

World Wide Web Consortiums Web content accessibility Guidelines

Required for Priority 1

Guideline 1. Provide equivalent alternatives to auditory and visual content.

Provide content that, when presented to the user, conveys essentially the same function or purpose as auditory or visual content.

Although some people cannot use images, movies, sounds, applets, etc. directly, they may still use pages that include *equivalent* information to the visual or auditory content. The equivalent information must serve the same purpose as the visual or auditory content. Thus, a text equivalent for an image of an upward arrow that links to a table of contents could be "Go to table of contents". In some cases, an equivalent should also describe the appearance of visual content (e.g., for complex charts, billboards, or diagrams) or the sound of auditory content (e.g., for audio samples used in education).

This guideline emphasizes the importance of providing *text equivalents* of non-text content (images, pre-recorded audio, video). The power of text equivalents lies in their capacity to be rendered in ways that are accessible to people from various disability groups using a variety of technologies. Text can be readily output to speech synthesizers and *braille displays*, and can be presented visually (in a variety of sizes) on computer displays and paper. Synthesized speech is critical for individuals who are blind and for many people with the reading difficulties that often accompany cognitive disabilities, learning disabilities, and deafness. Braille is essential for individuals who are both deaf and blind, as well as many individuals whose only sensory disability is blindness. Text displayed visually benefits users who are deaf as well as the majority of Web users.

Providing non-text equivalents (e.g., pictures, videos, and pre-recorded audio) of text is also beneficial to some users, especially nonreaders or people who have difficulty reading. In movies or visual presentations, visual action such as body language or other visual cues may not be accompanied by enough audio information to convey the same information. Unless verbal descriptions of this visual information are provided, people who cannot see (or look at) the visual content will not be able to perceive it.

Checkpoints:

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.

[Priority 1]

For example, in HTML:

- Use "alt" for the IMG, INPUT, and APPLET elements, or provide a text equivalent in the content of the OBJECT and APPLET elements.
- For complex content (e.g., a chart) where the "alt" text does not provide a complete text equivalent, provide an additional description using, for example, "longdesc" with IMG or FRAME, a link inside an OBJECT element, or a [description link](#).
- For image maps, either use the "alt" attribute with AREA, or use the MAP element with A elements (and other text) as content.

Refer also to [checkpoint 9.1](#) and [checkpoint 13.10](#).

Quicktest! A good test to determine if a text equivalent is useful is to imagine reading the document aloud over the telephone. What would you say upon encountering this image to make the page comprehensible to the listener?

1.2 Provide redundant text links for each active region of a server-side image map. [Priority 1]

Refer also to [checkpoint 1.5](#) and [checkpoint 9.1](#).

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.

[Priority 1]

Synchronize the [auditory description](#) with the audio track as per [checkpoint 1.4](#). Refer to [checkpoint 1.1](#) for information about textual equivalents for visual information.

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.

[Priority 1]

Guideline 2. Don't rely on color alone.

Ensure that text and graphics are understandable when viewed without color.

If color alone is used to convey information, people who cannot differentiate between certain colors and users with devices that have non-color or non-visual displays will not receive the information. When foreground and background colors are too close to the same hue, they may not provide sufficient contrast when viewed using monochrome displays or by people with different types of color deficits.

Checkpoints:

2.1 Ensure that all information conveyed with color is also available without color, for example from context or markup. **[Priority 1]**

Quicktest! To determine if content is structural or presentational, create an outline of your document. Each point in the hierarchy denotes a structural change. Use structural markup to mark these changes and presentational markup to make them more apparent visually and aurally. Notice that horizontal rules will not appear in this outline and therefore are not structural, but presentational. **Note.** This quicktest addresses chapter, section, and paragraph structure. To determine structure within phrases, look for abbreviations, changes in natural language, definitions, and list items.

Quicktest! To test whether your document still works without colors, examine it with a monochrome monitor or browser colors turned off. Also, try setting up a color scheme in your browser that only uses black, white, and the four browser-safe greys and see how your page holds up.

Quicktest! To test whether color contrast is sufficient to be read by people with color deficiencies or by those with low resolution monitors, print pages on a black and white printer (with backgrounds and colors appearing in grayscale). Also try taking the printout and copying it for two or three generations to see how it degrades. This will show you where you need to add redundant cues (example: hyperlinks are usually underlined on Web pages), or whether the cues are too small or indistinct to hold up well.

Guideline 4. Clarify natural language usage

Use markup that facilitates pronunciation or interpretation of abbreviated or foreign text.

When content developers mark up natural language changes in a document, speech synthesizers and braille devices can automatically switch to the new language, making the document more accessible to multilingual users. Content developers should identify the predominant natural language of a

document's content (through markup or HTTP headers). Content developers should also provide expansions of abbreviations and acronyms.

In addition to helping assistive technologies, natural language markup allows search engines to find key words and identify documents in a desired language. Natural language markup also improves readability of the Web for all people, including those with learning disabilities, cognitive disabilities, or people who are deaf.

When abbreviations and natural language changes are not identified, they may be indecipherable when machine-spoken or brailled.

Checkpoints:

- 4.1** Clearly identify changes in the natural language of a document's text and any *text equivalents* (e.g., captions). **[Priority 1]**
For example, in HTML use the "lang" attribute. In XML, use "xml:lang".

Guideline 5. Create tables that transform gracefully.



Ensure that tables have necessary markup to be transformed by accessible browsers and other user agents.

Tables should be used to mark up truly *tabular information* ("data tables"). Content developers should avoid using them to lay out pages ("layout tables"). Tables for any use also present special problems to users of *screen readers* (refer to [checkpoint 10.3](#)).

Some *user agents* allow users to navigate among table cells and access header and other table cell information. Unless marked-up properly, these tables will not provide user agents with the appropriate information. ([Refer also to guideline 3.](#))

The following checkpoints will directly benefit people who access a table through auditory means (e.g., a screen reader or an automobile-based personal computer) or who view only a portion of the page at a time (e.g., users with blindness or low vision using speech output or a *braille display*, or other users of devices with small displays, etc.).

Checkpoints:

- 5.1** For data tables, identify row and column headers. **[Priority 1]**
For example, in HTML, use TD to identify data cells and TH to identify 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. **[Priority 1]**

For example, in HTML, use THEAD, TFOOT, and TBODY to group rows, COL and COLGROUP to group columns, and the "axis", "scope", and "headers" attributes, to describe more complex relationships among data.

Guideline 6. Ensure that pages featuring new technologies transform gracefully.

Ensure that pages are accessible even when newer technologies are not supported or are turned off.

Although content developers are encouraged to use new technologies that solve problems raised by existing technologies, they should know how to make their pages still work with older browsers and people who choose to turn off features.

Checkpoints:

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. **[Priority 1]**

When content is organized logically, it will be rendered in a meaningful order when style sheets are turned off or not supported.

6.2 Ensure that equivalents for dynamic content are updated when the dynamic content changes. **[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. **[Priority 1]**

For example, ensure that links that trigger scripts work when scripts are turned off or not supported (e.g., do not use "javascript:" as the link target). If it is not possible to make the page usable without scripts, provide a text equivalent with the NOSCRIPT element, or use a server-side script instead of a client-side script, or provide an alternative accessible page as per [checkpoint 11.4](#). [Refer also to guideline 1](#).

Guideline 7. Ensure user control of time-sensitive content changes.

Ensure that moving, blinking, scrolling, or auto-updating objects or pages may be paused or stopped.

Some people with cognitive or visual disabilities are unable to read moving text quickly enough or at all. Movement can also cause such a distraction that the rest of the page becomes unreadable for people with cognitive disabilities. [Screen readers](#) are unable to read moving text. People with physical disabilities might not be able to move quickly or accurately enough to interact with moving objects.

Note. All of the following checkpoints involve some content developer responsibility [until user agents](#) provide adequate feature control mechanisms.

Checkpoints:

7.1 [Until user agents](#) allow users to control flickering, avoid causing the screen to flicker. **[Priority 1]**

Note. People with photosensitive epilepsy can have seizures triggered by flickering or flashing in the 4 to 59 flashes per second (Hertz) range with a peak sensitivity at 20 flashes per second as well as quick changes from dark to light (like strobe lights).

Guideline 9. Design for device-independence.

Use features that enable activation of page elements via a variety of input devices.

[Device-independent](#) access means that the user may interact with the user agent or document with a preferred input (or output) device -- mouse, keyboard, voice, head wand, or other. If, for example, a form control can only be activated with a mouse or other pointing device, someone who is using the page without sight, with voice input, or with a keyboard or who is using some other non-pointing input device will not be able to use the form.

Note. Providing text equivalents for image maps or images used as links makes it possible for users to interact with them without a pointing device. [Refer also to guideline 1.](#)

Generally, pages that allow keyboard interaction are also accessible through speech input or a command line interface.

Checkpoints:

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. **[Priority 1]**

Refer also to [checkpoint 1.1](#), [checkpoint 1.2](#), and [checkpoint 1.5](#).

Guideline 11. Use W3C technologies and guidelines.

Use W3C technologies (according to specification) and follow accessibility guidelines. Where it is not possible to use a W3C technology, or doing so results in material that does not transform gracefully, provide an alternative version of the content that is accessible.

The current guidelines recommend W3C technologies (e.g., HTML, CSS, etc.) for several reasons:

- W3C technologies include "built-in" accessibility features.
- W3C specifications undergo early review to ensure that accessibility issues are considered during the design phase.
- W3C specifications are developed in an open, industry consensus process.

Many non-W3C formats (e.g., PDF, Shockwave, etc.) require viewing with either plug-ins or stand-alone applications. Often, these formats cannot be viewed or navigated with standard [user agents](#) (including [assistive technologies](#)). Avoiding non-W3C and non-standard features (proprietary elements, attributes, properties, and extensions) will tend to make pages more accessible to more people using a wider variety of hardware and software. When inaccessible technologies (proprietary or not) must be used, equivalent accessible pages must be provided.

Even when W3C technologies are used, they must be used in accordance with accessibility guidelines. When using new technologies, ensure that they transform gracefully ([Refer also to guideline 6](#)).

Note. Converting documents (from PDF, PostScript, RTF, etc.) to W3C markup languages (HTML, XML) does not always create an accessible document. Therefore, validate each page for accessibility and usability after the conversion process (refer to the [section on validation](#)). If a page does not readily convert, either revise the page until its original representation converts appropriately or provide an HTML or plain text version.

Checkpoints:

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. **[Priority 1]**

Note. Content developers should only resort to alternative pages when other solutions fail because alternative pages are generally updated less often than "primary" pages. An out-of-date page may be as frustrating as one that is inaccessible since, in both cases, the information presented on the original page is unavailable. Automatically generating alternative pages may lead to more frequent updates, but content developers must still be careful to ensure that generated pages always make sense, and that users are able to navigate a site by following links on primary pages, alternative pages, or both. Before resorting to an alternative page, reconsider the design of the original page; making it accessible is likely to improve it for all users.

Guideline 12. Provide context and orientation information.

Provide context and orientation information to help users understand complex pages or elements.

Grouping elements and providing contextual information about the relationships between elements can be useful for all users. Complex relationships between parts of a page may be difficult for people with cognitive disabilities and people with visual disabilities to interpret.

Checkpoints:

12.1 Title each frame to facilitate frame identification and navigation.

[Priority 1]

For example, in HTML use the "title" attribute on FRAME elements.

Guideline 14. Ensure that documents are clear and simple.



Ensure that documents are clear and simple so they may be more easily understood.

Consistent page layout, recognizable graphics, and easy to understand language benefit all users. In particular, they help people with cognitive disabilities or who have difficulty reading. (However, ensure that images have text equivalents for people who are blind, have low vision, or for any user who cannot or has chosen not to view graphics. [Refer also to guideline 1.](#))

Using clear and simple language promotes effective communication. Access to written information can be difficult for people who have cognitive or learning disabilities. Using clear and simple language also benefits people whose first language differs from your own, including those people who communicate primarily in sign language.

Checkpoints:

14.1 Use the clearest and simplest language appropriate for a site's content. [Priority 1]

- [14.1](#) Use the clearest and simplest language appropriate for a site's content. [Priority 1]
- [13.8](#) Place distinguishing information at the beginning of headings, paragraphs, lists, etc. [Priority 3]
- [14.2](#) Supplement text with graphic or auditory presentations where they will facilitate comprehension of the page. [Priority 3]

The following sections discuss techniques for helping comprehension of a page or site.

Writing style

The following writing style suggestions should help make the content of your site easier to read for everyone, especially people with reading and/or cognitive disabilities. Several guides discuss these and other writing style issues in more detail.

1. Strive for clear and accurate headings and link descriptions. This includes using link phrases that are terse and that make sense when read out of context or as part of a series of links (Some users browse by jumping from link to link and listening only to link text.) Use informative headings so that users can scan a page quickly for information rather than reading it in detail.
2. State the topic of the sentence or paragraph at the beginning of the sentence or paragraph (this is called "front-loading"). This will help both people who are skimming visually, but also people who use speech synthesizers. "Skimming" with speech currently means that the user jumps from heading to heading, or paragraph to paragraph and listens to just enough words to determine whether the current chunk of information (heading, paragraph, link, etc.) interests them. If the main idea of the paragraph is in the middle or at the end, speech users may have to listen to most of the document before finding what they want. Depending on what the user is looking for and how much they know about the topic, search features may also help users locate content more quickly.
3. Limit each paragraph to one main idea.
4. Avoid slang, jargon, and specialized meanings of familiar words, unless defined within your document.

5. Favor words that are commonly used. For example, use "begin" rather than "commence" or use "try" rather than "endeavor."
6. Use active rather than passive verbs.
7. Avoid complex sentence structures.

To help determine whether your document is easy to read, consider using the Gunning-Fog reading measure (described in [\[SPOOL\]](#) with examples and the algorithm online at [\[TECHHEAD\]](#)). This algorithm generally produces a lower score when content is easier to read. As example results, the Bible, Shakespeare, Mark Twain, and TV Guide all have Fog indexes of about 6. Time, Newsweek, and the Wall St. Journal an average Fog index of about 11.