ASSURANCE OF DISCONTINUANCE

Pursuant to the provisions of Section 63(12) of the New York Executive Law, Eliot Spitzer, Attorney General of the State of New York (the “Attorney General”), has made an inquiry into whether the PRICELINE.COM web site of priceline.com Incorporated (“Priceline”) is accessible to the assistive technology used by the blind and visually impaired (the “Inquiry”). Based upon the results of the Inquiry, the Attorney General finds as follows:

DEFINITIONS

1. The terms used in this Assurance to refer to the applicable technology shall be given the same meaning as used in the Web Content Accessibility Guidelines 1.0 (“W3C Guidelines”), such as in the Glossary provided at http://www.w3c.org/TR/WCAG10/ and elsewhere on the W3C.ORG web site, and, as to any terms not found in the W3C Guidelines or on the W3C.ORG web site, as they are used commonly in the technology industry. A copy of the W3C Guidelines is annexed hereto as Exhibit A.

FINDINGS OF THE ATTORNEY GENERAL

2. Priceline is an Internet-based company that offers travel services and products through the PRICELINE.COM web site.

4. Through the PRICELINE.COM web site, Priceline offers “name your own price” travel services and products, such as airline tickets, hotel rooms, rental cars and packaged vacations and cruises. Priceline also markets, through the PRICELINE.COM web site, financial services, such as home mortgages, refinancing and home equity loans through an independent licensee.

5. Many blind and visually impaired persons use assistive technology, such as screen reader software, to operate a computer, including surfing the Internet. Assistive technology, such as Freedom Scientific’s JAWS 4.5 for Windows and Microsoft’s Window Eyes, converts text into speech. When a screen reader user directs his/her computer to a web page, for example, the screen reader software reads each section of the web page aloud, generally from top to bottom and left to right, as if it were reading a book.

6. The computer code of a web page dictates what the screen reader software reads aloud. Therefore, in order to be accessible, a web site must utilize computer code that is comprehensible to a screen reader software user. For example, all graphics must have comprehensible labels, each link on a web page must be meaningfully labeled, tables must have appropriately placed row and column headers (so that each heading is read aloud with each table entry, thereby giving meaning to the table entries), and each edit field (i.e., the box where the Internet user must input information, such as name and address) must be labeled to indicate which information is requested. Absent these accessible codes, blind and visually impaired
persons cannot navigate a web site.

7. Upon investigation, the Attorney General determined that the PRICELINE.COM web site had several elements that made it inaccessible to screen reader software. For example, certain links and images were not properly labeled, certain tables and forms were not labeled and pop-up windows did not provide a "close window" link at the top of the window. Accordingly, critical functions of the PRICELINE.COM web site, such as making a hotel reservation, were difficult, if not impossible, for any blind or visually impaired individual using screen reader software.

8. It is estimated that there are more than 450,000 blind and visually impaired New York residents.

9. The Americans With Disabilities Act of 1990, 42 U.S.C. §§ 12131-12134 ("ADA"), prohibits any "public accommodation" from denying disabled individuals, on the basis of their disability, the opportunity to participate in or benefit from the goods or services of the "public accommodation." 42 U.S.C. § 12182(b)(1)(A)(i). Under the ADA, private entities that own, lease (or lease to) or operate a place of public accommodation, such as travel services, other service establishments and sales establishments, may not discriminate against the disabled in the provision of goods, services, facilities, privileges, advantages or accommodations. 42 U.S.C. § 12182(a). New York law provides similar protections to blind and visually impaired persons. See, e.g., Section 296(2)(a) of the Executive Law and Section 40-c of the Civil Rights Law.

10. The Attorney General alleges that, by failing to ensure that the PRICELINE.COM web site is accessible to the assistive technology used by the blind and visually impaired, Priceline has not complied with the ADA, New York Executive Law or New York Civil Rights
Law.

11. **IT NOW APPEARS** that Priceline is willing to enter into this Assurance of Discontinuance ("Assurance") without admitting the Attorney General’s allegations, and that the Attorney General is willing to accept the terms of this Assurance of Discontinuance pursuant to Executive Law § 63(15) in lieu of commencing suit. Therefore, Priceline and the Attorney General (collectively referred to hereafter as the "Parties") agree as follows:

**AGREEMENT**

12. This Assurance shall be binding on and apply to Priceline whether acting now or hereafter in its own capacity or through any of its officers, directors, servants, agents, employees, assignees, or any individual, subsidiary, division, or other entity, as well as any successors in interest.

13. This Assurance applies to the entire PRICELINE.COM web site, including all directories, subdirectories and web pages, with a limited exception for pages that are displayed directly by a third-party using the third-party’s software, such that the page is displayed through a server that is owned, operated and programmed by the third-party. With respect to those pages falling under this limited exception, Priceline shall: (i) provide to the Attorney General within thirty (30) days following execution by both parties of this Assurance an affidavit listing all of the paths along the PRICELINE.COM web site that contain pages falling under the exception and certifying that these paths, indeed, use web pages that are displayed directly by a third-party using the third-party’s software, such that the page is displayed through a server that is owned, operated and programmed by the third-party; (ii) update this affidavit on July 1, 2004 and January 17, 2005; and (iii) request that each of the providers of these pages comply with the
various checkpoints contained in Paragraph 18 of this Assurance, maintaining for inspection by the Attorney General all related correspondence. If Priceline’s Auditor, as defined in Paragraph 21 herein, confirms that a web page falling under the limited exception is in compliance with the various checkpoints contained in Paragraph 18 of this Assurance, then Priceline need not include reference to that particular web page in the affidavits referred to herein, but must include reference to the applicable path if there are other pages along the path that qualify for the exception described herein.

14. This Assurance shall also apply to any web site through which Priceline offers the same products and services that are currently available at PRICELINE.COM. For example, if Priceline moves the products and services currently available through its PRICELINE.COM web site to PRICELINE.TRAVEL, then this Assurance and each of its terms would apply to PRICELINE.TRAVEL.

15. Within ten (10) days of the execution date of this Assurance, Priceline shall pay to the New York Department of Law by certified check the sum of $37,500 as and for the costs of the Inquiry.

16. Nothing in this Assurance shall be construed to alter or enhance any existing legal rights of any consumer or to deprive any person or entity of any existing private rights under the law.

17. The W3C Guidelines have been authored by the Web Accessibility Initiative (“WAI”) of the World Wide Web Consortium (“W3C”), an organization that develops technical specifications for the web’s infrastructure. These are industry guidelines that measure the accessibility of a web site to assistive technology commonly used by the blind and visually

-5-
impaired. The W3C Guidelines are divided into “Priority 1 Checkpoints,” which, according to
the WAI, “must” be implemented for accessibility, “Priority 2 Checkpoints,” which “should be”
implemented for accessibility, and “Priority 3 Checkpoints,” which “may be” implemented.

18. Priceline shall implement the following steps in order to ensure that the
PRICELINE.COM web site is accessible to the blind and visually impaired (where one of the
following subparagraphs refers to a W3C Guidelines Checkpoint, the Checkpoint number is
provided):

a. Priceline shall develop and maintain a web page and Frequently Asked
Questions page specifically for blind and visually impaired users.

b. Priceline shall educate its programmers about the applicable W3C
Guidelines referred to in Paragraph 18(d)-(z) herein and draft a set of code drafting guidelines to
implement these changes (the “code drafting guidelines”). Priceline shall distribute these code
drafting guidelines to its programmers and shall update them from time to time as necessary.

Any coding changes to the PRICELINE.COM web site implemented after execution of this
Assurance must be made based upon these code drafting guidelines.

c. Priceline shall include on its web site a link that will allow consumers
utilizing assistive technology to skip redundant navigation links. This “skip navigation link”
shall appear before the “PRICELINE.COM” link as the first link on each page of the
PRICELINE.COM web site, including the home page, and shall take the user to the first item of
content on each page after the last element in the navigation bar.

d. Checkpoint 1.1: Priceline shall establish text equivalents for all non-text
elements on the PRICELINE.COM web site. In particular with respect to maps, Priceline uses
two forms of maps on its web site: a general area map ("Region Map") and a detailed area map ("Zone Map"), examples of which are annexed hereto as Exhibits B and C, respectively. For the Region Map, Priceline shall provide text equivalents for the titles of the zones (i.e., "Fisherman’s Wharf") and shall provide a link to the Zone Map for each zone of a region.

e. Checkpoints 1.2 through 1.4: The PRICELINE.COM web site does not currently use server-side image maps, but does use visual tracks (i.e., visual presentations) and time-based multimedia presentations. Priceline shall synchronize equivalent alternatives (i.e., captions or auditory descriptions of the visual track) with the presentation. In addition, Priceline shall comply with Checkpoint 1.2 should it add server-side image maps to the PRICELINE.COM web site in the future.

f. Checkpoint 2.1: If there are instances on the PRICELINE.COM web site in which information is conveyed with color alone, Priceline shall revise the site to convey the information without color.

g. Checkpoint 3.1: Unless the images are properly labeled, Priceline shall, where an appropriate markup language exists, use markup rather than images to convey information.

h. Checkpoint 3.4: If there are instances on the PRICELINE.COM web site in which absolute units (i.e., measurements that remain constant) are being used that prevent resizing text, Priceline shall use relative units (i.e., measurements that can be changed by the user).

i. Checkpoint 3.5: Priceline shall use header elements to convey document structure, and shall use them according to specification (further information on “specification”
can be found under “1.2.1 Section headings” of the HTML Techniques for Web Content
Accessibility Guidelines 1.0, found on the W3C.ORG web site).

j. Checkpoint 4.1: The PRICELINE.COM web site does not currently have any changes in natural language, but Priceline shall identify such changes in its coding if it makes them in the future.

k. Checkpoints 5.1 and 5.2: Priceline shall determine whether data tables on the PRICELINE.COM web site comply with these Checkpoints and shall revise coding as necessary to bring the web site into compliance with these Checkpoints.

l. Checkpoint 5.3: Priceline shall use tables for layout on the PRICELINE.COM web site only where readable table summaries or empty summaries are provided. This requirement will not be included in the Audit Reports referred to herein in Paragraphs 21 and 22.

m. Checkpoint 6.1: The PRICELINE.COM web site does not currently use style sheets for content positioning. However, Priceline shall organize documents so that they may be read without style sheets, should Priceline use style sheets for positioning in the future.

n. Checkpoint 6.2: Priceline shall ensure that, with respect to dynamic content (i.e., web content that changes based upon input to a form or other user action without reloading the page), equivalent non-dynamic content is available to a blind or visually impaired user and that such equivalent content is updated when the dynamic content changes.

o. Checkpoint 6.3: Priceline shall label all links to any calendars appearing on the home page and several other pages of the site, so that screen reader users can understand that these calendars constitute a non-accessible alternative to the text input fields.
p. Checkpoint 7.1: The PRICELINE.COM web site does not currently cause the screen to flicker, and Priceline shall take appropriate steps to ensure that the screen does not flicker in the future.

q. Checkpoint 7.3: Until and unless browsers are widely available that allow users to freeze moving content, Priceline shall not employ moving content as to material information on its web site.

r. Checkpoint 7.5: Priceline shall include, on any “wait page” displayed on the PRICELINE.COM web site when a long operation (such as a search for available airline tickets) is requested by the user, a message for screen reader users to indicate that a wait is required and that the user will be redirected to the results page once the search is completed.

s. Checkpoint 9.1: The PRICELINE.COM web site does not currently use server-side image maps. Priceline shall ensure that its web site uses only client-side image maps, except where the regions cannot be defined with an available geometric shape.

t. Checkpoint 10.1: Priceline shall provide a “close window” link at the top of the window for all pop-up windows that appear on the PRICELINE.COM web site.

u. Checkpoint 10.2: Priceline shall include on the PRICELINE.COM web site label tags on form controls (i.e., coding elements which define a form) if the label text does not immediately precede the form control.

v. Checkpoint 11.4: Consistent with Checkpoint 11.4, if, after using its best efforts, Priceline is unable to make a particular web page on the PRICELINE.COM web site accessible to assistive technology, Priceline shall, with respect to that 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 page. If Priceline finds it necessary to exercise this option, then Priceline will provide advance written notice to the Attorney General prior to implementing Checkpoint 11.4, including a memorandum outlining the "best efforts," as that term is referred to and discussed under Checkpoint 11.4, used by Priceline and the reasons that Priceline was unable, through these efforts, to make its existing web site accessible. The Attorney General hereby expressly reserves its right to independently analyze whether "best efforts" have, in fact, been made and, where it reasonably believes that "best efforts" have not been made, to pursue any remedies, including without limitation, those arising under this Assurance.

w. Checkpoint 12.1: Priceline does not currently use frames on the PRICELINE.COM web site, but should Priceline add frames to the web site in the future, Priceline will title the frames.

x. Checkpoint 13.1: Priceline shall clearly identify the target of each link on the PRICELINE.COM web site.

y. Checkpoint 13.2: To add semantic information to its home page, Priceline shall provide metadata (i.e., on a web page, general information embedded in code about the content of the page) on the home page of the PRICELINE.COM web site.

z. Checkpoint 14.1: Priceline shall continue to strive to use the clearest and simplest language appropriate for the content of the PRICELINE.COM web site.

19. Priceline shall implement the initiatives set forth in Paragraph 18 in two phases. In the first phase, Priceline shall implement all accessibility measures described herein as to its "air product" (i.e., all pages within the "Airlines" section of the PRICELINE.COM web site) and
such measures have, in fact, been implemented as of the date of this Assurance. In the second phase, Priceline shall implement all accessibility measures as to the remainder of the PRICELINE.COM web site no later than October 15, 2004.

20. Within thirty (30) days of the date of this Assurance, and within thirty (30) days of October 15, 2004, Priceline shall forward to the Attorney General statements by a Priceline officer, confirming, respectively, that Priceline has implemented the changes described herein to its air product, and has implemented the accessibility measures with respect to the remainder of the PRICELINE.COM web site.

21. Priceline shall retain the services of an independent third party auditor, approved by the Attorney General, who shall review the PRICELINE.COM web site (the “Auditor”) to determine the various ways in which the PRICELINE.COM web site does or does not comply with Paragraph 18 (except Paragraph 18(1)) of this Assurance (“Audit Report”). In producing an Audit Report, the Auditor shall utilize: (i) the most current version of Freedom Scientific’s JAWS for Windows, unless the most current version has been available to the general public for ninety (90) days or less, in which case the Auditor shall utilize the version immediately previous to the most current version; and (ii) Microsoft’s Internet Explorer, version 6 or later. Priceline shall deliver to the Attorney General an Audit Report covering the “air product” by no later than July 1, 2004 and an Audit Report covering the remainder of the PRICELINE.COM web site by no later than January 17, 2005. The Attorney General must agree, in writing, to any extension of these deadlines.

22. In addition to the Audit Reports set forth in Paragraph 21, Priceline shall deliver to the Attorney General Audit Reports, each covering the entire PRICELINE.COM web site, by
no later than October 3, 2005 and October 3, 2006. The Attorney General must agree, in writing, to any extension of these deadlines. Prior to issuing to the Attorney General the Audit Reports referred to in Paragraphs 21 and 22 herein, the Auditor may discuss the results of the audit with Priceline and Priceline may rectify any issues uncovered during the audit process. In addition, to the extent that an Audit Report issued to the Attorney General identifies any material problems with the PRICELINE.COM web site, Priceline shall have forty-five (45) days to rectify any such problems before the Attorney General pursues its remedies under this Assurance.

23. This Assurance and any forebearance herein is conditioned upon Priceline’s compliance with the conditions herein, and upon the truthfulness of Priceline’s statements herein and during the course of the Inquiry.
24. The acceptance of this Assurance by the Attorney General shall not be deemed approval by the Attorney General of any of Priceline's business practices, and Priceline shall make no representation to the contrary.

IN WITNESS WHEREOF, the Attorney General and Priceline, intending to be legally bound hereby, have executed this Assurance on the dates written below.

Dated: New York, New York
April 8, 2004

ELIOT SPITZER
Attorney General of the
State of New York
120 Broadway
New York, New York 10271
(212) 416-8433

By:  
Elizabeth Nieliwocki
Assistant Attorney General
INTERNET BUREAU

By:  
Kenneth M. Dreifach
Assistant Attorney General
In Charge
INTERNET BUREAU

PRICELINE.COM INCORPORATED
800 Connecticut Avenue, Suite 8
Norwalk, Connecticut 06854
(203) 299-8000

By:  
P. Millones, Jr.
Executive Vice President
and General Counsel

STROOCK & STROOCK & LAVAN LLP
180 Maiden Lane
New York, New York 10038-4982
(212) 806-5400

By:  
James L. Bernard
Robert Abrams
Counsel for priceline.com Incorporated
These guidelines explain how to make Web content accessible to people with disabilities. The guidelines are intended for all Web content developers (page authors and site designers) and for developers of authoring tools. The primary goal of these guidelines is to promote accessibility. However, following them will also make Web content more available to all users, whatever user agent they are using (e.g., desktop browser, voice browser, mobile phone, automobile-based personal computer, etc.) or constraints they may be operating under (e.g., noisy surroundings, under- or over-illuminated rooms, in a hands-free environment, etc.). Following these guidelines will also help people find information on the Web more quickly. These guidelines do not discourage content developers from using images, video, etc., but rather explain how to make multimedia content more
accessible to a wide audience.

This is a reference document for accessibility principles and design ideas. Some of the strategies discussed in this document address certain Web internationalization and mobile access concerns. However, this document focuses on accessibility and does not fully address the related concerns of other W3C Activities. Please consult the W3C Mobile Access Activity home page and the W3C Internationalization Activity home page for more information.

This document is meant to be stable and therefore does not provide specific information about browser support for different technologies as that information changes rapidly. Instead, the Web Accessibility Initiative (WAI) Web site provides such information (refer to [WAI-UA-SUPPORT]).

This document includes an appendix that organizes all of the checkpoints by topic and priority. The checkpoints in the appendix link to their definitions in the current document. The topics identified in the appendix include images, multimedia, tables, frames, forms, and scripts. The appendix is available as either a tabular summary of checkpoints or as a simple list of checkpoints.

A separate document, entitled "Techniques for Web Content Accessibility Guidelines 1.0" ([TECHNIQUES]), explains how to implement the checkpoints defined in the current document. The Techniques Document discusses each checkpoint in more detail and provides examples using the Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), Synchronized Multimedia Integration Language (SMIL), and the Mathematical Markup Language (MathML). The Techniques Document also includes techniques for document validation and testing, and an index of HTML elements and attributes (and which techniques use them). The Techniques Document has been designed to track changes in technology and is expected to be updated more frequently than the current document. **Note.** Not all browsers or multimedia tools may support the features described in the guidelines. In particular, new features of HTML 4.0 or CSS 1 or CSS 2 may not be supported.

"Web Content Accessibility Guidelines 1.0" is part of a series of accessibility guidelines published by the Web Accessibility Initiative. The series also includes User Agent Accessibility Guidelines ([WAI-USERAGENT]) and Authoring Tool Accessibility Guidelines ([WAI-AUTOOLS]).

**Status of this document**

This document has been reviewed by W3C Members and other interested parties and has been endorsed by the Director as a W3C Recommendation. It is a stable document and may be used as reference material or cited as a normative reference from another documents. W3C's role in making the Recommendation is to draw attention to the specification and to promote its widespread deployment.
This enhances the functionality and universality of the Web.

The English version of this specification is the only normative version. However, for translations in other languages see http://www.w3.org/WAI/GL/WAI-WEBCONTENT-TRANSLATIONS.

The list of known errors in this document is available at http://www.w3.org/WAI/GL/WAI-WEBCONTENT-ERRATA. Please report errors in this document to wai-wcag-editor@w3.org.

A list of current W3C Recommendations and other technical documents can be found at http://www.w3.org/TR.

This document has been produced as part of the W3C Web Accessibility Initiative. The goal of the Web Content Guidelines Working Group is discussed in the Working Group charter.

Table of Contents

Abstract
Status of this document
1. Introduction
2. Themes of Accessible Design
   2.1 Ensuring Graceful Transformation
   2.2 Making Content Understandable and Navigable
3. How the Guidelines are Organized
   3.1 Document conventions
4. Priorities
5. Conformance
6. Web Content Accessibility Guidelines
   1. Provide equivalent alternatives to auditory and visual content.
   2. Don't rely on color alone.
   3. Use markup and style sheets and do so properly.
   4. Clarify natural language usage
   5. Create tables that transform gracefully.
   7. Ensure user control of time-sensitive content changes.
   8. Ensure direct accessibility of embedded user interfaces.
  10. Use interim solutions.
  11. Use W3C technologies and guidelines.
  12. Provide context and orientation information.
  13. Provide clear navigation mechanisms.
  14. Ensure that documents are clear and simple.

Appendix A. -- Validation
Appendix B. -- Glossary

http://www.w3.org/TR/WCAG10/
1. Introduction

For those unfamiliar with accessibility issues pertaining to Web page design, consider that many users may be operating 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 screen, a small screen, or a slow Internet connection.
- They may not speak or understand fluently the language in which the document is written.
- They may be in a situation where their eyes, ears, or hands are busy or interfered with (e.g., driving to work, working in a loud environment, etc.).
- They may have an early version of a browser, a different browser entirely, a voice browser, or a different operating system.

Content developers must consider these different situations during page design. While there are several situations to consider, each accessible design choice generally benefits several disability groups at once and the Web community as a whole. For example, by using *stylesheets* to control font styles and eliminating the FONT element, HTML authors will have more control over their pages, make those pages more accessible to people with low vision, and by sharing the style sheets, will often shorten page download times for all users.

The guidelines discuss accessibility issues and provide accessible design solutions. They address typical scenarios (similar to the font style example) that may pose problems for users with certain disabilities. For example, the first guideline explains how content developers can make images accessible. Some users may not be able to see images, others may use text-based browsers that do not support images, while others may have turned off support for images (e.g., due to a slow Internet connection). The guidelines do not suggest avoiding images as a way to improve accessibility. Instead, they explain that providing a *text equivalent* of the image will make it accessible.

How does a text equivalent make the image accessible? Both words in "text equivalent" are important:

http://www.w3.org/TR/WCAG10/
Text content can be presented to the user as synthesized speech, braille, and visually-displayed text. Each of these three mechanisms uses a different sense -- ears for synthesized speech, tactile for braille, and eyes for visually-displayed text -- making the information accessible to groups representing a variety of sensory and other disabilities.

In order to be useful, the text must convey the same function or purpose as the image. For example, consider a text equivalent for a photographic image of the Earth as seen from outer space. If the purpose of the image is mostly that of decoration, then the text "Photograph of the Earth as seen from outer space" might fulfill the necessary function. If the purpose of the photograph is to illustrate specific information about world geography, then the text equivalent should convey that information. If the photograph has been designed to tell the user to select the image (e.g., by clicking on it) for information about the earth, equivalent text would be "Information about the Earth". Thus, if the text conveys the same function or purpose for the user with a disability as the image does for other users, then it can be considered a text equivalent.

Note that, in addition to benefitting users with disabilities, text equivalents can help all users find pages more quickly, since search robots can use the text when indexing the pages.

While Web content developers must provide text equivalents for images and other multimedia content, it is the responsibility of user agents (e.g., browsers and assistive technologies such as screen readers, braille displays, etc.) to present the information to the user.

Non-text equivalents of text (e.g., icons, pre-recorded speech, or a video of a person translating the text into sign language) can make documents accessible to people who may have difficulty accessing written text, including many individuals with cognitive disabilities, learning disabilities, and deafness. Non-text equivalents of text can also be helpful to non-readers. An auditory description is an example of a non-text equivalent of visual information. An auditory description of a multimedia presentation's visual track benefits people who cannot see the visual information.

2. Themes of Accessible Design

The guidelines address two general themes: ensuring graceful transformation, and making content understandable and navigable.

2.1 Ensuring Graceful Transformation

By following these guidelines, content developers can create pages that transform gracefully. Pages that transform gracefully remain accessible despite any of the constraints described in the introduction, including physical, sensory, and cognitive disabilities, work constraints, and technological barriers. Here are some keys to
designing pages that transform gracefully:

- Separate structure from presentation (refer to the difference between content, structure, and presentation).
- Provide text (including text equivalents). Text can be rendered in ways that are available to almost all browsing devices and accessible to almost all users.
- Create documents that work even if the user cannot see and/or hear. Provide information that serves the same purpose or function as audio or video in ways suited to alternate sensory channels as well. This does not mean creating a prerecorded audio version of an entire site to make it accessible to users who are blind. Users who are blind can use screen reader technology to render all text information in a page.
- Create documents that do not rely on one type of hardware. Pages should be usable by people without mice, with small screens, low resolution screens, black and white screens, no screens, with only voice or text output, etc.

The theme of graceful transformation is addressed primarily by guidelines 1 to 11.

2.2 Making Content Understandable and Navigable

Content developers should make content understandable and navigable. This includes not only making the language clear and simple, but also providing understandable mechanisms for navigating within and between pages. Providing navigation tools and orientation information in pages will maximize accessibility and usability. Not all users can make use of visual clues such as image maps, proportional scroll bars, side-by-side frames, or graphics that guide sighted users of graphical desktop browsers. Users also lose contextual information when they can only view a portion of a page, either because they are accessing the page one word at a time (speech synthesis or braille display), or one section at a time (small display, or a magnified display). Without orientation information, users may not be able to understand very large tables, lists, menus, etc.

The theme of making content understandable and navigable is addressed primarily in guidelines 12 to 14.

3. How the Guidelines are Organized

This document includes fourteen guidelines, or general principles of accessible design. Each guideline includes:

- The guideline number.
- The statement of the guideline.
- Guideline navigation links. Three links allow navigation to the next guideline (right arrow icon), the previous guideline (left arrow icon), or the current
Guideline's position in the table of contents (up arrow icon).
- The rationale behind the guideline and some groups of users who benefit
  from it.
- A list of checkpoint definitions.

The checkpoint definitions in each guideline explain how the guideline applies in
typical content development scenarios. Each checkpoint definition includes:

- The checkpoint number.
- The statement of the checkpoint.
- The priority of the checkpoint. Priority 1 checkpoints are highlighted through
  the use of style sheets.
- Optional informative notes, clarifying examples, and cross references to
  related guidelines or checkpoints.
- A link to a section of the Techniques Document ([TECHNIQUES]) where
  implementations and examples of the checkpoint are discussed.

Each checkpoint is intended to be specific enough so that someone reviewing a
page or site may verify that the checkpoint has been satisfied.

3.1 Document conventions

The following editorial conventions are used throughout this document:

- Element names are in uppercase letters.
- Attribute names are quoted in lowercase letters.
- Links to definitions are highlighted through the use of style sheets.

4. Priorities

Each checkpoint has a priority level assigned by the Working Group based on the
checkpoint's impact on accessibility.

[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.

Some checkpoints specify a priority level that may change under certain (indicated) conditions.

5. Conformance

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 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: "Web Content Accessibility Guidelines 1.0"
- The guidelines URI: http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505
- The conformance level satisfied: "A", "Double-A", or "Triple-A".
- The scope covered by the claim (e.g., page, site, or defined portion of a site.).

**Example of Form 1:**

This page conforms to W3C's "Web Content Accessibility Guidelines 1.0", available at http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505, level Double-A.

**Form 2:** Include, on each page claiming conformance, one of three icons provided by W3C and link the icon to the appropriate W3C explanation of the claim. Information about the icons and how to insert them in pages is available at [WCAG-ICONS].

6. Web Content Accessibility Guidelines

Guideline 1. Provide equivalent alternatives to http://www.w3.org/TR/WCAG10/
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:

   http://www.w3.org/TR/WCAG10/
• 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.

Techniques for checkpoint 1.1

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.
Techniques for checkpoint 1.2

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.
Techniques for checkpoint 1.3

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]
Techniques for checkpoint 1.4

1.5 Until user agents render text equivalents for client-side image map links, provide redundant text links for each active region of a client-side image map. [Priority 3]
Refer also to checkpoint 1.2 and checkpoint 9.1.
Techniques for checkpoint 1.5

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]
   Techniques for checkpoint 2.1

2.2 Ensure that foreground and background color combinations provide sufficient contrast when viewed by someone having color deficits or when viewed on a black and white screen. [Priority 2 for images, Priority 3 for text].
   Techniques for checkpoint 2.2

Guideline 3. Use markup and style sheets and do so properly.

Mark up documents with the proper structural elements. Control presentation with style sheets rather than with presentation elements and attributes.

Using markup improperly -- not according to specification -- hinders accessibility. Misusing markup for a presentation effect (e.g., using a table for layout or a header to change the font size) makes it difficult for users with specialized software to understand the organization of the page or to navigate through it. Furthermore, using presentation markup rather than structural markup to convey structure (e.g., constructing what looks like a table of data with an HTML PRE element) makes it difficult to render a page intelligibly to other devices (refer to the description of difference between content, structure, and presentation).

Content developers may be tempted to use (or misuse) constructs that achieve a desired formatting effect on older browsers. They must be aware that these practices cause accessibility problems and must consider whether the formatting effect is so critical as to warrant making the document inaccessible to some users.

At the other extreme, content developers must not sacrifice appropriate markup because a certain browser or assistive technology does not process it correctly. For example, it is appropriate to use the TABLE element in HTML to mark up tabular information even though some older screen readers may not handle side-by-side text correctly (refer to checkpoint 10.3). Using TABLE correctly and creating tables that transform gracefully (refer to guideline 5) makes it possible for software to render tables other than as two-dimensional grids.

Checkpoints:
3.1 When an appropriate markup language exists, use markup rather than images to convey information. [Priority 2]

- For example, use MathML to mark up mathematical equations, and style sheets to format text and control layout. Also, avoid using images to represent text -- use text and style sheets instead. Refer also to guideline 6 and guideline 11.

Techniques for checkpoint 3.1

3.2 Create documents that validate to published formal grammars. [Priority 2]

- For example, include a document type declaration at the beginning of a document that refers to a published DTD (e.g., the strict HTML 4.0 DTD).

Techniques for checkpoint 3.2

3.3 Use style sheets to control layout and presentation. [Priority 2]

- For example, use the CSS 'font' property instead of the HTML FONT element to control font styles.

Techniques for checkpoint 3.3

3.4 Use relative rather than absolute units in markup language attribute values and style sheet property values. [Priority 2]

- For example, in CSS, use 'em' or percentage lengths rather than 'pt' or 'cm', which are absolute units. If absolute units are used, validate that the rendered content is usable (refer to the section on validation).

Techniques for checkpoint 3.4

3.5 Use header elements to convey document structure and use them according to specification. [Priority 2]

- For example, in HTML, use H2 to indicate a subsection of H1. Do not use headers for font effects.

Techniques for checkpoint 3.5

3.6 Mark up lists and list items properly. [Priority 2]

- For example, in HTML, nest OL, UL, and DL lists properly.

Techniques for checkpoint 3.6

3.7 Mark up quotations. Do not use quotation markup for formatting effects such as indentation. [Priority 2]

- For example, in HTML, use the Q and BLOCKQUOTE elements to markup short and longer quotations, respectively.

Techniques for checkpoint 3.7

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".
   Techniques for checkpoint 4.1

4.2 Specify the expansion of each abbreviation or acronym in a document where it first occurs. [Priority 3]
   For example, in HTML, use the "title" attribute of the ABBR and ACRONYM elements. Providing the expansion in the main body of the document also helps document usability.
   Techniques for checkpoint 4.2

4.3 Identify the primary natural language of a document. [Priority 3]
   For example, in HTML set the "lang" attribute on the HTML element. In XML, use "xml:lang". Server operators should configure servers to take advantage of HTTP content negotiation mechanisms ([RFC2068], section 14.13) so that clients can automatically retrieve documents of the preferred language.
   Techniques for checkpoint 4.3

**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.
   Techniques for checkpoint 5.1
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.
   Techniques for checkpoint 5.2
5.3 Do not use tables for layout unless the table makes sense when linearized. Otherwise, if the table does not make sense, provide an alternative equivalent (which may be a linearized version). [Priority 2]
   Note. Once user agents support style sheet positioning, tables should not be used for layout. Refer also to checkpoint 3.3.
   Techniques for checkpoint 5.3
5.4 If a table is used for layout, do not use any structural markup for the purpose of visual formatting. [Priority 2]
   For example, in HTML do not use the TH element to cause the content of a (non-table header) cell to be displayed centered and in bold.
   Techniques for checkpoint 5.4
5.5 Provide summaries for tables. [Priority 3]
   For example, in HTML, use the "summary" attribute of the TABLE element.
   Techniques for checkpoint 5.5
5.6 Provide abbreviations for header labels. [Priority 3]
   For example, in HTML, use the "abbr" attribute on the TH element.
   Techniques for checkpoint 5.6

Refer also to checkpoint 10.3.

Guideline 6. Ensure that pages featuring new
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.
   - Techniques for checkpoint 6.1

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

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.
   - Techniques for checkpoint 6.3

6.4 For scripts and applets, ensure that event handlers are input device-independent. [Priority 2]
   - Refer to the definition of device independence.
   - Techniques for checkpoint 6.4

6.5 Ensure that dynamic content is accessible or provide an alternative presentation or page. [Priority 2]
   - For example, in HTML, use NOFRAMES at the end of each frameset. For some applications, server-side scripts may be more accessible than client-side scripts.
   - Techniques for checkpoint 6.5

Refer also to checkpoint 11.4.
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).

Techniques for checkpoint 7.1

7.2 Until user agents allow users to control blinking, avoid causing content to blink (i.e., change presentation at a regular rate, such as turning on and off). [Priority 2]

Techniques for checkpoint 7.2

7.3 Until user agents allow users to freeze moving content, avoid movement in pages. [Priority 2]

   When a page includes moving content, provide a mechanism within a script or applet to allow users to freeze motion or updates. Using stylesheet with scripting to create movement allows users to turn off or override the effect more easily. Refer also to guideline 8.

Techniques for checkpoint 7.3

7.4 Until user agents provide the ability to stop the refresh, do not create periodically auto-refreshing pages. [Priority 2]

   For example, in HTML, don't cause pages to auto-refresh with "HTTP-EQUIV=refresh" until user agents allow users to turn off the feature.

Techniques for checkpoint 7.4

7.5 Until user agents provide the ability to stop auto-redirect, do not use markup to redirect pages automatically. Instead, configure the server to perform redirects. [Priority 2]

Techniques for checkpoint 7.5

Note. The BLINK and MARQUEE elements are not defined in any W3C HTML specification and should not be used. Refer also to guideline 11.
Guideline 8. Ensure direct accessibility of embedded user interfaces.

Ensure that the user interface follows principles of accessible design: device-independent access to functionality, keyboard operability, self-voicing, etc.

When an embedded object has its "own interface", the interface -- like the interface to the browser itself -- must be accessible. If the interface of the embedded object cannot be made accessible, an alternative accessible solution must be provided.

Note. For information about accessible interfaces, please consult the User Agent Accessibility Guidelines ([WAI-USERAGENT]) and the Authoring Tool Accessibility Guidelines ([WAI-AUTOOL]).

Checkpoint:

8.1 Make programmatic elements such as scripts and applets directly accessible or compatible with assistive technologies [Priority 1 if functionality is important and not presented elsewhere, otherwise Priority 2.]

Refer also to guideline 6.

Techniques for checkpoint 8.1


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.
   Techniques for checkpoint 9.1

9.2 Ensure that any element that has its own interface can be operated in a device-independent manner. [Priority 2]
   Refer to the definition of device independence.
   Refer also to guideline 8.
   Techniques for checkpoint 9.2

9.3 For scripts, specify logical event handlers rather than device-dependent event handlers. [Priority 2]
   Techniques for checkpoint 9.3

9.4 Create a logical tab order through links, form controls, and objects. [Priority 3]
   For example, in HTML, specify tab order via the "tabindex" attribute or ensure a logical page design.
   Techniques for checkpoint 9.4

9.5 Provide keyboard shortcuts to important links (including those in client-side image maps), form controls, and groups of form controls. [Priority 3]
   For example, in HTML, specify shortcuts via the "accesskey" attribute.
   Techniques for checkpoint 9.5

Guideline 10. Use interim solutions.

Use interim accessibility solutions so that assistive technologies and older browsers will operate correctly.

For example, older browsers do not allow users to navigate to empty edit boxes. Older screen readers read lists of consecutive links as one link. These active elements are therefore difficult or impossible to access. Also, changing the current window or popping up new windows can be very disorienting to users who cannot see that this has happened.

Note. The following checkpoints apply until user agents (including assistive technologies) address these issues. These checkpoints are classified as "interim", meaning that the Web Content Guidelines Working Group considers them to be valid and necessary to Web accessibility as of the publication of this document. However, the Working Group does not expect these checkpoints to be necessary in the future, once Web technologies have incorporated anticipated features or capabilities.

http://www.w3.org/TR/WCAG10/ 4/8/04
Checkpoints:

10.1 Until user agents allow users to turn off spawned windows, do not cause pop-ups or other windows to appear and do not change the current window without informing the user. [Priority 2]
   For example, in HTML, avoid using a frame whose target is a new window.
   Techniques for checkpoint 10.1

10.2 Until user agents support explicit associations between labels and form controls, for all form controls with implicitly associated labels, ensure that the label is properly positioned. [Priority 2]
   The label must immediately precede its control on the same line (allowing more than one control/label per line) or be in the line preceding the control (with only one label and one control per line).
   Refer also to checkpoint 12.4.
   Techniques for checkpoint 10.2

10.3 Until user agents (including assistive technologies) render side-by-side text correctly, provide a linear text alternative (on the current page or some other) for all tables that lay out text in parallel, word-wrapped columns. [Priority 3]
   Note. Please consult the definition of linearized table. This checkpoint benefits people with user agents (such as some screen readers) that are unable to handle blocks of text presented side-by-side; the checkpoint should not discourage content developers from using tables to represent tabular information.
   Techniques for checkpoint 10.3

10.4 Until user agents handle empty controls correctly, include default, place-holding characters in edit boxes and text areas. [Priority 3]
   For example, in HTML, do this for TEXTAREA and INPUT.
   Techniques for checkpoint 10.4

10.5 Until user agents (including assistive technologies) render adjacent links distinctly, include non-link, printable characters (surrounded by spaces) between adjacent links. [Priority 3]
   Techniques for checkpoint 10.5

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:

http://www.w3.org/TR/WCAG10/ 4/8/04
- 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:**

11.1 Use W3C technologies when they are available and appropriate for a task and use the latest versions when supported. [Priority 2]
Refer to the list of references for information about where to find the latest W3C specifications and [WAI-UA-SUPPORT] for information about user agent support for W3C technologies.
Techniques for checkpoint 11.1

11.2 Avoid deprecated features of W3C technologies. [Priority 2]
For example, in HTML, don't use the deprecated FONT element; use style sheets instead (e.g., the 'font' property in CSS).
Techniques for checkpoint 11.2

11.3 Provide information so that users may receive documents according to their preferences (e.g., language, content type, etc.) [Priority 3]
**Note.** Use content negotiation where possible.
Techniques for checkpoint 11.3

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]
Techniques for checkpoint 11.4

**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.


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.
Techniques for checkpoint 12.1

12.2 Describe the purpose of frames and how frames relate to each other if it is not obvious by frame titles alone. [Priority 2]
For example, in HTML, use "longdesc," or a description link.
Techniques for checkpoint 12.2

12.3 Divide large blocks of information into more manageable groups where natural and appropriate. [Priority 2]
For example, in HTML, use OPTGROUP to group OPTION elements inside a SELECT; group form controls with FIELDSET and LEGEND; use nested lists where appropriate; use headings to structure documents, etc. Refer also to guideline 3.
Techniques for checkpoint 12.3

12.4 Associate labels explicitly with their controls. [Priority 2]
For example, in HTML use LABEL and its "for" attribute.
Techniques for checkpoint 12.4

Guideline 13. Provide clear navigation mechanisms.
Provide clear and consistent navigation mechanisms -- orientation information, navigation bars, a site map, etc. -- to increase the likelihood that a person will find what they are looking for at a site.

Clear and consistent navigation mechanisms are important to people with cognitive disabilities or blindness, and benefit all users.

Checkpoints:

13.1 Clearly identify the target of each link. [Priority 2]

*Link text* should be meaningful enough to make sense when read out of context -- either on its own or as part of a sequence of links. Link text should also be terse.

For example, in HTML, write "Information about version 4.3" instead of "click here". In addition to clear link text, content developers may further clarify the target of a link with an informative link title (e.g., in HTML, the "title" attribute).

Techniques for checkpoint 13.1

13.2 Provide metadata to add semantic information to pages and sites. [Priority 2]

For example, use RDF ([RDF]) to indicate the document's author, the type of content, etc.

*Note.* Some HTML user agents can build navigation tools from document relations described by the HTML LINK element and "rel" or "rev" attributes (e.g., rel="next", rel="previous", rel="index", etc.). Refer also to checkpoint 13.5.

Techniques for checkpoint 13.2

13.3 Provide information about the general layout of a site (e.g., a site map or table of contents). [Priority 2]

In describing site layout, highlight and explain available accessibility features.

Techniques for checkpoint 13.3

13.4 Use navigation mechanisms in a consistent manner. [Priority 2]

Techniques for checkpoint 13.4

13.5 Provide navigation bars to highlight and give access to the navigation mechanism. [Priority 3]

Techniques for checkpoint 13.5

13.6 Group related links, identify the group (for user agents), and, *until user agents* do so, provide a way to bypass the group. [Priority 3]

Techniques for checkpoint 13.6

13.7 If search functions are provided, enable different types of searches for different skill levels and preferences. [Priority 3]

Techniques for checkpoint 13.7

13.8 Place distinguishing information at the beginning of headings, paragraphs, lists, etc. [Priority 3]

*Note.* This is commonly referred to as "front-loading" and is especially helpful for people accessing information with serial
devices such as speech synthesizers.

Techniques for checkpoint 13.8

13.9 Provide information about document collections (i.e., documents comprising multiple pages.). [Priority 3]
For example, in HTML specify document collections with the LINK element and the "rel" and "rev" attributes. Another way to create a collection is by building an archive (e.g., with zip, tar and gzip, stuffit, etc.) of the multiple pages.

Note. The performance improvement gained by offline processing can make browsing much less expensive for people with disabilities who may be browsing slowly.

Techniques for checkpoint 13.9

13.10 Provide a means to skip over multi-line ASCII art. [Priority 3]
Refer to checkpoint 1.1 and the example of ascii art in the glossary.

Techniques for checkpoint 13.10

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]
Techniques for checkpoint 14.1

14.2 Supplement text with graphic or auditory presentations where they will facilitate comprehension of the page. [Priority 3]
Refer also to guideline 1.
Techniques for checkpoint 14.2

14.3 Create a style of presentation that is consistent across pages. [Priority 3]
Appendix A. -- Validation

Validate accessibility with automatic tools and human review. Automated methods are generally rapid and convenient but cannot identify all accessibility issues. Human review can help ensure clarity of language and ease of navigation.

Begin using validation methods at the earliest stages of development. Accessibility issues identified early are easier to correct and avoid.

Following are some important validation methods, discussed in more detail in the section on validation in the Techniques Document.

1. Use an automated accessibility tool and browser validation tool. Please note that software tools do not address all accessibility issues, such as the meaningfulness of link text, the applicability of a text equivalent, etc.
2. Validate syntax (e.g., HTML, XML, etc.).
3. Validate style sheets (e.g., CSS).
4. Use a text-only browser or emulator.
5. Use multiple graphic browsers, with:
   - sounds and graphics loaded,
   - graphics not loaded,
   - sounds not loaded,
   - no mouse,
   - frames, scripts, style sheets, and applets not loaded
6. Use several browsers, old and new.
7. Use a self-voicing browser, a screen reader, magnification software, a small display, etc.
8. Use spell and grammar checkers. A person reading a page with a speech synthesizer may not be able to decipher the synthesizer's best guess for a word with a spelling error. Eliminating grammar problems increases comprehension.
9. Review the document for clarity and simplicity. Readability statistics, such as those generated by some word processors may be useful indicators of clarity and simplicity. Better still, ask an experienced (human) editor to review written content for clarity. Editors can also improve the usability of documents by identifying potentially sensitive cultural issues that might arise due to language or icon usage.
10. Invite people with disabilities to review documents. Expert and novice users with disabilities will provide valuable feedback about accessibility or usability problems and their severity.

http://www.w3.org/TR/WCAG10/
Appendix B. -- Glossary

**Accessible**

Content is accessible when it may be used by someone with a disability.

**Applet**

A program inserted into a Web page.

**Assistive technology**

Software or hardware that has been specifically designed to assist people with disabilities in carrying out daily activities. Assistive technology includes wheelchairs, reading machines, devices for grasping, etc. In the area of Web Accessibility, common software-based assistive technologies include screen readers, screen magnifiers, speech synthesizers, and voice input software that operate in conjunction with graphical desktop browsers (among other user agents). Hardware assistive technologies include alternative keyboards and pointing devices.

**ASCII art**

ASCII art refers to text characters and symbols that are combined to create an image. For example ";-)" is the smiley emoticon. The following is an ascii figure showing the relationship between flash frequency and photoconvulsive response in patients with eyes open and closed [skip over ascii figure or consult a description of chart]:

```
0 5 10 15 20 25 30 35 40 45 50 55 60 65 70
Flash frequency (Hertz)
```

**Authoring tool**

HTML editors, document conversion tools, tools that generate Web content from databases are all authoring tools. Refer to the "Authoring Tool Accessibility Guidelines" ([WAI-AUTOOLS]) for information about developing accessible tools.

http://www.w3.org/TR/WCAG10/
**Backward compatible**

Design that continues to work with earlier versions of a language, program, etc.

**Braille**

Braille uses six raised dots in different patterns to represent letters and numbers to be read by people who are blind with their fingertips. The word "Accessible" in braille follows:

```
A

B R A I L L E
```

A *braille display*, commonly referred to as a "dynamic braille display," raises or lowers dot patterns on command from an electronic device, usually a computer. The result is a line of braille that can change from moment to moment. Current dynamic braille displays range in size from one cell (six or eight dots) to an eighty-cell line, most having between twelve and twenty cells per line.

**Content developer**

Someone who authors Web pages or designs Web sites.

**Deprecated**

A deprecated element or attribute is one that has been outdated by newer constructs. Deprecated elements may become obsolete in future versions of HTML. The [index of HTML elements and attributes in the Techniques Document](http://www.w3.org/TR/WCAG10/) indicates which elements and attributes are deprecated in HTML 4.0.

Authors should avoid using deprecated elements and attributes. User agents should continue to support for reasons of backward compatibility.

**Device independent**

Users must be able to interact with a user agent (and the document it renders) using the supported input and output devices of their choice and according to their needs. Input devices may include pointing devices, keyboards, braille devices, head wands, microphones, and others. Output devices may include monitors, speech synthesizers, and braille devices.

Please note that "device-independent support" does not mean that user agents must support every input or output device. User agents should offer redundant input and output mechanisms for those devices that are supported. For example, if a user agent supports keyboard and mouse input, users should be able to interact with all features using either the keyboard or
the mouse.

**Document Content, Structure, and Presentation**

The content of a document refers to what it says to the user through natural language, images, sounds, movies, animations, etc. The structure of a document is how it is organized logically (e.g., by chapter, with an introduction and table of contents, etc.). An *element* (e.g., P, STRONG, BLOCKQUOTE in HTML) that specifies document structure is called a structural element. The presentation of a document is how the document is rendered (e.g., as print, as a two-dimensional graphical presentation, as an text-only presentation, as synthesized speech, as braille, etc.) An *element* that specifies document presentation (e.g., B, FONT, CENTER) is called a presentation element.

Consider a document header, for example. The content of the header is what the header says (e.g., "Sailboats"). In HTML, the header is a structural element marked up with, for example, an H2 element. Finally, the presentation of the header might be a bold block text in the margin, a centered line of text, a title spoken with a certain voice style (like an aural font), etc.

**Dynamic HTML (DHTML)**

DHTML is the marketing term applied to a mixture of standards including HTML, *style sheets*, the Document Object Model [DOM1] and scripting. However, there is no W3C specification that formally defines DHTML. Most guidelines may be applicable to applications using DHTML, however the following guidelines focus on issues related to scripting and style sheets: guideline 1, guideline 3, guideline 6, guideline 7, and guideline 9.

**Element**

This document uses the term "element" both in the strict SGML sense (an element is a syntactic construct) and more generally to mean a type of content (such as video or sound) or a logical construct (such as a header or list). The second sense emphasizes that a guideline inspired by HTML could easily apply to another markup language.

Note that some (SGML) elements have content that is rendered (e.g., the P, LI, or TABLE elements in HTML), some are replaced by external content (e.g., IMG), and some affect processing (e.g., STYLE and SCRIPT cause information to be processed by a style sheet or script engine). An element that causes text characters to be part of the document is called a text element.

**Equivalent**

Content is "equivalent" to other content when both fulfill essentially the same function or purpose upon presentation to the user. In the context of this document, the equivalent must fulfill essentially the same function for the
person with a disability (at least insofar as is feasible, given the nature of the
disability and the state of technology), as the primary content does for the
person without any disability. For example, the text "The Full Moon" might
convey the same information as an image of a full moon when presented to
users. Note that equivalent information focuses on **fulfilling the same
function**. If the image is part of a link and understanding the image is crucial
to guessing the link target, an equivalent must also give users an idea of the
link target. Providing equivalent information for inaccessible content is one of
the primary ways authors can make their documents accessible to people
with disabilities.

As part of fulfilling the same function of content an equivalent may involve a
description of that content (i.e., what the content looks like or sounds like).
For example, in order for users to understand the information conveyed by a
complex chart, authors should describe the visual information in the chart.
Since text content can be presented to the user as synthesized speech,
braille, and visually-displayed text, these guidelines require **text equivalents**
for graphic and audio information. Text equivalents must be written so that
they convey all essential content. **Non-text equivalents** (e.g., an auditory
description of a visual presentation, a video of a person telling a story using
sign language as an equivalent for a written story, etc.) also improve
accessibility for people who cannot access visual information or written text,
including many individuals with blindness, cognitive disabilities, learning
disabilities, and deafness.

Equivalent information may be provided in a number of ways, including
through attributes (e.g., a text value for the "alt" attribute in HTML and SMIL),
as part of element content (e.g., the OBJECT in HTML), as part of the
document's prose, or via a linked document (e.g., designated by the
"longdesc" attribute in HTML or a **description link**). Depending on the
complexity of the equivalent, it may be necessary to combine techniques
(e.g., use "alt" for an abbreviated equivalent, useful to familiar readers, in
addition to "longdesc" for a link to more complete information, useful to first-
time readers). The details of how and when to provide equivalent information
are part of the Techniques Document ([TECHNIQUES]).

A **text transcript** is a text equivalent of audio information that includes
spoken words and non-spoken sounds such as sound effects. A **caption** is a
text transcript for the audio track of a video presentation that is synchronized
with the video and audio tracks. Captions are generally rendered visually by
being superimposed over the video, which benefits people who are deaf and
hard-of-hearing, and anyone who cannot hear the audio (e.g., when in a
crowded room). A **collated text transcript** combines (collates) captions with
text descriptions of video information (descriptions of the actions, body
language, graphics, and scene changes of the video track). These text
equivalents make presentations accessible to people who are deaf-blind and
to people who cannot play movies, animations, etc. It also makes the
information available to search engines.

One example of a non-text equivalent is an **auditory description** of the key

http://www.w3.org/TR/WCAG10/ 4/8/04
visual elements of a presentation. The description is either a prerecorded human voice or a synthesized voice (recorded or generated on the fly). The auditory description is synchronized with the audio track of the presentation, usually during natural pauses in the audio track. Auditory descriptions include information about actions, body language, graphics, and scene changes.

**Image**

A graphical presentation.

**Image map**

An image that has been divided into regions with associated actions. Clicking on an active region causes an action to occur.

When a user clicks on an active region of a client-side image map, the user agent calculates in which region the click occurred and follows the link associated with that region. Clicking on an active region of a server-side image map causes the coordinates of the click to be sent to a server, which then performs some action.

Content developers can make client-side image maps accessible by providing device-independent access to the same links associated with the image map's regions. Client-side image maps allow the user agent to provide immediate feedback as to whether or not the user's pointer is over an active region.

**Important**

Information in a document is important if understanding that information is crucial to understanding the document.

**Linearized table**

A table rendering process where the contents of the cells become a series of paragraphs (e.g., down the page) one after another. The paragraphs will occur in the same order as the cells are defined in the document source. Cells should make sense when read in order and should include structural elements (that create paragraphs, headers, lists, etc.) so the page makes sense after linearization.

**Link text**

The rendered text content of a link.

**Natural Language**

Spoken, written, or signed human languages such as French, Japanese, American Sign Language, and braille. The natural language of content may be indicated with the "lang" attribute in HTML ([HTML40], section 8.1) and the "xml:lang" attribute in XML ([XML], section 2.12).
Navigation Mechanism

A navigation mechanism is any means by which a user can navigate a page or site. Some typical mechanisms include:

**navigation bars**
A navigation bar is a collection of links to the most important parts of a document or site.

**site maps**
A site map provides a global view of the organization of a page or site.

**tables of contents**
A table of contents generally lists (and links to) the most important sections of a document.

Personal Digital Assistant (PDA)

A PDA is a small, portable computing device. Most PDAs are used to track personal data such as calendars, contacts, and electronic mail. A PDA is generally a handheld device with a small screen that allows input from various sources.

Screen magnifier

A software program that magnifies a portion of the screen, so that it can be more easily viewed. Screen magnifiers are used primarily by individuals with low vision.

Screen reader

A software program that reads the contents of the screen aloud to a user. Screen readers are used primarily by individuals who are blind. Screen readers can usually only read text that is printed, not painted, to the screen.

Style sheets

A style sheet is a set of statements that specify presentation of a document. Style sheets may have three different origins: they may be written by content providers, created by users, or built into user agents. In CSS ([CSS2]), the interaction of content provider, user, and user agent style sheets is called the cascade.

**Presentation markup** is markup that achieves a stylistic (rather than structuring) effect such as the B or I elements in HTML. Note that the STRONG and EM elements are not considered presentation markup since they convey information that is independent of a particular font style.

Tabular information

When tables are used to represent logical relationships among data -- text, numbers, images, etc., that information is called "tabular information" and the tables are called "data tables". The relationships expressed by a table may be rendered visually (usually on a two-dimensional grid), aurally (often preceding cells with header information), or in other formats.
**Until user agents ...**

In most of the checkpoints, content developers are asked to ensure the accessibility of their pages and sites. However, there are accessibility needs that would be more appropriately met by *user agents* (including *assistive technologies*). As of the publication of this document, not all user agents or assistive technologies provide the accessibility control users require (e.g., some user agents may not allow users to turn off blinking content, or some screen readers may not handle tables well). Checkpoints that contain the phrase "until user agents ..." require content developers to provide additional support for accessibility until most user agents readily available to their audience include the necessary accessibility features.

**Note.** The W3C WAI Web site (refer to [WAI-UA-SUPPORT](http://www.w3.org/WAI/UA-SUPPORT)) provides information about user agent support for accessibility features. Content developers are encouraged to consult this page regularly for updated information.

**User agent**

Software to access Web content, including desktop graphical browsers, text browsers, voice browsers, mobile phones, multimedia players, plug-ins, and some software assistive technologies used in conjunction with browsers such as screen readers, screen magnifiers, and voice recognition software.

---

**Acknowledgments**

**Web Content Guidelines Working Group Co-Chairs:**
- Chuck Letourneau, Starling Access Services
- Gregg Vanderheiden, Trace Research and Development

**W3C Team contacts:**
- Judy Brewer and Daniel Dardailler

We wish to thank the following people who have contributed their time and valuable comments to shaping these guidelines:

The original draft of this document is based on "The Unified Web Site Accessibility Guidelines" ([UWSAG](http://www.w3.org/WAI/UA-SUPPORT)) compiled by the Trace R & D Center at the University of Wisconsin. That document includes a list of additional contributors.
References

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

[CSS1]
The latest version of CSS1 is available at: http://www.w3.org/TR/REC-CSS1.

[CSS2]
The latest version of CSS2 is available at: http://www.w3.org/TR/REC-CSS2.

[DOM1]
The latest version of DOM Level 1 is available at: http://www.w3.org/TR/REC-DOM-Level-1

[HTML40]
The latest version of HTML 4.0 is available at: http://www.w3.org/TR/REC-html40.

[HTML32]

[MATHML]
The latest version of MathML 1.0 is available at: http://www.w3.org/TR/REC-MathML.

[PNG]

[RDF]
The latest version of RDF 1.0 is available at: http://www.w3.org/TR/REC-rdf-syntax

[RFC2068]

[SMIL]
The latest version of SMIL 1.0 is available at: http://www.w3.org/TR/REC-smil

[TECHNIQUES]
"Techniques for Web Content Accessibility Guidelines 1.0", W. Chisholm, G. Vanderheiden, I. Jacobs, eds. This document explains how to implement the checkpoints defined in "Web Content Accessibility Guidelines 1.0". The latest draft of the techniques is available at: http://www.w3.org/TR/WAI-WEBCONTENT-TECHS/

[WAI-AUTOOLS]
"Authoring Tool Accessibility Guidelines", J. Treviranus, J. Richards, I. Jacobs, C. McCathieNevile, eds. The latest Working Draft of these guidelines for designing accessible authoring tools is available at: http://www.w3.org/TR/WAI-AUTOOLS/

[WAI-UA-SUPPORT]
This page documents known support by user agents (including assistive technologies) of some accessibility features listed in this document. The page is available at: http://www.w3.org/WAI/Resources/WAI-UA-Support

[WAI-USERAGENT]
"User Agent Accessibility Guidelines", J. Gunderson and I. Jacobs, eds. The latest Working Draft of these guidelines for designing accessible user agents is available at: http://www.w3.org/TR/WAI-USERAGENT/

[WCAG-ICONS]
Information about conformance icons for this document and how to use them is available at http://www.w3.org/WAI/WCAG1-Conformance.html

[UWSAG]
"The Unified Web Site Accessibility Guidelines", G. Vanderheiden, W. Chisholm, eds. The Unified Web Site Guidelines were compiled by the Trace R & D Center at the University of Wisconsin under funding from the National Institute on Disability and Rehabilitation Research (NIDRR), U.S. Dept. of Education. This document is available at: http://www.tracecenter.org/docs/html_guidelines/version8.htm

/XML]
The latest version of XML 1.0 is available at: http://www.w3.org/TR/REC-xml

http://www.w3.org/TR/WCAG10/

4/8/04
Select the areas in San Francisco (change city) where you want to stay. You'll always stay in a nationally recognized name-brand or quality independent hotel.

Improve Your Chances - Select 2 or More Areas Below:

- Cathedral Hill
- Civic Center Area
- Downtown - Union Square West
- Fisherman's Wharf
- San Francisco Airport (SFO)
- South San Francisco
- South of Market - Convention Center
- Union Square East - Embarcadero

Click on an area for more details.

If these areas don't work for you, you can select specific areas within nearby metropolitan regions. Click on a link below for more information:

San Rafael - Larkspur - Novato

Fisherman's Wharf

Beautiful views, excellent restaurants, and a constant array of activities have made this area of San Francisco popular with visitors. Pier 39, Ghirardelli Square and Alcatraz are only a few of the many attractions you will find in the Fisherman's Wharf area.

Attractions in or near this neighborhood
- Pier 39
- Ghirardelli Square
- Alcatraz
- San Francisco Art Institute

Priceline hotel participants are major national brands or the highest quality independent hotels. We screen participating hotels carefully and update our information constantly to ensure you always receive quality accommodations. We will find you a room in the very best available hotel willing to agree to your price in the shaded area shown above.