

Web Accessibility Tools: InFocus & LIFT for DreamWeaver

High Tech Center Training Unit

Of the California Community Colleges at the
Foothill-De Anza Community College District

21050 McClellan Road
Cupertino, CA 95014
(408) 996-4636

www.htctu.net

Table of Contents

INFOCUS 4.2	5
LIFT 1.4.1	15
STANDARDS FOR ELECTRONIC AND INFORMATION TECHNOLOGY: AN OVERVIEW	25
FEDERAL ACCESSIBILITY STANDARDS FOR WEB-BASED INTRANET AND INTERNET INFORMATION AND APPLICATIONS.....	31
WEB CONTENT ACCESSIBILITY GUIDELINES (WCAG) 1.0	35
WAI PRIORITY 1 CHECKPOINTS.....	36
WEB RESOURCES	39

InFocus 4.2

Publisher:

SSB Technologies
1000 Sansome St, Suite 280
San Francisco, CA 94111
(415) 975-8000
Fax: (415) 732-7508
<http://www.ssbtechnologies.com>

Retail Cost: \$899 (Government/Education based price)

System Requirements:

1. A PC-based computer running Windows 98/ME/2000/XP. It is recommended to have a Pentium III 450MHz processor or faster.
2. 256MB RAM or greater and 43MB Hard Drive space available.
3. Color monitor capable of 800x600 or greater and mouse/pointing device.
4. Internet Explorer 5.5 or later (not required but recommended to take advantage of certain evaluation features in InFocus).

Description:

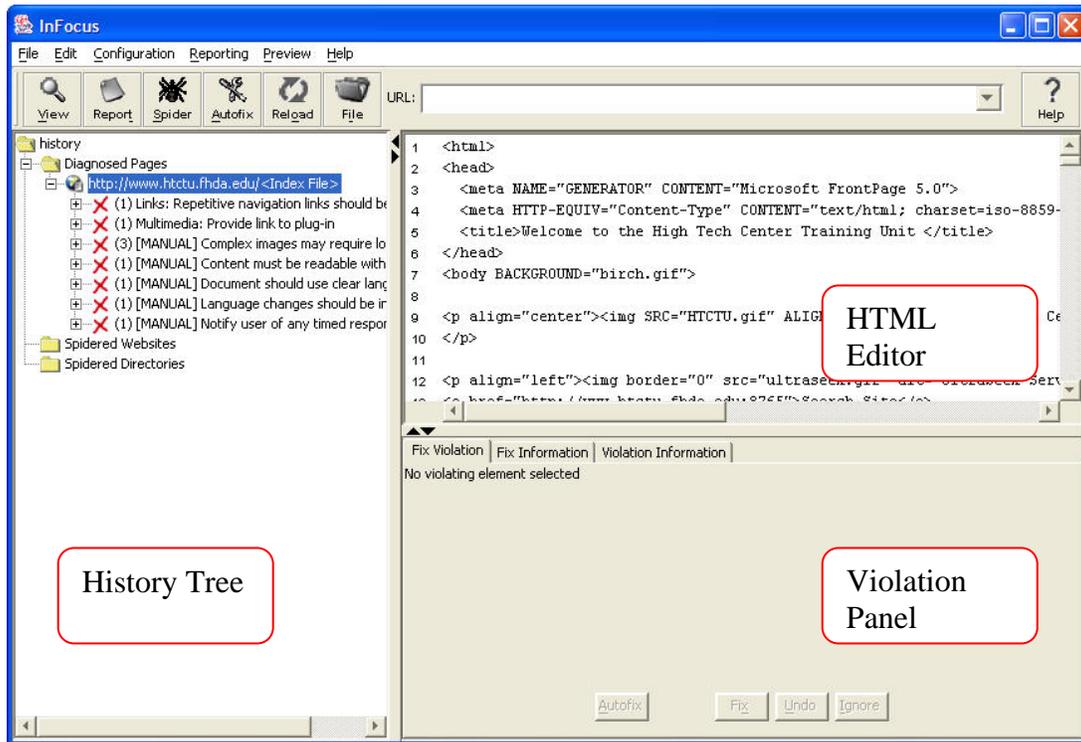
InFocus is a JAVA-based application designed to assess web page and website accessibility and provide mechanisms to correct accessibility errors. InFocus provides the capability to evaluate a website stored locally or on a remote server, analyze the web pages for accessibility errors against the US Section 508 Standards or WCAG 1.0 Guidelines, and then prompt the developer for changes to the webpage to make the content conform to web accessibility standards. InFocus can also save evaluation reports of different websites and export the web accessibility evaluations into HTML, XML, MS Excel formats to assist in the tracking of website retrofitting. In addition to the InFocus tool, SSB Technologies currently provides a web-based evaluation tool called AskAlice, which will evaluate your website and provide feedback as to the accessibility issues that may exist. Visit the SSB Technologies website for additional information.

Using InFocus

InFocus provides a number of evaluation and repair tools to streamline the web accessibility evaluation and retrofitting process. InFocus provides tools that can render a webpage in grayscale to assess color issues, assist in the repair of javascripted functions

to include keyboard access, and update alternative text descriptions across a website. If you have questions that are not addressed in this training, please use the **Help** menu available from the menu bar after starting InFocus.

InFocus Interface



History Tree Pane:

The History Tree pane provides a history of all the pages that have been diagnosed or "spidered". Identified errors are listed in a hierarchical manner from the parent file. Select the "plus" symbol to identify child elements or the specific line of code where the error occurs. The History Tree will also identify if the error in question is a Manual check requiring the content developer to assess the accessibility of this error independently of the InFocus tool.

HTML Editor Pane:

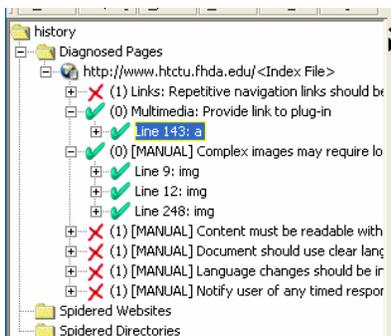
The HTML editor provides an HTML-view of the webpage that you have assessed. While you are not able to make changes by hand to the code, it is possible to use the Violation Panel to address web accessibility errors identified in the HTML Editor Panel. Errors will appear highlighted in yellow in the HTML Editor Panel (the highlight color may be changed by selecting **Choose Highlight Color** from the under **Configuration** on the menu bar).

Violation Panel Pane:

The Violation Panel allows users to make changes to the web page code, learn about how InFocus will fix the web page error, or learn more about why this is a web accessibility error and what guideline or standard was identified during the evaluation process. The "Fix Violation" tab provides the mechanism to make changes to the HTML code. The "Fix Information" tab provides information as to exactly how InFocus will make the change to the HTML code to make the web page more accessible. The "Violation Information" tab identifies the US Section 508 Standard or W3C WCAG 1.0 Guideline that was violated and explains why this fix is necessary to make the web page more usable (and accessible) to someone using assistive computer technology.

To start evaluation on a webpage or website:

1. Select **File** from the menu bar.
2. Choose **Open File** to assess a webpage or **Open URL** to assess a website.
3. Browse to the selected HTML file you wish to evaluate or enter the correct URL of the website you wish to evaluate. When you enter the URL of the website, InFocus will only evaluate the main page (i.e., index file) of the website. To evaluate the main page and sub-pages of the website, use the Spider function.
4. The website main page or selected file will appear in the left side of the screen under **Diagnosed Pages**. Select the "plus" symbol to expand the page and review the accessibility errors.
5. Web accessibility errors will appear as a red X while corrected web accessibility errors will appear as a green checkmark.

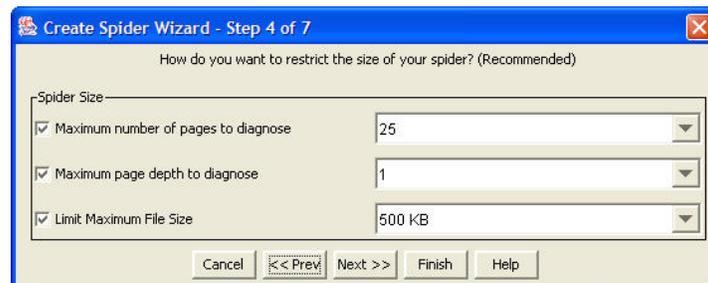


Using the Spider to Evaluate a Website

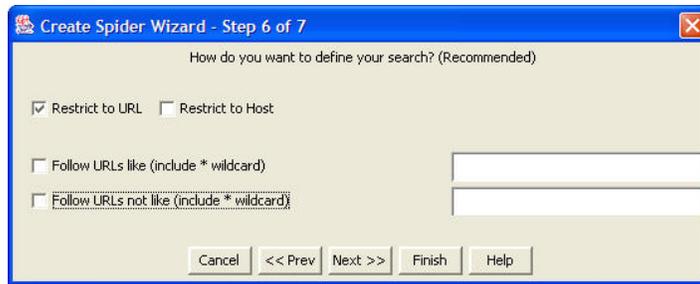
The Spider function provides a developer to "dive into" the website and evaluate web page accessibility throughout a specified number of sub-pages or directories. Before starting a Spider, InFocus will begin a seven-step process that will customize how the Spider function will evaluate the website and how "deep" to search. If you are unsure of what options to select, you can click on the **Help** button, which will provide specific

information regarding the Spider function. The Spider function can be used on live websites or those stored locally on the computer.

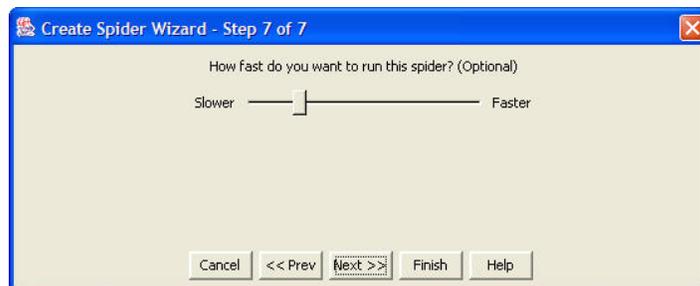
1. To run the Spider function, select **Edit** from the menu bar and choose **Create Spider**, or select the **Spider** button from the toolbar.
2. Identify the URL for a website or the folder for a local directory. Select **Next**.
3. Identify a Login name and password if you require authentication to access the live website.
4. Identify the type of report that you would like to run (e.g., HTML, MS Excel, XML, etc.) or if you wish to add the pages to the History. Adding the pages to the History will allow you to run future reports (it is recommended to select both checkboxes in order to give the most flexibility when running the reports).
5. Identify the parameters for the Spider function. This will include the number of pages, how many links off the main page and what the maximum file size limit. HTML file sizes in excess of this value will not be evaluated. If you are only interested in assessing the main page, use the **Open URL** function instead of running a Spider.



6. In Step 5, identify the types of files that you would like the Spider to check or avoid. This can help widen or narrow your search parameters when Spidering a website.
7. Step 6 will allow you to refine the Spidering parameters and allow you to seek out certain pages that you wish to evaluate (or avoid). By selecting the **Restrict to URL** option, InFocus will only seek out web pages that have the root URL you provided in Step 1. By selecting **Restrict to Host**, InFocus will seek out all pages that contain the domain name within the specified URL. If you are Spidering a page that includes a port number, use the **Restrict to Host** option.



8. In the final step, identify how fast you wish the Spider to run. Setting the slider to the **Faster** setting will put a high demand on the system resources, while moving the slider to the **Slower** setting will put a lower strain on the system resources (but take the Spider longer to evaluate the pages).



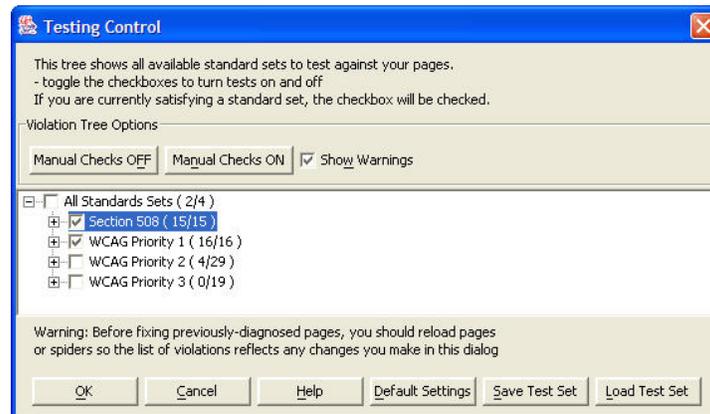
Setting your Evaluation Criteria

InFocus will assess web pages and websites based on the US Section 508 Standards as well as the W3C Web Content Accessibility Guidelines 1.0 (WCAG 1.0). In addition to both sets of accessibility techniques, InFocus provides the capability to selectively apply accessibility standards to the evaluation process.

To modify the evaluation techniques:

1. Select **Testing Control** from under **Configuration** on the menu bar, or the keyboard command **Ctrl+T**.
2. Select the set of techniques that you would like to evaluate your web content against. If you would like to only evaluate your web content against the US Section 508 Standards, then uncheck the WCAG Priority 1, WCAG Priority 2, and WCAG Priority 3 checkboxes.
3. You may customize the checklist to only address one specific element of the US Section 508 Standards or W3C WCAG 1.0. This can aid in the evaluation of a site for a specific web accessibility error (e.g., a lack of alternative text or forms labeled correctly). Uncheck the boxes of the accessibility techniques that you do not wish to evaluate. Select **Save Test Set** to save a copy of this test set.

- If you have made a number of changes, but wish to return to the default settings (i.e., evaluate against the US Section 508 Standards and W3C Priority 1 Guidelines), select the **Default Settings**. This will return your evaluation set to the original installation techniques. For more information about the US Section 508 Standards and the W3C WCAG 1.0 web accessibility techniques, please refer to the additional documentation in this manual.



Generating Evaluation Reports

Upon completion of a Spider, InFocus can assemble the accessibility evaluation into various reporting formats. If you chose to set your Spider to **Generate Report after Run** and did not select the **Add Generated Pages to History** (see Step 3 of the Create Spider Wizard), you will *not* have the ability to generate reports after completing the Spider.

To generate a report:

- First run a Spider (make sure to select **Add Generated Pages to History** in Step 3 of the Create Spider Wizard) or evaluate a single Web page.
- Select **Reporting** from the menu bar.
- Choose either **Generate Reports** or **Generate Charts**. If you choose to generate a report, you will have the option of choosing a HTML, Excel, EARL, or XML report format. If you choose to generate to a chart, you will have the option of choosing a Pie Chart or a Bar Chart.



4. If you wish to generate a report or a chart that includes multiple pages, hold the "Shift" key and select additional pages under the **Spidered Pages** or **Diagnosed Pages** directory.
5. Charts – A Pie Chart will identify the pages and number of violations. A Bar Chart will identify the total fixed and unfixed violations for a page or range of pages.
6. Reports – Reports can be generated in HTML, MS Excel, XML, or in the EARL format. EARL (Evaluation and Reporting Language) is a format developed by the World Wide Web Consortium for the exchange of web accessibility evaluation information. Reports will be saved to your Temp directory unless you choose otherwise.

Advanced Evaluation Functions

InFocus provides several additional evaluation tools designed to check several manual checkpoints of the web accessibility evaluation criteria. Under **Preview** in the menu bar, there are five options that will provide a preview of the web page with the specified function enabled or disabled.



Style sheets off (Ctrl + F5)

Turning off the style sheets will provide the developer with information as to what the page will look like if style sheets are disabled by a user or if a text-only browser is being used to view the web page. If style sheets are turned off, it is important that the page can still be read and interpreted correctly.

Linearized (Ctrl + F6)

The Linearization function will remove graphics and formatting from the web page and present the information as left-justified text. Assistive computer technology (specifically screen-readers) will read information on a web page in this format even though the page may be visually formatted in a different reading order. The key element when using the linearization tool is to assess the reading order of the on-screen content and insure the text flow will be correct in the Linearized version as well as the non-linearized version. It is important to note that this is not how every screen-reader will interpret the page, only an example of how assistive computer technology may operate.

Grayscale (Ctrl + F7)

Grayscale provides the web page author with information as to how the web page would look when viewed without color. Of particular importance is that the web page does not communicate information through color alone (e.g., "All the red hyperlinks take you to the Home page"). Individuals not able to visually process certain colors would have difficulty interpreting the information on the page. Using the Grayscale function provides authors with a chance to check if color is being used to convey information.

JavaScript Off (Ctrl + F8)

The JavaScript function allows the developer to assess the web page when JavaScripting is turned off. It is important to make sure the page is still readable and can be interpreted when JavaScripting is turned off in the browser.

Text-Only (Ctrl + F9)

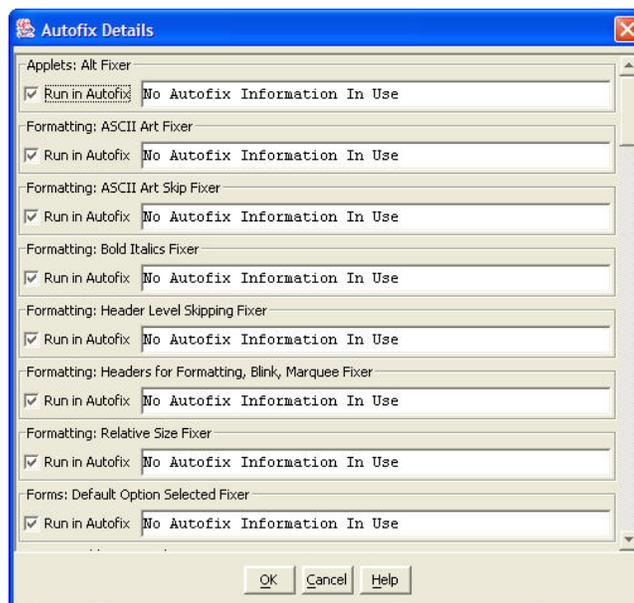
The Text-Only function displays only the text blocks of the page. Any other text elements (e.g., alternative text) will not be displayed on the screen.

Autofix Violation Tool

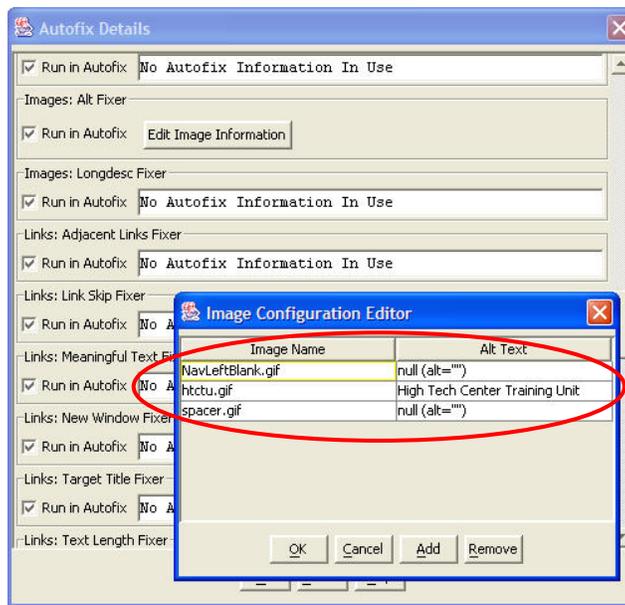
The Autofix Violation tool provides a web page developer to fix common violations without having to sequentially check each reported violation. For example, if a graphic is used as a spacer image, it is still necessary to provide alternative text (in the form alt=""). Not providing alternative text for spacer images on a web page causes assistive computer technology (i.e., screen-reader applications) to interpret the text associated with the tag. This generally results in the screen-reader "reading" the associated URL to the user.

To check the Autofix Options:

1. Select **Configuration** from the menu bar and choose **View Autofix Details**.



2. If you have not created any autofixes during the repair of a website/web page, then you will not be able to view any of the current autofixes.
3. The advantage of the Autofix Violation tool, is that it provides a mechanism in which to automatically repair certain errors, such as invalid alternative text, across a website. While you cannot make changes in the Autofix Details window, you can specify, the alternative text to be used for certain filenames.
4. In the Autofix Details window, scroll to the "Alt Fixer" option and select the "Edit Information Button". This will provide you with an interface in which to enter the image name (i.e., file name of image) and the appropriate alternative text for the image.



5. To perform an Autofix on a selected web page or set of web pages, select **Edit** from the menu bar and choose **Autofix Violations**. Identify the pages you wish to Autofix and select "OK". The Autofix Violation Tool will then make the necessary changes to the HTML code.

LIFT 1.4.1

Publisher:

UsableNet
83 Mercer Street, 2nd Floor
New York, NY 10012
(212) 965-5388
Fax: (212) 965-5391
contact@usablenet.com or <http://www.usablenet.com/>

Retail Cost:

LIFT for Dreamweaver or LIFT for FrontPage - \$299
Dreamweaver NN/g version - \$549
(Education pricing available)

System Requirements:

1. A PC/Mac-based computer running Windows 95 or later, or OS 8.6, 9.1, or OSX 10.1. For both configurations, greater than 64MB of RAM (128 recommended) and 20 MB hard disk drive is required.
2. LIFT for Dreamweaver requires Dreamweaver MX, Dreamweaver 4.0, or UltraDev 4.0. LIFT for FrontPage requires MS FrontPage 2000/2002, IE 5 or later (LIFT for FrontPage does not support Windows 95).
3. Color monitor capable of 800x600 at 256 colors or greater.
4. LIFT Machine is an enterprise level website evaluation and reporting application that generates results of website accessibility issues. Contact UsableNet for additional information regarding this tool for website evaluation and reporting.

Description:

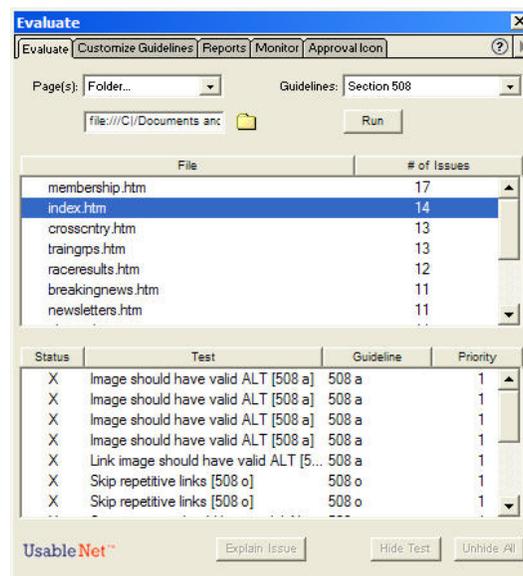
LIFT for Dreamweaver and LIFT for FrontPage is a Windows and Mac-based application designed to assess web page and website accessibility and provide tools to correct accessibility errors. LIFT provides the capability to evaluate a website stored locally and analyze the web pages for accessibility errors against the US Section 508 Standards or WCAG 1.0 Guidelines. LIFT then prompts the developer for changes to the webpage to make the content conform to web accessibility standards. LIFT provides the flexibility to customize reporting guidelines to focus error reports to identify specific website errors as well as monitor web pages under customized evaluation rules. LIFT can also export the web accessibility evaluations and reports into HTML or XML format to assist in website retrofitting.

Evaluating Web Content

LIFT provides the web content developer with additional tools to evaluate web content against the W3C WCAG 1.0 techniques or against the US Section 508 Standards for Electronic and Information Technology. Additionally, it is possible to evaluate web pages against tests that examine alternative text and accessibility of data tables.

To evaluate a web page or website:

1. Begin Dreamweaver. Select **LIFT** from the menu bar (if this option is not available, please see the LIFT documentation for installing LIFT into Dreamweaver or FrontPage). Choose **Evaluate**.
2. The **Evaluate** interface will appear. If your website exists as a folder, choose **Folder** from the drop down menu next to **Page(s)**. You may also choose a local site (as defined by Dreamweaver) or selected files in a site.
3. After selecting the content you wish to evaluate, select the **Guidelines** that you prefer to evaluate against and select **Run**. This will perform an evaluation against the specific techniques you chose and identify the number of accessibility issues reported by page.



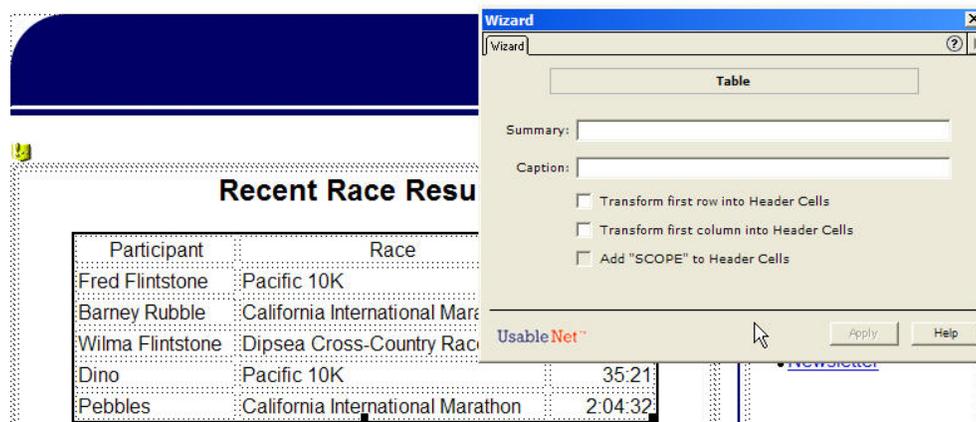
4. Selecting an accessibility violation in the lower pane will open the HTML file and shift focus in Dreamweaver to the element that did not pass the evaluation test.
5. Make the appropriate correction to the HTML code in the authoring tool interface. If you are unsure as to how to correct the accessibility violation, select the **Explain Now** button for additional information.

Using the Fix Wizard

The Fix Wizard can assist in the development and retrofit of web pages/websites by prompting the user to make the necessary changes to the HTML code. The Fix Wizard tool can assist in the repair of images lacking alternative text, repairing multimedia accessibility, making data tables accessible, and making on-line forms more accessible. If you have difficulty making a repair or are unsure as to how to proceed, select the **Help** button from within the Fix Wizard interface.

To start the Fix Wizard:

1. First perform an evaluation on the local site or folder. While not necessary, it is helpful to have the evaluation to refer to when making changes.
2. Identify the accessibility violation in the HTML code. If the accessibility error involves a data table, select the entire table in the authoring tool (or the entire image, etc.). Complete the necessary fields as requested by the Fix Wizard tool.



3. Once you have completed the necessary fields for the accessibility repair, select **Apply**.

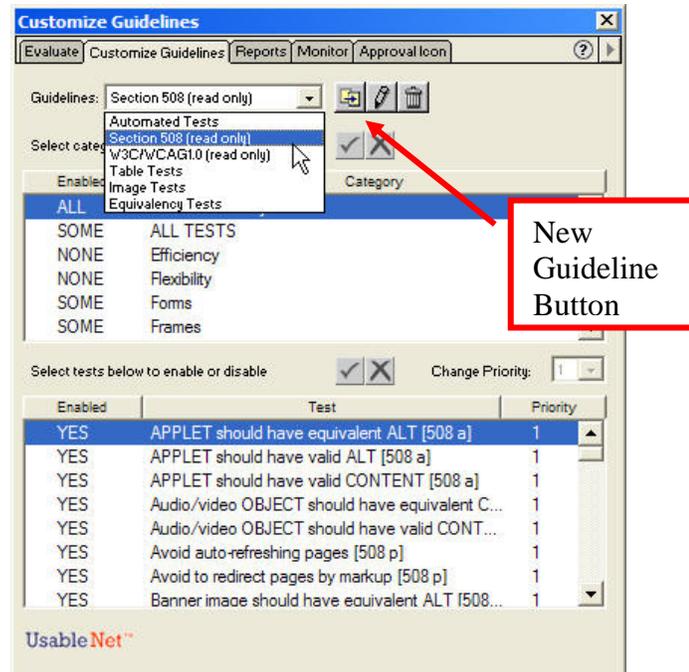
When the Fix Wizard tool interface is "grayed out", then a web page element has been selected that either cannot be fixed with the Fix Wizard tool or there is no accessibility violation with the particular web page element. However, in order to be sure that the web content is accessible, it is recommended to run the accessibility evaluation on the particular page(s) in question.

Customizing the Evaluation Techniques

LIFT provides the flexibility to evaluate web page/website content according to the W3C WCAG 1.0 techniques or the US Section 508 Standards for Electronic and Information Technology. It is also possible to run several customizable evaluation tests that may include some, all, or a combination of the accessibility techniques. Once you have customized a specific set of accessibility techniques, you will be able to refer to this set of techniques under the **Evaluate** and **Monitor** tabs.

To customize the evaluation techniques:

1. Select **LIFT** from the menu bar, and choose **Customize Guidelines**. Alternatively, you may select **Customize Guidelines** tab from the LIFT palette.
2. Choose the techniques that you would like to evaluate the web page content against. To customize the accessibility techniques, choose the **New Guideline** button. This will create a copy of the currently selected accessibility techniques. You may then rename this copy in order to refer to this set of techniques at a later time.



3. In the lower pane, identify which tests that desire to be enabled or disabled in your evaluation techniques. Use the red X to disable a specific web accessibility test and the green checkmark to enable a specific web accessibility test. This provides the option of creating a set of evaluation techniques that will only evaluate web content for a specific set of accessibility errors.
4. Some tests may be linked and you will be unable to turn off a specific US Section 508 Standards test while enabling a similar W3C WCAG 1.0 specific test.

Generating a Report

Reports can be generated from LIFT to document the current accessibility of a website and track the progress towards making a website fully accessible. Reports may be generated in both a summary format (i.e., Executive Summary) or in a detailed page report (i.e., Page Details) to assess the accessibility of the web content. Additionally, reports can be generated in either a HTML or XML format as well as providing documentation for the report author and date stamp.

To generate a report using LIFT:

1. First, run an evaluation on the web page/website. An evaluation must be completed before a report can be generated. For information on how to run an evaluation, refer to the section "Evaluating Web Content".
2. After the evaluation has been completed, you may generate a report. An Executive Summary report provides a general overview of the accessibility of a website, whereas the Page Detail report identifies the specific accessibility violations for each page evaluated.
3. Choose the report format of HTML or XML. XML provides flexibility to import the evaluation report into a database for tracking purposes.

The screenshot shows the 'Reports' window in the UsableNet application. The window title is 'Reports' and it has a menu bar with 'Evaluate', 'Customize Guidelines', 'Reports', 'Monitor', and 'Approval Icon'. The 'Reports' menu is currently selected. The main content area includes a 'Report Type' dropdown menu set to 'Executive Summary', a 'Format' dropdown menu set to 'HTML', an 'Author' text box containing 'Sean Keegan', a 'Date' text box containing 'Thursday, May 29, 2003 12:05:43', and a 'Site' text box containing 'Lessons - Dreamweaver'. To the right of these fields is a 'Show:' section with five checkboxes: 'Accessibility Tests Only' (unchecked), 'Failed Tests' (checked), 'Manual Tests' (checked), 'Passed Tests' (unchecked), and 'Not Applicable Tests' (unchecked). Below the 'Show:' section is a 'Notes' section with a text area containing the text 'This is the first review of the website with no accessibility violations fixed.' At the bottom left of the window is the 'UsableNet' logo, and at the bottom right is a 'Create Report' button.

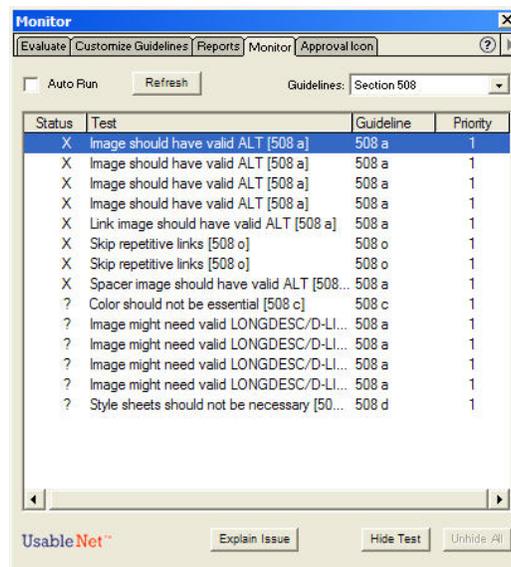
4. Under the **Show** heading, you can filter the report information based on manual tests, passed tests, and accessibility tests. Generally, you will be interested in Failed Tests for tracking results from evaluation to evaluation.
5. Enter information into the **Notes** area at the bottom of the "Reports" window that you wish to use to describe the report content.
6. Select the **Create Report** button to generate a report. If the report format you chose was HTML, then a browser will open with the results of the report.

Monitor Window

The **Monitor** window in LIFT assesses the current web page for potential accessibility violations and prompts the developer for corrections during the build process. If an accessibility violation is selected in the reporting area, the focus of the authoring tool will shift to the accessibility violation in question for repair. If more information is needed, a developer can select the **Explain Issue** button, which will provide additional information as to the changes necessary to make the element more accessible and usable. It is also possible to have the web page checked according to the customized evaluation techniques under the **Customize Guidelines** tab.

To use the Monitor window:

1. Open a web page in the HTML authoring tool that you wish to monitor.
2. Select **Monitor** from **LIFT** on the menu bar (or select the **Monitor** tab from the LIFT window).



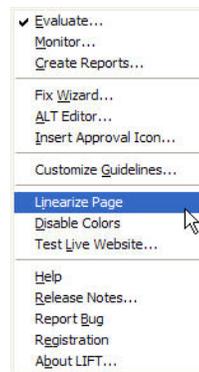
3. Select **Refresh** to load the current accessibility violations for the web page. The violations identified will be based on the **Guidelines** setting. You can customize the evaluation guidelines under the **Customize Guidelines** tab.
4. Select the **Auto Run** checkbox if you would like the monitor window to automatically keep track of the accessibility/usability violations to the current web page. If you do not select this option, it will be necessary to select the **Refresh** button to reveal any repairs you have made to the web content.
5. Select the **Explain Issue** button at the bottom of the window to learn more about the necessary changes to the web page to increase accessibility/usability of the content.

Advanced Evaluation Features

LIFT provides additional evaluation features that can aid a developer in creating accessible and usable web content. The **Linearize Page** function provides a version of the web page if viewed through a text-only browser (or what a screen-reader might interpret). A critical element with the **Linearize Page** function is that the reading order of the page is maintained, such that a web page with multiple columns will be linearized in a manner that provides for a coherent reading experience. A second feature is the **Disable Colors** function, which renders the web page as if viewed through a grayscale filter. The **Disable Colors** filter allows a web page creator to assess if the web content relies on color to convey meaningful information. It is an accessibility violation (and a usability violation) to convey information through color alone.

To use the **Linearize Page** function:

1. Open the web page that you wish to linearize in your HTML authoring tool (i.e., Dreamweaver or FrontPage).
2. From the **LIFT** menu bar, choose **Linearize Page**.

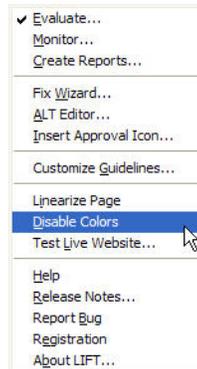


3. A web browser will begin and the web page currently open in your HTML authoring tool will be rendered in a linearized fashion in the new browser window. It is important to check for reading order as well as the proper alternative text for images.

To use the **Disable Colors** function:

1. Open the web page that you wish to linearize in your HTML authoring tool (i.e., Dreamweaver or FrontPage).

2. From the **LIFT** menu bar, choose **Disable Colors**.



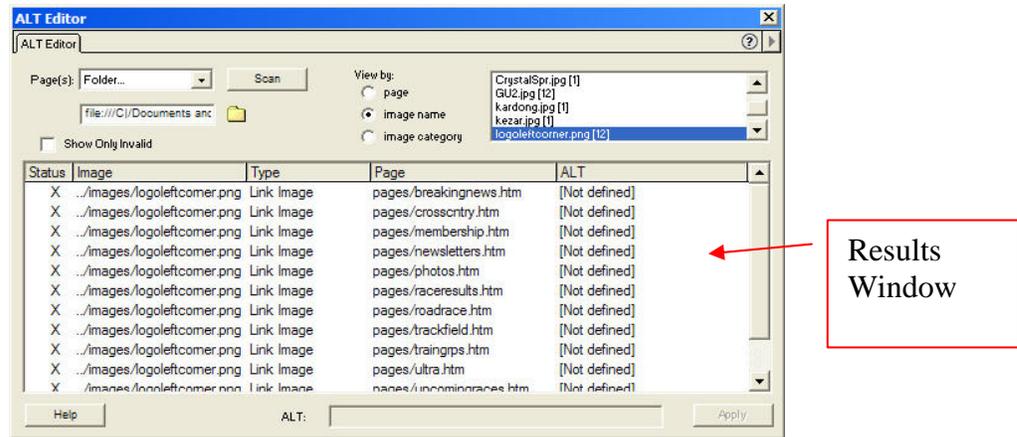
3. A web browser will begin and the web page currently open in your HTML authoring tool will be rendered with a grayscale filter applied in the new browser window. It is important to check for the use of color conveying information with the filter applied.

The Alt Editor Tool

Providing alternative text for images and graphic elements within a web page is a necessary step towards making web content accessible and usable. Websites that contain a number of graphic elements can be difficult to retrofit as the alternative text needs to be entered for each image file. This process can become tedious when the same graphic element is used on multiple pages and the same alternative text is required for each element. Using the **Alt Editor** tool in LIFT allows a developer to quickly and easily retrofit graphic content throughout a website without opening each web page and identifying each graphic element requiring alternative text. Additionally, images used as "spacers" or visual "eye-candy" that do not require alternative text can be easily retrofitted to include the null, or blank, alt-tag.

To use the Alt Editor tool:

1. Open the HTML authoring tool of your choice (i.e., Dreamweaver, FrontPage).
2. Select **LIFT** from the menu bar and choose **Alt Editor**. This will open the **Alt Editor** window. If you have not performed a scan, then you will not see information in the results window.
3. To perform a scan of all the images in your website, select the web page, local Dreamweaver site, or the folder in which your web content resides and press **Scan**. This will check all the web page images and identify the images in the results window.



4. Once you have performed a scan of the web pages, you can view the images by page, by image name, or by image category. To make global corrections (i.e., same alternative text across different web pages), view by the image name and highlight all the appropriate images in the results window (HINT: use the shift key to select multiple instances of the same image). Enter the appropriate alternative text for the image in the **ALT:** field and select **Apply**. This will apply the alternative text you supplied to all the highlighted images across your website.
5. Images that contain a "X" in the Status field are images that are lacking a valid alt-tag. If the results window has a mixed array of images that contain valid and invalid alternative text, it is possible to show only the invalid images. Select the **Show Only Invalid** checkbox to sort the results window and display only the invalid alternative text.
6. If you require help or would like additional information about the Alt Editor tool, select the **Help** button in the lower left corner.

Standards for Electronic and Information Technology: An Overview

The Access Board is an independent Federal agency devoted to accessibility for people with disabilities. On December 21, 2000, the Board issued accessibility standards for electronic and information technology under section 508 of the Rehabilitation Act, as amended. The Board also develops and maintains accessibility guidelines for the built environment, transit vehicles, and telecommunications equipment under other laws and enforces design standards for federally funded facilities. Presented here is an overview of the new standards for electronic and information technology and section 508. Questions about the 508 standards should be sent to 508@access-board.gov.

The Law: Section 508

In 1998, Congress amended the Rehabilitation Act and strengthened provisions covering access to information in the Federal sector. As amended, section 508 of the Rehabilitation Act requires access to the Federal government's electronic and information technology. The law covers all types of electronic and information technology in the Federal sector and is not limited to assistive technologies used by people with disabilities. It applies to all Federal agencies when they develop, procure, maintain, or use such technology. Federal agencies must ensure that this technology is accessible to employees and the public to the extent it does not pose an "undue burden." The law directs the Access Board to develop access standards for this technology that will become part of the Federal procurement regulations.

The scope of section 508 is limited to the Federal sector. It does *not* apply to the private sector, nor does it generally impose requirements on the recipients of Federal funds. (However, States receiving assistance under the Assistive Technology Act State Grant program are required to comply with section 508 and the Board's standards, according to the Department of Education, which administers the Act. The Department plans to issue guidance on how the standards apply to the States under the Assistive Technology Act. For further information, e-mail the [Department of Education](#) or call (202) 205-5666 (voice) or 202-205-5516 (TTY)).

Development of Standards

Shortly after the law was enacted, the Access Board created an advisory committee to develop recommendations on the standards to be developed. In May 1999, the Electronic and Information Technology Access Advisory Committee (EITAAC) completed its work and presented its recommendations to the Board. The committee consisted of 27 representatives from industry, various disability organizations, and other groups with an interest in the issues to be addressed. On March 31, 2000, the Board published proposed standards based closely on the committee's report. The proposed standards were available for public comment for 60 days through publication in the *Federal Register*. The Board sought information and comment on various issues through questions it posed in a

discussion provided in the proposed rule. Over 100 individuals and organizations submitted comments on the standards. Comments were submitted by Federal agencies, representatives of the information technology industry, disability groups, and persons with disabilities. The Board finalized the standards according to its review of the comments and republished them in the *Federal Register*. The final standards, which will become part of the Federal Acquisition Regulations, will help Federal agencies determine whether or not a technology product or system is accessible.

Enforcement and Effective Date

Section 508 uses the Federal procurement process to ensure that technology acquired by the Federal government is accessible. The law also sets up an administrative process under which individuals with disabilities can file a complaint alleging that a Federal agency has not complied with the standards. This process uses the same complaint procedures established under section 504 of the Rehabilitation Act (which covers access to Federally funded programs and services). Individuals may also file a civil action against an agency to seek injunctive relief and attorney's fees (but not compensatory or punitive damages). The enforcement provisions of section 508 take effect six months from the date the Board published its final standards. The Board published its standards on December 21, 2000. Therefore, the enforcement provisions of section 508 are effective as of June 21, 2001. (Originally, they were to take effect August 7, 2000, but section 508 was further amended to allow time for compliance after publication of the standards).

By statute, the enforcement provisions of section 508 apply only to electronic and information technology *procured on or after the effective date*. As a result, section 508 does not authorize complaints or lawsuits to retrofit technology procured before this date to meet the Board's standards. However, even though section 508's enforcement mechanisms apply only to *procurement*, the law does require access to technology *developed, used or maintained* by a Federal agency. Further, other sections of the Rehabilitation Act require access to Federal programs (section 504) and accommodation of Federal employees with disabilities (sections 501 and 504); it is possible that Federal agencies will use the Board's section 508 standards as a yardstick to measure compliance with these other sections of the law.

"Undue Burden"

A Federal agency does not have to comply with the technology accessibility standards if it would impose an undue burden to do so. This is consistent with language used in the Americans with Disabilities Act (ADA) and other civil rights legislation, where the term 'undue burden' has been defined as "significant difficulty or expense." However, the agency must explain why meeting the standards would pose an undue burden for a given procurement action, and must still provide people with disabilities access to the information or data that is affected.

The Standards

General (Subpart A)

The standards define the types of technology covered and set forth provisions that establish a minimum level of accessibility. The application section (1194.2) outlines the scope and coverage of the standards. The standards cover the full range of electronic and information technologies in the Federal sector, including those used for communication, duplication, computing, storage, presentation, control, transport and production. This includes computers, software, networks, peripherals and other types of electronic office equipment. The standards define *electronic and information technology*, in part, as "any equipment or interconnected system or subsystem of equipment, that is used in the creation, conversion, or duplication of data or information."

Subpart A also explains what is exempt (1194.3), defines terms (1194.4), and generally recognizes alternatives to what is required that provide equal or greater access (1194.5). Consistent with the law, the standards exempt systems used for military command, weaponry, intelligence, and cryptologic activities (but not routine business and administrative systems used for other defense-related purposes or by defense agencies or personnel). The standards also exempt "back office" equipment used only by service personnel for maintenance, repair, or similar purposes.

The standards cover technology procured by Federal agencies under contract with a private entity, but apply only to those products directly relevant to the contract and its deliverables. An exception clarifies that the standards do not apply to technology that is incidental to a Federal contract. Thus, those products that are not specified as part of a contract with a Federal agency would not have to comply with the standards. For example, a firm that produces a report for a Federal agency under a contract would not have to procure accessible computers and word processing software even if they were used exclusively for the contract; however, compliance would be required if such products were to become the property of the Federal agency as contract deliverables or if the Federal agency purchased the products to be used by the contractor as part of the project. If a Federal agency contracts with a firm to develop its web site, the standards would apply to the new web site for the agency but not to the firm's own web site.

Technical Standards (Subpart B)

The standards provide criteria specific to various types of technologies, including: software applications and operating systems; web-based information or applications; telecommunication products; video and multimedia products; self contained, closed products (e.g., information kiosks, calculators, and fax machines); desktop and portable computers.

This section provides technical specifications and performance-based requirements, which focus on the functional capabilities of covered technologies. This dual approach recognizes the dynamic and continually evolving nature of the technology involved as well as the need for clear and specific standards to facilitate compliance. Certain provisions are designed to ensure compatibility with adaptive equipment people with

disabilities commonly use for information and communication access, such as screen readers, Braille displays, and TTYs.

Software Applications and Operating Systems (1194.21)

Most of the specifications for software pertain to usability for people with vision impairments. For example, one provision requires alternative keyboard navigation, which is essential for people with vision impairments who cannot rely on pointing devices, such as a mouse. Other provisions address animated displays, color and contrast settings, flash rate, and electronic forms, among others.

Web-based Intranet and Internet Information and Applications (1194.22)

The criteria for web-based technology and information are based on access guidelines developed by the Web Accessibility Initiative of the World Wide Web Consortium. Many of these provisions ensure access for people with vision impairments who rely on various assistive products to access computer-based information, such as screen readers, which translate what's on a computer screen into automated audible output, and refreshable Braille displays. Certain conventions, such as verbal tags or identification of graphics and format devices, like frames, are necessary so that these devices can "read" them for the user in a sensible way. The standards do not prohibit the use of web site graphics or animation. Instead, the standards aim to ensure that such information is also available in an accessible format. Generally, this means use of text labels or descriptors for graphics and certain format elements. (HTML code already provides an "Alt Text" tag for graphics which can serve as a verbal descriptor for graphics). This section also addresses the usability of multimedia presentations, image maps, style sheets, scripting languages, applets and plug-ins, and electronic forms.

The standards apply to Federal web sites but not to private sector web sites (unless a site is provided under contract to a Federal agency, in which case only that web site or portion covered by the contract would have to comply). Accessible sites offer significant advantages that go beyond access. For example, those with "text-only" options provide a faster downloading alternative and can facilitate transmission of web-based data to cell phones and personal digital assistants.

Telecommunications Products (1194.23)

The criteria of this section are designed primarily to ensure access to people who are deaf or hard of hearing. This includes compatibility with hearing aids, cochlear implants, assistive listening devices, and TTYs. TTYs are devices that enable people with hearing or speech impairments to communicate over the telephone; they typically include an acoustic coupler for the telephone handset, a simplified keyboard, and a visible message display. One requirement calls for a standard non-acoustic TTY connection point for telecommunication products that allow voice communication but that do provide TTY functionality. Other specifications address adjustable volume controls for output, product interface with hearing technologies, and the usability of keys and controls by people who may have impaired vision or limited dexterity or motor control.

Video or Multimedia Products (1194.24)

Multimedia products involve more than one media and include, but are not limited to, video programs, narrated slide production, and computer generated presentations. Provisions address caption decoder circuitry (for any system with a screen larger than 13 inches) and secondary audio channels for television tuners, including tuner cards for use in computers. The standards also require captioning and audio description for certain training and informational multimedia productions developed or procured by Federal agencies. The standards also provide that viewers be able to turn captioning or video description features on or off.

Self Contained, Closed Products (1194.25)

This section covers products that generally have imbedded software but are often designed in such a way that a user cannot easily attach or install assistive technology. Examples include information kiosks, information transaction machines, copiers, printers, calculators, fax machines, and similar types of products. The standards require that access features be built into the system so users do not have to attach an assistive device to it. Other specifications address mechanisms for private listening (handset or a standard headphone jack), touchscreens, auditory output and adjustable volume controls, and location of controls in accessible reach ranges.

Desktop and Portable Computers (1194.26)

This section focuses on keyboards and other mechanically operated controls, touch screens, use of biometric form of identification, and ports and connectors.

Functional Performance Criteria (Subpart C)

The performance requirements of this section are intended for overall product evaluation and for technologies or components for which there is no specific requirement under the technical standards in Subpart B. These criteria are designed to ensure that the individual accessible components work together to create an accessible product. They cover operation, including input and control functions, operation of mechanical mechanisms, and access to visual and audible information. These provisions are structured to allow people with sensory or physical disabilities to locate, identify, and operate input, control and mechanical functions and to access the information provided, including text, static or dynamic images, icons, labels, sounds or incidental operating cues.

Information, Documentation, and Support (Subpart D)

The standards also address access to all information, documentation, and support provided to end users (e.g., Federal employees) of covered technologies. This includes user guides, installation guides for end-user installable devices, and customer support and technical support communications. Such information must be available in alternate formats upon request at no additional charge. Alternate formats or methods of communication, can include Braille, cassette recordings, large print, electronic text, Internet postings, TTY access, and captioning and audio description for video materials.

Federal Accessibility Standards for Web-based Intranet and Internet Information and Applications

(Provided by www.usability.gov)

This page contains excerpts from Electronic and Information Technology Accessibility Standards issued by the ARCHITECTURAL AND TRANSPORTATION BARRIERS COMPLIANCE BOARD.

Shown below are:

[Paragraphs from the Overview of the Standards](#)

[Subpart B — Technical Standards: Sec. 1194.22 Web-based intranet and internet information and applications.](#)

This page only contains an excerpt of the summary and standards that directly relate to Web sites. Other parts of the standard may apply to your situation. See the complete, officially posted standards at <http://www.access-board.gov/news/508-final.htm>.

Electronic and Information Technology Accessibility Standards
ARCHITECTURAL AND TRANSPORTATION BARRIERS COMPLIANCE BOARD
[Published in the Federal Register on December 21, 2000]

Summary:

Web-based Intranet and Internet Information and Applications (1194.22)

The criteria for web-based technology and information are based on access guidelines developed by the Web Accessibility Initiative of the World Wide Web Consortium. Many of these provisions ensure access for people with vision impairments who rely on various assistive products to access computer-based information, such as screen readers, which translate what's on a computer screen into automated audible output, and refreshable Braille displays. Certain conventions, such as verbal tags or identification of graphics and format devices, like frames, are necessary so that these devices can "read" them for the user in a sensible way. The standards do not prohibit the use of web site graphics or animation. Instead, the standards aim to ensure that such information is also available in an accessible format. Generally, this means use of text labels or descriptors for graphics and certain format elements. (HTML code already provides an "Alt Text" tag for graphics which can serve as a verbal descriptor for graphics). This section also addresses the usability of multimedia presentations, image maps, style sheets, scripting languages, applets and plug-ins, and electronic forms.

The standards apply to Federal web sites but not to private sector web sites (unless a site is provided under contract to a Federal agency, in which case only that web site or portion covered by the contract would have to comply). Accessible sites offer significant advantages that go beyond access. For example, those with "text-only" options provide a

faster downloading alternative and can facilitate transmission of web-based data to cell phones and personal digital assistants.

§ 1194.22 Web-based intranet and internet information and applications.

(a) A text equivalent for every non-text element shall be provided (e.g., via "alt", "longdesc", or in element content).

(b) Equivalent alternatives for any multimedia presentation shall be synchronized with the presentation.

(c) Web pages shall be designed so that all information conveyed with color is also available without color, for example from context or markup.

(d) Documents shall be organized so they are readable without requiring an associated style sheet.

(e) Redundant text links shall be provided for each active region of a server-side image map.

(f) Client-side image maps shall be provided instead of server-side image maps except where the regions cannot be defined with an available geometric shape.

(g) Row and column headers shall be identified for data tables.

(h) Markup shall be used to associate data cells and header cells for data tables that have two or more logical levels of row or column headers.

(i) Frames shall be titled with text that facilitates frame identification and navigation.

(j) Pages shall be designed to avoid causing the screen to flicker with a frequency greater than 2 Hz and lower than 55 Hz.

(k) A text-only page, with equivalent information or functionality, shall be provided to make a web site comply with the provisions of this part, when compliance cannot be accomplished in any other way. The content of the text-only page shall be updated whenever the primary page changes.

(l) When pages utilize scripting languages to display content, or to create interface elements, the information provided by the script shall be identified with functional text that can be read by assistive technology.

(m) When a web page requires that an applet, plug-in or other application be present on the client system to interpret page content, the page must provide a link to a plug-in or applet that complies with §1194.21(a) through (l).

(n) When electronic forms are designed to be completed on-line, the form shall allow people using assistive technology to access the information, field elements, and

functionality required for completion and submission of the form, including all directions and cues.

(o) A method shall be provided that permits users to skip repetitive navigation links.

(p) When a timed response is required, the user shall be alerted and given sufficient time to indicate more time is required.

Note to §1194.22:

1. The Board interprets paragraphs **(a)** through **(k)** of this section as consistent with the following priority 1 Checkpoints of the Web Content Accessibility Guidelines 1.0 (WCAG 1.0) (May 5, 1999) published by the Web Accessibility Initiative of the World Wide Web Consortium:

Section 1194.22 Paragraph	WCAG 1.0 Checkpoint
(a)	1.1
(b)	1.4
(c)	2.1
(d)	6.1
(e)	1.2
(f)	9.1
(g)	5.1
(h)	5.2
(i)	12.1
(j)	7.1
(k)	11.4

2. Paragraphs **(l)**, **(m)**, **(n)**, **(o)**, and **(p)** of this section are different from WCAG 1.0. Web pages that conform to WCAG 1.0, level A (i.e., all priority 1 checkpoints) must also meet paragraphs **(l)**, **(m)**, **(n)**, **(o)**, and **(p)** of this section to comply with this section. WCAG 1.0 is available at <http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505>.

Web Content Accessibility Guidelines (WCAG) 1.0

Introduction

The Web Content Accessibility Guidelines (WCAG), version 1.0, is a set of guidelines designed to assist web developers in designing and creating universally accessible websites. These guidelines were developed by the Web Accessibility Initiative (WAI), a subgroup of the World Wide Web Consortium (W3C). The guidelines for accessible web design are separated into various checkpoints that are assigned priority levels. The priority levels are as follows:

Priority 1:

A Web content developer **must** satisfy this checkpoint. Otherwise, one or more groups will find it impossible to access information in the document. Satisfying this checkpoint is a basic requirement for some groups to be able to use Web documents.

Priority 2:

A Web content developer **should** satisfy this checkpoint. Otherwise, one or more groups will find it difficult to access information in the document. Satisfying this checkpoint will remove significant barriers to accessing Web documents.

Priority 3:

A Web content developer **may** address this checkpoint. Otherwise, one or more groups will find it somewhat difficult to access information in the document. Satisfying this checkpoint will improve access to Web documents.

The goal of these guidelines, as outlined by the WAI, is to encourage the development and implementation of accessible on-line content for all users, not just users requiring assistive technology. According to the WAI, the guidelines are "...not to discourage content developers from using images, video, etc., but rather explain how to make multimedia content more accessible to a wide audience."

The following pages separate the WCAG 1.0 guidelines into the three priority levels established by the WAI along with the corresponding accessibility checkpoint.

WAI Priority 1 Checkpoints

In General (Priority 1)

- [1.1](#) Provide a text equivalent for every non-text element (e.g., via "alt", "longdesc", or in element content). *This includes:* images, graphical representations of text (including symbols), image map regions, animations (e.g., animated GIFs), applets and programmatic objects, ascii art, frames, scripts, images used as list bullets, spacers, graphical buttons, sounds (played with or without user interaction), stand-alone audio files, audio tracks of video, and video.
- [2.1](#) Ensure that all information conveyed with color is also available without color, for example from context or markup.
- [4.1](#) Clearly identify changes in the natural language of a document's text and any text equivalents (e.g., captions).
- [6.1](#) Organize documents so they may be read without style sheets. For example, when an HTML document is rendered without associated style sheets, it must still be possible to read the document.
- [6.2](#) Ensure that equivalents for dynamic content are updated when the dynamic content changes.
- [7.1](#) Until user agents allow users to control flickering, avoid causing the screen to flicker.
- [14.1](#) Use the clearest and simplest language appropriate for a site's content.

And if you use images and image maps (Priority 1)

- [1.2](#) Provide redundant text links for each active region of a server-side image map.
- [9.1](#) Provide client-side image maps instead of server-side image maps except where the regions cannot be defined with an available geometric shape.

And if you use tables (Priority 1)

- [5.1](#) For data tables, identify row and column headers.
- [5.2](#) For data tables that have two or more logical levels of row or column headers, use markup to associate data cells and header cells.

And if you use frames (Priority 1)

- [12.1](#) Title each frame to facilitate frame identification and navigation.

And if you use applets and scripts (Priority 1)

- [6.3](#) Ensure that pages are usable when scripts, applets, or other programmatic objects are turned off or not supported. If this is not possible, provide equivalent information on an alternative accessible page.

And if you use multimedia (Priority 1)

- [1.3](#) Until user agents can automatically read aloud the text equivalent of a visual track, provide an auditory description of the important information of the visual track of a multimedia presentation.
- [1.4](#) For any time-based multimedia presentation (e.g., a movie or animation), synchronize equivalent alternatives (e.g., captions or auditory descriptions of the visual track) with the presentation.

And if all else fails (Priority 1)

- [11.4](#) If, after best efforts, you cannot create an accessible page, provide a link to an alternative page that uses W3C technologies, is accessible, has equivalent information (or functionality), and is updated as often as the inaccessible (original) page.

Web Resources

Accessible Web Design Guidelines

Microsoft:

<http://www.microsoft.com/enable/dev/web/default.htm>

WebAIM:

<http://www.webaim.org/howto/>

EASI:

<http://www.isc.rit.edu/~easi/>

Evaluation Tools

InFocus and AskAlice

<http://www.ssbtechnologies.com/>

LIFT for DreamWeaver (and FrontPage)

<http://www.usablenet.com>

A-Prompt

<http://aprompt.snow.utoronto.ca/>

VisCheck

<http://www.vischeck.com>

Standards and Guidelines

Section 508 Standards

<http://www.section508.org>

<http://www.access-board.gov>

W3C Web Accessibility Initiative (WAI) Guidelines

<http://www.w3c.org/WAI/>

Usability.gov

<http://www.usability.gov>