Authoring Tool Accessibility Guidelines 1.0 (working draft)

W3C Working Draft 9 October 1999

This version:
    http://www.w3.org/WAI/AU/WAI-AUTOOLS-19991009
    (plain text, HTML gzip tar archive, HTML zip archive, PostScript, PDF)
Latest version:
    http://www.w3.org/WAI/AU/WAI-AUTOOLS
Previous version:
    http://www.w3.org/WAI/AU/WAI-AUTOOLS-19991009

Editors:
    Jutta Treviranus <jutta.treviranus@utoronto.ca>
    Jan Richards <jan.richards@utoronto.ca>
    Ian Jacobs <ij@w3.org>
    Charles McCathieNevile <charles@w3.org>

Copyright © 1999 W3C® (MIT, INRIA, Keio), All Rights Reserved. W3C liability, trademark, document use and software licensing rules apply.

Abstract

This document provides guidelines for Web authoring tool developers. Its purpose is two-fold: to assist developers in designing authoring tools that generate accessible Web content and to assist developers in creating an accessible authoring interface.

Accessible Web content is achieved by encouraging authoring tool users ("authors") to create accessible Web content through mechanisms such as prompts, alerts, checking and repair functions, help files and automated tools. It is equally important that all people can be the authors of Web content, rather than merely recipients. The tools used to create this information must therefore be accessible themselves. Adoption of these guidelines will contribute to the proliferation of Web pages that can be read by a broader range of readers and in authoring tools that can be used by a broader range of authors.

This document is part of a series of accessibility documents published by the W3C Web Accessibility Initiative.

Status of this document

This is a working group draft revision of the Authoring Tool Accessibility Guidelines. This draft follows the working group meeting on 7 and 8 October 1999 to resolve issues arising during last call, and reflects resolutions of that meeting. A log of changes between successive working drafts is available, as is the list of Last Call issues raised. For further information consult the minutes of Working Group Meetings. A new draft will be published around 14 October incorporating editorial amendments suggestede by reviewers during last call.

This is a draft document and may be updated, replaced or rendered obsolete by other documents at any time. It is inappropriate to use W3C Working Drafts as reference material or to cite them as other than "work in progress". This is work in progress and does not imply endorsement by either W3C or its member organizations.

The goals of the WAI AU Working Group are discussed in the WAI AU charter.

Please send comments about this document to the public mailing list: w3c-wai-au@w3.org, archived at http://lists.w3.org/Archives/Public/w3c-wai-au

A list of the current AU Working Group members is available.

Table of Contents

1. Introduction

The guidelines in this document are designed to help authoring tool developers understand, and thereby reduce, barriers to the creation of accessible Web content. In these guidelines, the term authoring tool refers to the wide range of software used for creating Web content, including:

- Editing tools specifically designed to produce Web content (e.g., WYSIWYG HTML and XML editors);
- Tools that offer the option of saving material in a Web format (e.g., word processors or desktop publishing packages);
- Tools that translate documents into Web formats (e.g., filters to translate desktop publishing formats to HTML);
- Tools that produce multimedia, especially where it is intended for use on the Web (e.g., video production and editing suites, SMIL authoring packages);
- Tools for site management or site publication, including tools which generate Websites dynamically from a database, on-the-fly conversion and Web site publishing tools;
- Tools for management of layout (e.g., CSS formatting tools).

An accessible authoring tool is accessible software that produces accessible content for the Web. Thus the goals of this document can be stated as follows: that the authoring tool be accessible to authors regardless of disability, that the authoring tool generate accessible content by default, and that the authoring tool support and encourage the author in creating accessible content. Because most of the content of the Web is created using authoring tools, they play a critical role in ensuring the accessibility of the Web. Since the Web is both a means of receiving information and communicating information, it is important that both the Web content produced and the authoring tool itself be accessible.

For detailed information about what constitutes accessible content this document relies on the Web Content Accessibility Guidelines [WAI-WEBCONTENT [p. 15]]. This document provides guidelines for designing authoring tools that generate Web content that conforms to the Web Content Accessibility Guidelines and that support and encourage authors to create content that conforms to the Web Content Accessibility Guidelines. This is achieved by taking steps such as conformance to accessible standards (i.e., HTML 4.0), accessibility checking and correcting, prompting, appropriate documentation and help. Similarly, this document does not address general accessible software design but relies on other sources. It does address accessible design considerations specific to Web authoring tools such as providing flexible editing views, navigation aids and access to display properties for authors.

A separate document, entitled Techniques for Authoring Tool Accessibility [WAI-AUTOOLS-TECH [p. 15]], provides suggestions and examples of how each checkpoint might be satisfied, It also includes references to other accessibility resources (such as platform-specific software accessibility guidelines) which give additional information on how a tool may satisfy each checkpoint. Readers are strongly encouraged to become familiar with the techniques document. Please note that while there may be many helpful suggestions there, the requirements that need to be fulfilled are the checkpoints in this document, and ways other than those suggested may be appropriate for some tools.

1.1 How the Guidelines are organized.

This document includes guidelines which are general principles of accessible design. Each guideline includes:

- The guideline number;
- The statement of the guideline;
- The rationale behind the guideline;
- A list of checkpoint definitions.

The checkpoint definitions in each guideline specify requirements for authoring tools to follow the guideline. Each checkpoint definition includes:

- The checkpoint number;
- The statement of the checkpoint;
- The priority of the checkpoint;
- In some cases informative notes, clarifying examples, or cross references to related guidelines or checkpoints;
- A link to a section of the Techniques Document ([WAI-AUTOOLS-TECH] [p. 15]) where implementations and examples of the checkpoint are discussed;

Each checkpoint is intended to be specific enough that it can be verified, while being sufficiently general to allow developers the freedom to use the most appropriate strategies to meet the checkpoint.

The Techniques provided in the techniques document are suggestions for how implementation might be done, or where further information can be found. They are informative only, and other strategies may be used to meet the checkpoint as well as, or in place of, those discussed.

1.2 Checkpoint priorities

Each checkpoint has a priority level. The priority level reflects the impact of the checkpoint in meeting the goals of this document. These goals are:

- That the authoring tool be accessible
- That the authoring tool generate accessible content by default
- That the authoring tool encourage the creation of accessible content

The three priority levels are assigned as follows:

[Priority 1]
: If the checkpoint is essential to meeting those goals

[Priority 2]
: If the checkpoint is important to meeting those goals

[Priority 3]
: If the checkpoint is beneficial to meeting those goals

[Relative Priority]

Some checkpoints talk about production, generation, checking etc of various content that have different priorities in WCAG. The priority for these checkpoints in ATAG varies according to the priority of the checkpoints in WCAG.

Relative Priority is used for ATAG checkpoints which relate to different types of content, to ensure that the priority of each feature matches the priority given to the feature in WCAG, as follows:

- It is priority 1 to implement the checkpoint for content features which are a priority 1 requirement in WCAG.
- It is priority 2 to implement the checkpoint for content features which are a priority 2 requirement in WCAG.
- It is priority 3 to implement the checkpoint for content features which are a priority 3 requirement in WCAG.

For example, checking for accessibility errors ([##Cnotify-on-schedule]) has relative priority. This means that it is priority 1 for a tool to check for accessibility errors that are [Web-Content-Priority-1] [p. 15] , but priority 2 to check for accessibility errors that are [Web-Content-Priority-2] [p. 15] .

1.3 Conformance to these Guidelines

This section defines three levels of conformance to this document:

- Conformance Level "A": all Priority 1 checkpoints are satisfied;
- Conformance Level "Double-A": all Priority 1 and 2 checkpoints are satisfied;
- Conformance Level "Triple-A": all Priority 1, 2, and 3 checkpoints are satisfied;

Note. Conformance levels are spelled out in text (e.g., "Double-A" rather than "AA") so they may be understood when rendered to speech.

Claims of conformance to this document must use one of the following two forms.

Form 1: Specify:

- The guidelines' title: "Authoring Tool Accessibility Guidelines 1.0 (working draft)"
- The guidelines' URI: http://www.w3.org/WAI/AU/WAI-AUTOOLS-19991009
- The conformance level satisfied: "A", "Double-A", or "Triple-A".
- The product covered by the claim (e.g., tool name and version number, upgrades or plug-ins required).

Example of Form 1: "MyAuthoringTool version 2.3 conforms to W3C's "Authoring Tool Accessibility Guidelines 1.0 (working draft)", available at http://www.w3.org/WAI/AU/WAI-AUTOOLS-19991009, level Double-A."

Form 2: Include, on each statement of conformance, one of three icons provided by W3C and link the icon to the appropriate W3C explanation of the claim.

[Editors' note: In the event this document becomes a Recommendation, by that date WAI will provide a set of three icons, for "A", "Double-A", or "Triple-A" conformance levels of "Authoring Tool Accessibility Guidelines 1.0 (working draft)", together with a stable URI to the W3C Web site for linking the icons to the W3C explanation of conformance claims.]

2. Guidelines

Guideline 1. Support accessible authoring practices

Methods for ensuring accessible markup vary with different markup languages. If markup is automatically generated, many authors will be unaware of the accessibility status of the final product unless they expend extra effort to make appropriate corrections by hand. Since many authors are unfamiliar with accessibility, the onus is on the authoring tool to generate accessible markup, and where appropriate, to guide the author in producing accessible content.

Many applications feature the ability to convert documents from other formats (e.g., Rich Text Format) into a markup format, such as HTML. Markup changes may also be made to facilitate efficient editing and manipulation. These processes are usually hidden from the user's view and may create inaccessible markup or cause inaccessible markup to be produced.

Checkpoints:

1.1 Ensure that the author can produce accessible content in the markup language(s) supported by the tool. [Priority 1]
[#C+Rgenerate-access-markup] Ensure that the tool generates markup that conforms to the W3C's Web Content Accessibility Guidelines [WAI-WEBCONTENT] [p. 15] .
[#C+Ruse-accessible-templates] Ensure that templates provided by the tool conform to the Web Content Accessibility Guidelines [WAI-WEBCONTENT] [p. 15] .
1.2 Ensure that the tool preserves all accessibility information during authoring, transformations and conversions. [Priority 1]

Guideline 2. Generate standard markup

Conformance with standards promotes interoperability and accessibility. Where applicable use W3C Recommendations, which have been reviewed to ensure accessibility and interoperability. If there are no applicable W3C Recommendations, use a published standard that enables accessibility.

Checkpoints:

2.1 Use the latest versions of W3C Recommendations when they are available and appropriate for a task. [Priority 2]

These specifications have undergone review specifically to ensure that they do not compromise, and where possible they enhance, accessibility.

2.2 Ensure that the tool generates valid markup. [Priority 1]

This is necessary for user agents to be able to transform Web content to a presentation appropriate to a particular user's needs.

2.3 If markup generated by the tool differs from W3C specifications, inform the author. [Priority 3]

This allows the author to choose to conform.

Guideline 3. Support the creation of accessible content

Generating equivalent information, such as textual alternatives for images and audio descriptions of video, can be one of the most challenging aspects of Web design. Along with the necessity for structural information it is a cornerstone of accessible design, allowing information to be presented in a way most appropriate for the needs of the user without constraining the creativity of the author.

Automating the mechanics of this process, by prompting authors to include the relevant information at appropriate times, can greatly ease the burden for authors. Where such information can be mechanically determined (e.g., the function of icons in an automatically-generated navigation bar, or expansion of acronyms from a dictionary) and offered as a choice for the author the tool will assist the author, at the same time as it reinforces the need for such information and the author's role in ensuring that it is used appropriately in each instance.

Checkpoints:

[#C+Rprovide-missing-alt] Assist the author in providing alternative information (e.g., captions, long descriptions of graphics).

[#C+Rhelp-provide-structure] Help the author create structured content and separate information from its presentation.

[#C+Rinclude-pro-descs] Ensure that prepackaged content conforms to Web Content Accessibility Guidelines [WAI-WEBCONTENT] [p. 15] .

For example include synchronised text and audio equivalents with movies. (Refer also to [##C-no-default-alt]).

3.1 Do not insert automatically generated or place-holder equivalent alternatives. [Priority 1]

For example, in an automatically generated navigation bar, "search" may be appropriate alternative information for a button linked to a search function, but the filename of an image should not be inserted as a default.

Note that human-authored content may be available for an object whose function is known with certainty (See also 3.2, [##Cinclude-pro-descs]

3.2 Provide a mechanism to manage alternative information for multimedia objects, that retains and offers for editing pre-written or previously linked alternative information. [Priority 3]

Guideline 4. Provide methods of checking and correcting inaccessible content

Many authoring tools allow authors to create documents with little or no knowledge about the underlying markup. To ensure accessibility, authoring tools must be designed so that they may automatically identify inaccessible markup, and enable its correction even when the markup itself is hidden from the author.

In supporting the creation of accessible Web content, authoring tools should take into account the differing authoring styles of their users. In general, authors will prefer to be able to configure their tools to support their working style. Tools that allow such configuration can help authors to feel that accessible authoring is a natural practice (see also the previous guideline) rather than an intrusion on their normal work pattern. For example some users may prefer to be alerted to problems when they occur, whereas others may prefer to perform a check after the document is completed. This is analogous to programming environments that allow users to decide whether to check for correct code during editing or at compile time.

Note that many assistive technologies used with browsers and multimedia players are only able to provide access to Web documents that use valid mark-up. Therefore validation of mark-up is an essential aspect of authoring tool accessibility.

Checkpoints:

[#C+Rnotify-on-schedule] Check for and alert the author to accessibility problems.
[#C+Rdont-require-knowledge] Assist authors in correcting accessibility problems.
4.1 Allow the author to preserve markup not recognized by the tool. [Priority 2]
Notes:
1. The author may have included or imported markup that is not recognized by the tool, but which enhances accessibility.

4.2 Provide the author with a summary of the document accessibility status. [Priority 3]
4.3 Allow the author to transform presentation markup that is misused to convey structure into structural markup, and to transform presentation markup that is stylistic into style sheet markup. [Priority 3]

Guideline 5. Integrate accessibility solutions into the overall "look and feel"

When a new feature is added to an existing software tool without proper integration, the result is often an obvious discontinuity. Differing color schemes, fonts, interaction styles and even application stability can be factors affecting user acceptance of the new feature.

Checkpoints:

5.1 Ensure that functionalities related to accessible Authoring practices are integrated into the tool. [Priority 2]
5.2 Ensure that the [Web-Content-Priority-1] [p. 15] accessible authoring practices are among the most obvious and easily initiated by the author. [Priority 2]

Guideline 6. Promote accessibility in help and documentation

The issues surrounding Web accessibility are often unknown to Web authors. Help and documentation should explain accessibility problems and solutions, with examples.

Checkpoints:

6.1 Document all features that promote the production of accessible content. [Priority 1]
6.2 Ensure that creating accessible content is a naturally integrated part of the documentation, including examples. [Priority 2]
6.3 In a dedicated section, document all features of the tool that promote the production of accessible content. [Priority 3]

Guideline 7. Ensure that the Authoring Tool is Accessible to Authors with Disabilities

The authoring tool is a software program with standard user interface elements and as such should follow relevant user interface accessibility guidelines. In addition to applicable general interface accessibility guidelines there are interface design considerations that are specific to Web authoring tools.

One such consideration is that the author may need a different presentation to edit the Web content than the one they wish ultimately to be displayed. This implies display preferences that do not manifest themselves in the ultimate markup or style declarations.

Another consideration relates to the process of navigating and manipulating the document while authoring. Authoring Web content requires editing a potentially large and complex document. In order to edit a document the author must be able to locate and select specific elements, efficiently traverse the document, and quickly find and mark insertion points. Authors who use screen readers, refreshable braille displays, or screen magnifiers can make limited use (if at all) of visual artifacts that communicate the structure of the document and act as sign posts when traversing the document. Authors who use keyboard and mouse alternatives must make tiring repetitions of movement commands to navigate the document. There are strategies that make it easier to navigate and manipulate a marked-up document. Using the structure of a Web document, the author can be given a view of the document which allows the author to both get a good sense of the overall document and to navigate that document more easily.

Checkpoints:

7.1 Use all applicable operating system and accessibility standards and conventions (Priority 1 for standards and conventions which are essential to accessibility, Priority 2 for those that are important to accessibility, Priority 3 for those that are beneficial to accessibility). [Priority 1]
7.2 Allow the author to change the editing view [p. 13] without affecting the document markup. [Priority 1]

> This allows the author to edit the document according to their personal requirements, without changing the way the document looks or is rendered when published.

7.3 Allow the author to edit all properties of each element and object in an accessible fashion. [Priority 1]
7.4 Ensure the editing view allows navigation via the structure of the document in an accessible fashion. [Priority 1]
7.5 Enable editing of the structure of the document in an accessible fashion. [Priority 2]
7.6 Allow the author to search within editing views. [Priority 2]

3. Terms and Definitions

[Editors' note: This section will be updated to reflect comments received during last call. In particualr, it will be in alphabetical order]

User Configurable Schedule
: A user configurable schedule allows the user to determine the type of prompts and alerts that are used, including when they are presented.

Prompts
: Prompts are requests for user input, either information or a decision. Prompts require author response.

Alerts
: Alerts notify the author of something, or mark something for the author's attention. They may or may not require author response.

Authoring Tool
: As used in this document, an *Authoring Tool* is any software that is used to generate content for publishing on the Web. See also section 1.3 Scope of these guidelines.

Transformation
: A process whereby one object is changed, according to a discrete set of rules, into another, equivalent, object. This includes any application or application feature that allows content that is marked up in a particular markup language to be transformed into another markup language, such as software that allows the author to change the DTD defined for the original document to another DTD. .

Document
: A *document* is a series of elements that are defined by a language (e.g., HTML 4.0 or an XML application).

Element
: An element is any identifiable object within a document, for example a character, word, image, paragraph or spreadsheet cell. In HTML and XML an element refers to a pair of tags and their content, or an "empty" tag - one that has no closing tag or content.

Property
: A property is a piece of information about an element, for example structural information (e.g., it is item number 7 in a list, or plain text) or presentation information (e.g., that it is marked as bold, its font size is 14). In XML and HTML properties of an element include the name of the element (e.g., IMG or DL), the values of its attributes, and information associated by means of a style sheet. In a database, properties of a particular element may include values of the entry, and acceptable data types for that element.

Attributes
: in XML and HTML, an element may have any number of attributes. In the following example, the attributes of the beverage element are flavor, which has the value "lots", and colour, which has the value "red": <beverage flavor="lots" colour="red">my favorite</beverage> Some attributes are integral to document accessibility (e.g., the "alt", "title", and "longdesc" attributes in HTML

Rendered Content
: The *rendered content* is that which an element actually causes to be rendered by the user agent. This may differ from the element's structural content. For example, some elements cause external data to be rendered (e.g., the IMG element in HTML), and in some cases, browsers may render the value of an attribute (e.g., "alt", "title") in place of the element's content.

Accessible, Accessibility
: Within these guidelines, Accessible and Accessibility are used in the sense of being accessible to people regardless of disability.

To understand the accessibility issues relevant to authoring tool design, consider that many users may be creating documents in contexts very different from your own:

- They may not be able to see, hear, move, or may not be able to process some types of information easily or at all;
- They may have difficulty reading or comprehending text;
- They may not have or be able to use a keyboard or mouse;
- They may have a text-only display, or a small screen.

In addition, accessible design will benefit many people who do not have a physical disability but with similar needs. For example they may be working in a noisy environment and unable to hear, or need to use their eyes for another task, and be unable to view a screen. They may be using a small mobile device, with a small screen, no keyboard and no mouse.

Accessibility Information
: Accessibility information is content, including information and markup, which is used to improve the accessibility of a document.

Accessibility Solution, Accessible Authoring Practice
: These terms refer to Authoring practices that improve the accessibility of content generated by the tool.

Alternative Presentations and Alternative Information
: Certain types of content may not be accessible to all users (e.g., images or audio presentations), so alternative representations are used, such as transcripts for audio, or short functionally equivalent text (e.g., "site map link") and/or descriptive text equivalents (e.g., "Graph 2.5 shows that the population has doubled approximately every ten years for the last fifty years, increasing from about 10 million to 330 million in that time"). An object may have several alternative representations, for example a video, captions of the audio, audio description of the video, a series of still images, and textual representations of each of these.

Views
: An authoring tool may offer several *views* of the same document. For instance, one view may show raw markup, a second may show a structured tree view, a third may show markup with rendered objects while a final view shows an example of how the document may appear if it were to be rendered by a particular browser.

Editing view
: What is displayed by the authoring tool to the author during the editing process.

Markup Language
: The term *markup language* is used in this document to refer to the encoding language of a document, such as HTML, SVG, or MathML.

4. Acknowledgments

Many thanks to the following people who have contributed through review and comment: Jim Allan, Denis Anson, Kynn Bartlett, Harvey Bingham, Judy Brewer, Carl Brown, Dick Brown, Kelly Ford, Wendy Chisholm, Rob Cumming, Daniel Dardailler, Mark Day, BK Delong, Jamie Fox, Edna French, Sylvain Galineau, Al Gilman, Eric Hansen, Phill Jenkins, Len Kasday, Brian Kelly, Marja-Riitta Koivunen, Jaap van Lelieveld, William Loughborough, Karen McCall, Charles Oppermann, Dave Pawson, Dave Poehlman, Bruce Roberts, Chris Ridpath, Gregory Rosmaita, Jim Thatcher, Irène Vatton, Gregg Vanderheiden, Pawan Vora, Jason White, and Lauren Wood.

If you have contributed to the AU guidelines and your name does not appear please contact the editors to add your name to the list.

5. References

For the latest version of any W3C specification please consult the list of W3C Technical Reports.

[HTML40]
: "HTML 4.0 Recommendation", D. Raggett, A. Le Hors, and I. Jacobs, eds., 17 December 1997, revised 24 April 1998. This HTML 4.0 Recommendation is http://www.w3.org/TR/1998/REC-html40-19980424. The latest version of HTML 4.0 is available at http://www.w3.org/TR/REC-html40.

[WAI-AUTOOLS-TECH]
: "Authoring Tool Accessibility Techniques (Working Draft)", J. Treviranus, J. Richards, I. Jacobs, and C. McCathieNevile eds. The latest draft of the Techniques for Authoring Tool Accessibility Guidelines is available at http://www.w3.org/WAI/AU/WAI-AUTOOLS.

[WAI-USERAGENT]
: "User Agent Accessibility Guidelines", J. Gunderson and I. Jacobs, eds. The latest version of the User Agent Accessibility Guidelines is available at http://www.w3.org/WAI/UA/WAI-USERAGENT.

[WAI-WEBCONTENT]
: "Web Content Accessibility Guidelines 1.0", W. Chisholm, G. Vanderheiden, and I. Jacobs, eds., 5 May 1999. This Recommendation is http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505. The latest version of the Web Content Accessibility Guidelines 1.0" is available at http://www.w3.org/TR/WAI-WEBCONTENT/.

[WAI-WEBCONTENT-TECHS]
: "Techniques for Web Content Accessibility Guidelines 1.0", W. Chisholm, G. Vanderheiden, and I. Jacobs, eds. The latest version of Techniques for Web Content Accessibility Guidelines 1.0 is available at http://www.w3.org/TR/WAI-WEBCONTENT-TECHS/.

[Web-Content-Priority]
: Priorities defined by [WAI-WEBCONTENT] [p. 15] .