) Mahwah, NJ: Erlbaum. 2001.
- [25] Zajicek, M., Powell, C., & Reeves, C. A Web navigation tool for the blind, in Proceedings of ASSETS'98 (Marina del Rey, CA, April, 1998), ACM Press, 204-220.