

W3C



## Table of Contents

Abstract . . . . .	.1
Status of this document . . . . .	.1
1 Introduction . . . . .	.4
1.1 How the Techniques are organized . . . . .	.4
2 Guidelines . . . . .	.5
1. Support accessible authoring practices. . . . .	.5
2. Generate standard markup. . . . .	.8
3. Support the creation of accessible content. . . . .	.9
4. Provide ways of checking and correcting inaccessible content. . . . .	25
5. Integrate accessibility solutions into the overall "look and feel". . . . .	28
6. Promote accessibility in help and documentation. . . . .	29
7. Ensure that the authoring tool is accessible to authors with disabilities. . . . .	31
3 Techniques for User Prompting . . . . .	37

# 1 Introduction

## 2 Guidelines

Guideline 1. Support accessible authoring practices.



markup

Guideline 2. Generate standard markup.





format could also be published as an HTML document. Tools that dynamically generate Web content may use HTTP content negotiation to facilitate this.

- Satisfying checkpoint 3.5 would provide much of the required functionality. Refer also to checkpoint 4.1. Refer also to checkpoint 6.2..
- **Sample:** Amaya prompts the author to provide equivalent text for `IMG` and `AREA` elements, and `CAPTION` for the `TABLE` element.
- Some Techniques listed for different languages, according to Web Content Guidelines checkpoints:

WCAG Checkpoint 1.1 Provide a text equivalent for every non-text element

[Priority 1]

Refer also to WCAG checkpoint 9.1 and WCAG checkpoint 13.168 eA Td4.45.29 0 Tdoint

WCAG Checkpoint 1.4 For any time-based multimedia presentation (e.g., a movie or animation), synchronize equivalent alternatives (e.g., captions or

WCAG

Refer also to WCAG guideline 6.

Refer also to WCAG guideline 1.  
Techniques for WCAG checkpoint 14.2  
HTML

Techniques for WCAG checkpoint 1.3

WCAG Checkpoint 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 WCAG checkpoint 1.4

WCAG Checkpoint 1.5 Until user

## HTML

Automatically include (configurable or localised) quotation marks around quotations. This will encourage authors to use the markup, and not to misuse it.

Where material appears within quote marks ask the author if this is a quotation.

WCAG Checkpoint 4.1 Clearly identify changes in the natural language of a document's text and any text equivalents (e.g., captions). [Priority 1]

Techniques for WCAG checkpoint 4.1

WCAG Checkpoint 4.2 Specify the expansion of each abbreviation or acronym in a document where it first occurs. [Priority 3]

Techniques for WCAG checkpoint 4.2

## HTML

Ask the author to provide an expansion for



## HTML

- Prompt the author to identify tables which are used as layout devices.

## HTML

See also frames.

WCAG Checkpoint 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]

Refer also to WCAG guideline 1.

Techniques for WCAG checkpoint 6.3

## HTML

- Prompt for server-side alternatives for scripts and applets
- Prompt for `noscript` content for each `script`.
- Prompt for alternative content for applets and programmatic objects (for example `object` elements which have a `code` attribute).

WCAG Checkpoint 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 WCAG checkpoint 6.4

## Applet-development

- Prompt for `on*event` attributes to include device-independent alternatives. [Priority 2]

WCAG Checkpoint 8.1 Make programmatic elements such as scripts and applets directly accessible or compatible with assistive technologies

the author for `title` text to make this clear.  
WCAG Checkpoint 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]  
12.4.

Techniques for WCAG checkpoint 10.2  
WCAG Checkpoint 10.4 Until user agents handle empty controls correctly, include default, place-holding characters in edit boxes and text areas. [Priority 3]

Techniques for WCAG checkpoint 10.4  
HTML

Prompt the author for default place-holder text. Offer the value of the `name` attribute as a default.

WCAG Checkpoint 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 WCAG checkpoint 10.5

WCAG Checkpoint 11.2 Avoid deprecated features of W3C technologies. [Priority 2]

Techniques for WCAG checkpoint 11.2

WCAG Checkpoint 11.3  
documents according to their preferences (e.g., language, content type, etc.) [Priority 3]

**Note.** Use content negotiation where possible.

Techniques for WCAG checkpoint 11.3

WCAG Checkpoint 11.4 If, after best efforts, you cannot create an accessible page, provide a link to an alternative page that uses W3C

design [Priority 3]

Techniques for WCAG checkpoint 11.2

WCAG Checkpoint 11.4

document [Priority 3]

Techniques for WCAG checkpoint 11.2

HTML 93.82 1980 93.82 c 1964.81 193.82 19522 T2483.2 19522 T2483.2

HTML

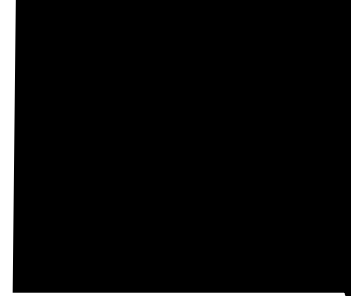
where a frameset defines a navigation frame and a welcome page, include the content of each of these frames in the `noframes`.

WCAG Checkpoint 12.3 Divide large blocks of information into more manageable groups where natural and appropriate. [Priority 2]

Refer also to WCAG guideline 3.

## HTML

provide information about document collections if lists of links are a group and should be a map.



own descriptions, and show authors the importance of description writing.  
Refer also to checkpoint 3.5.

**Sample:** Amaya does not provide any clip art or other prepackaged content.

3.4 Do not automatically generate

- The use of i 69Resource Description Framework (RDF) [RDF10] , or formats like SVG can enable a tool to maintain and use libraries of information within i 69tool and on i 69Web.
- This checkpoint is priority 3, meaning i at in itself, it does not have a critical effect on an authoring iool's likelihood of producing accessible mark-up. However, certain implementations of i is Alternative Information Management Mechanism (AIMM) [APROMPT] have i 69potential to simultaneously satisfy several higher priority checkpoints and dramatically improve i 69usability of an access aware authoring iool. In particular:
  1. The AIMM [APROMPT] should maintain a list of associations between object file names and authored responses to prompts for alternative information (per checkpoint 3.1). The alternative information may take



## Guideline 4. Provide

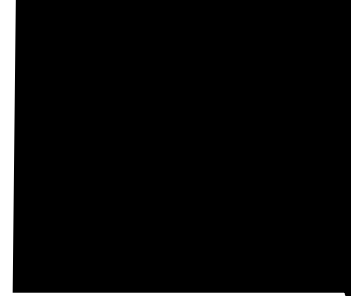
author some idea of whether a document's logical order has been correctly



•

with text links, or `AREA` elements that all have relevant "alt" attribute values.





attribute and a "longdesc" attribute wherever appropriate).

Ensure that electronic documentation complies with the Web Content

•



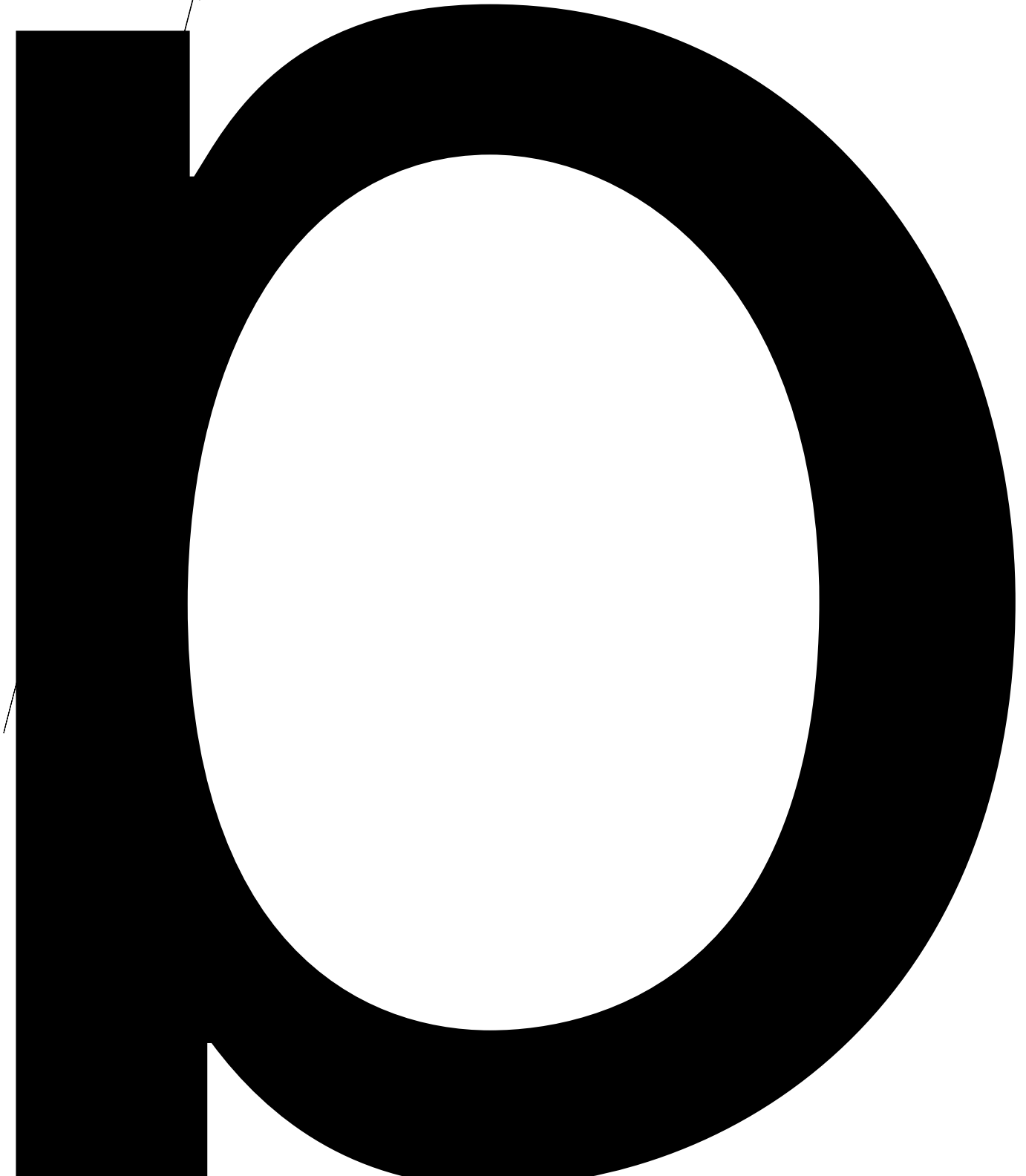




- Provide mouse access to functions where possible.

### *Icons, Graphics, Sounds*

- Provide graphical (text) equivalents for sound warnings.
- Allow sounds to be turned off.
- Provide text equivalents for images/icons.
- Use customizable (or removable) colors/patterns.



without changing the way the document is rendered when published.

- In representing the source structure of a document mark elements with text brackets rather than with purely graphic representations. For example, "</>" is regarded as a text bracket, since it is made of character elements.
- Allow the author to create audio style sheets using a graphical representation rather than an audio one (with accessible representation, of course).
- An authoring tool that offers a "rendered view" of a document, such as a browser preview mode, may provide an editing view whose presentation can be controlled independently of the rendered view.
- A WYSIWYG editor may allow an author to specify a local style sheet, that will override the "published" style of the document in the editing view.
- **Sample:** Amaya allows the author to create local style sheets, and to enable or disable each style sheet that is linked to a document.

7.3 Allow the author to edit all properties of each element and object in an accessible fashion. [Priority 1] (Checkpoint 7.3)

- An authoring tool may offer several editing views of the same document, such as a source mode that allows direct editing of all properties.
- Allow the author to individually edit each attribute of the elements in an HTML or XML document, for example, through a menu. **Note:** This must include the ability to add valid values for attributes that are not present, as well as changing current values of attributes.
- For a site management tool, allow the author to render a site map in text form (e.g., as a structured tree file).

bontrnt twhil editing



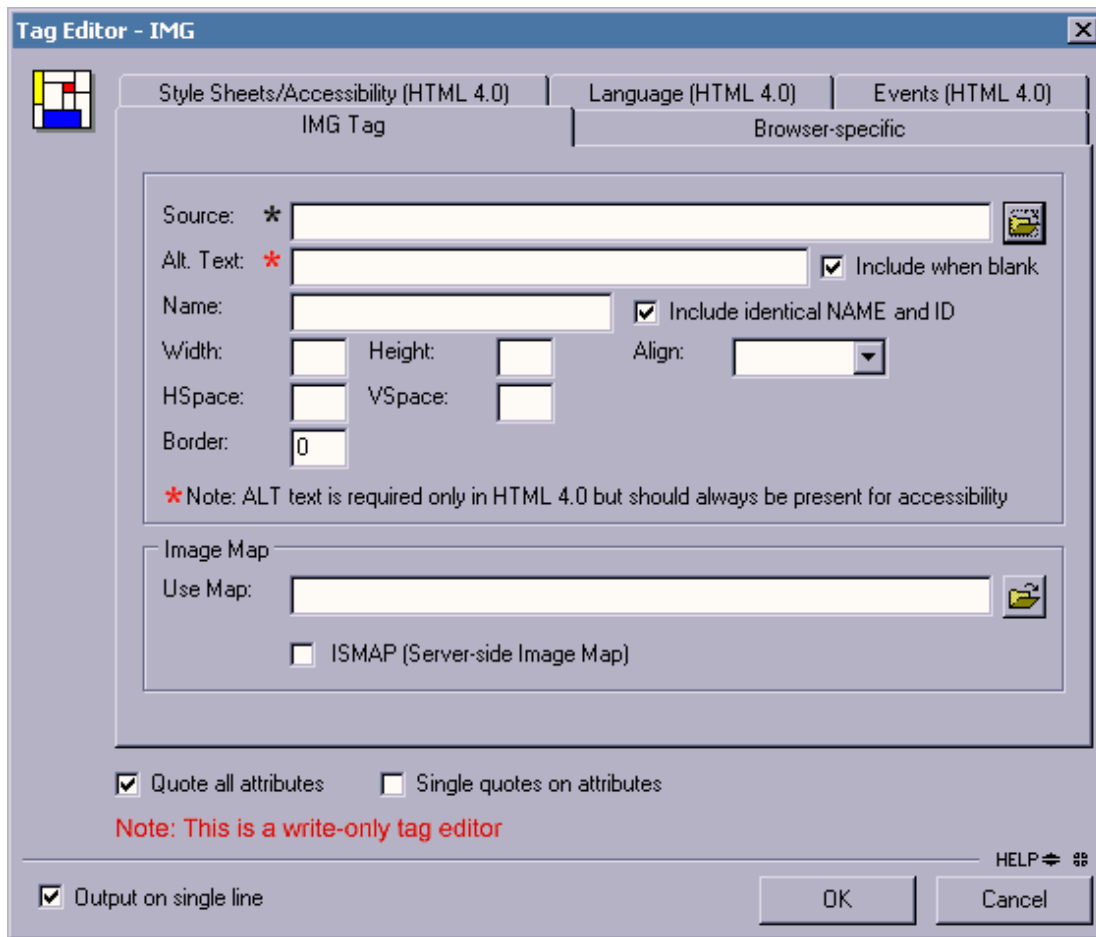
- The ability to search for a particular type of structure is useful in a structured document, structured image such as a complex SVG image, etc.
- In an image editor, the ability to select an area by properties (such as color, or closeness of color) is useful. This is common in middle range and high end image processing software.
- The ability to search a database for particular content, or to search a collection of files at once (a simple implementation of the latter is the Unix function "grep") is an important tool in managing large collections,

The importance of these concepts in the document and a perceived ambiguity of





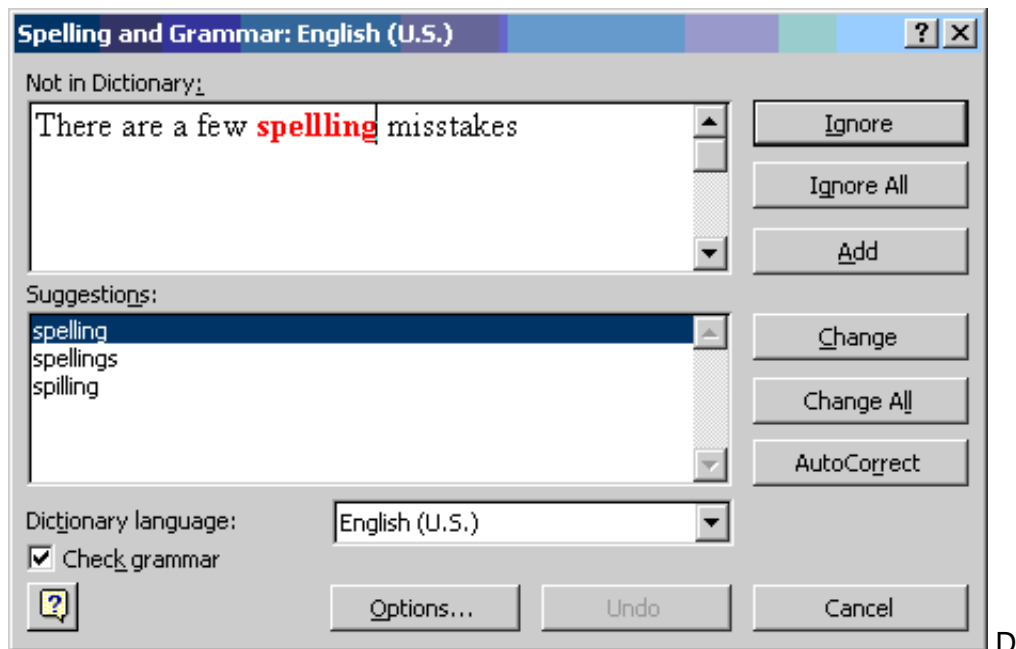




D

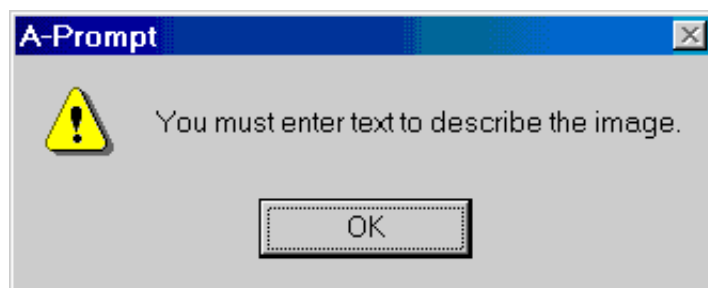
**Related Prompts:**

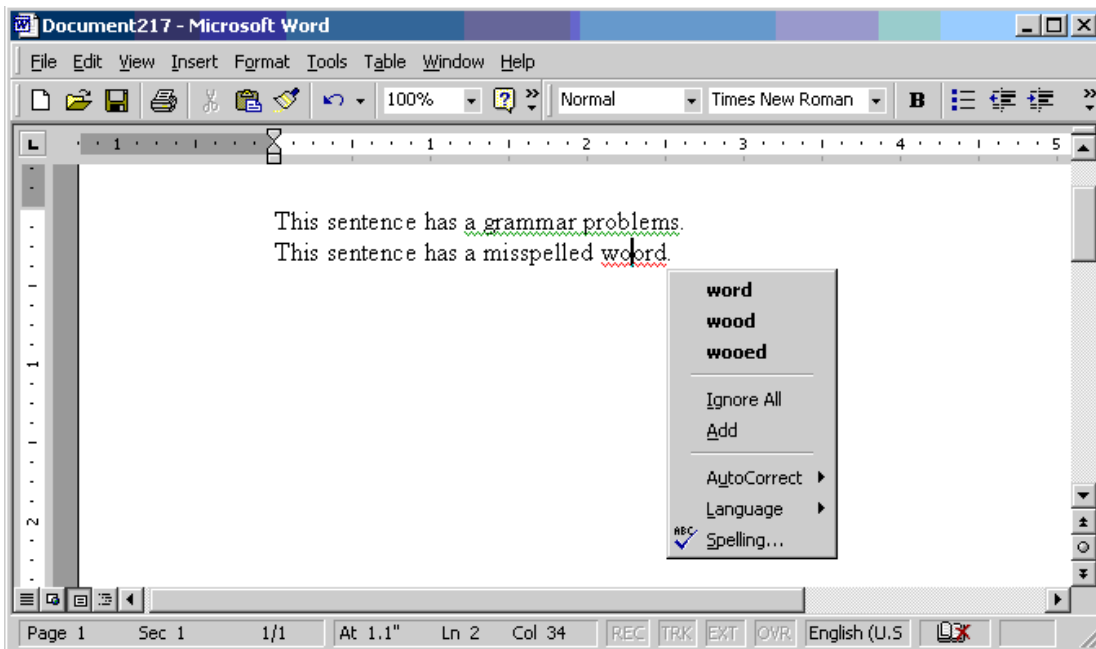




In an accessibility checker, the same is true, however the dialog template has to be somewhat more flexible since the problems can range from a missing text string for a multimedia object to missing structural information for a table to improper use of colour. In the following example, from A-Prompt, the author is prompted to add alternate text for an image as part (8 of 20) of a correction run. Notice that, like the spell checker, the prompt includes a statement of the problem (i.e. "missing alternate text for an image"), the problem instance (i.e. earthrise.gif), and suggested fixes (i.e. a suggestion from the alt-text registry, "An earth-rise as seen from the surface of the

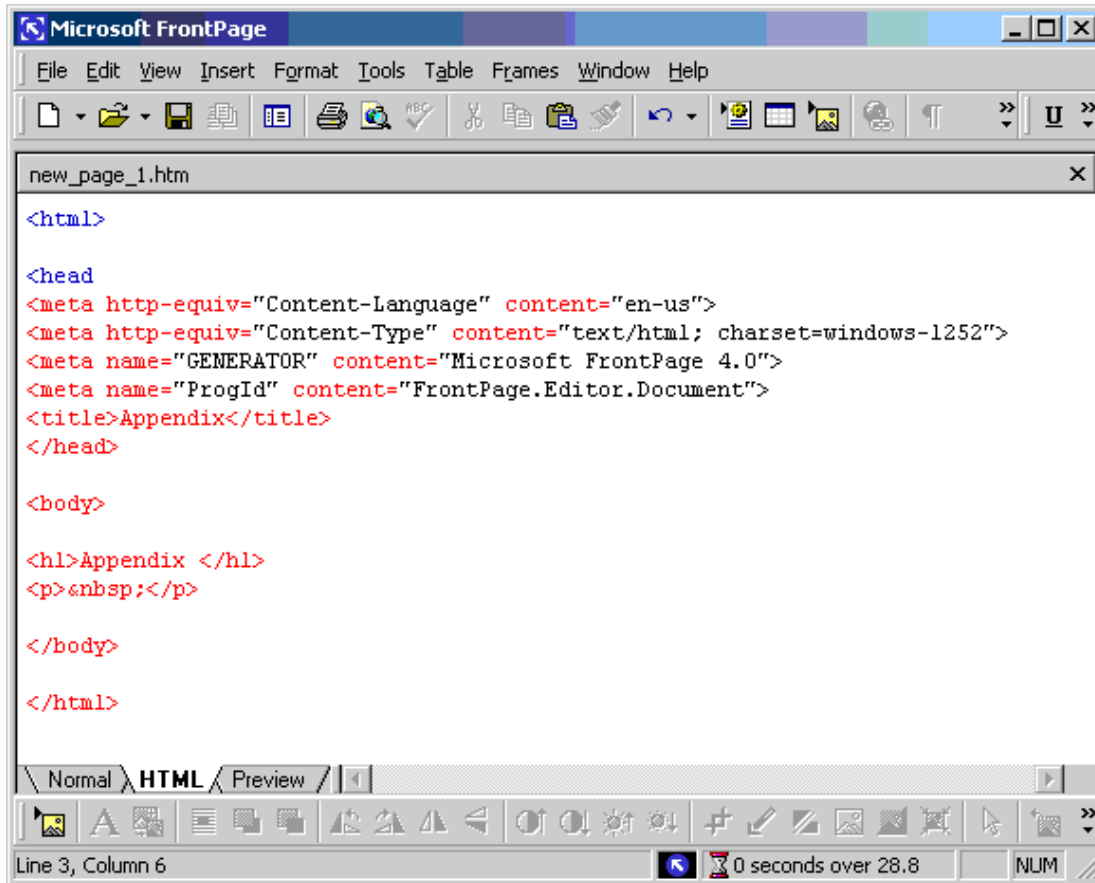






D

Another Microsoft product, FrontPage 2000, uses unintrusive alerts in its HTML editing environment to indicate syntax errors. As the author types, syntax is automatically checked. The author is allowed to make syntax errors, but the colour of the text signals that an error has been made.



The image shows a screenshot of the Microsoft FrontPage 4.0 application window. The title bar reads "Microsoft FrontPage". The menu bar includes "File", "Edit", "View", "Insert", "Format", "Tools", "Table", "Frames", "Window", and "Help". The toolbar contains various icons for file operations and editing. The main window displays the HTML source code for a file named "new\_page\_1.htm". The code is as follows:

```
<html>

<head
<meta http-equiv="Content-Language" content="en-us">
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<meta name="GENERATOR" content="Microsoft FrontPage 4.0">
<meta name="ProgId" content="FrontPage.Editor.Document">
<title>Appendix</title>
</head>

<body>

<h1>Appendix </h1>
<p>&nbsp;</p>

</body>

</html>
```

At the bottom of the window, there is a status bar showing "Line 3, Column 6", a timer indicating "0 seconds over 28.8", and a "NUM" button.

## 4 Glossary of Terms and Definitions



challenging, since author perceptions of alerts, prompts, and warnings can influence opinions of the tool and even of accessible authoring.

An ***Unintrusive Alert***

***Auditory Descri 0 cm BT -22***





to be rendered by the user agent. This may differ from the element's structural content. For example, some elements can have external data to be rendered (e.g., the `IMG`

***User-Configurable Schedule***

A "user-configurable schedule" allows the user to determine the type of prompts

## 6 References

For the latest version of any W3C specification please consult the list of W3C Technical Reports at <http://www.w3.org/TR>.

[ACCESS-AWARE]

"The Three-tions of Accessibility-Aware HTML Authoring Tools," J. Richards.

[AMAYA]

Amaya, dev

WYSIWYG-s

specification

<http://www.w>

eds., 12 May 1998. This CSS2 Recommendation is  
<http://www.w3.org/TR/1998/REC-CSS2-19980512>. The latest version of CSS2  
is available at <http://www.w3.org/TR/REC-CSS2>.

**[CSS2-ACCESS]**

"Accessibility Features of CSS," I. :acobs and J. Brewer, eds., 4 August 1999.

The latest version of Accessibility Features of CSS is available at  
<http://www.w3.org/TR/CSS-access>.

**[ED-DEPT]**

"Requirements for Accessible Software Design," US Department of Education,  
version 1.1 March 6, 1997.

**[EITACC]**

"EITACC Desktop Software standards," Electronic Information Technology  
Access Advisory (EITACC) Committee.

**[HTML-XML-VALIDATOR]**

The W3C HTML Validation Service validates HTML and XHTML markup.

**[HTML4]**

"HTML 4.01 Recommendation," D. Raggett, A. Le Hors, and I. :acobs, eds., 24  
December 1999. This HTML 4.01 Recommendation is  
<http://www.w3.org/TR/1999/REC-html401-19991224>. The latest version of  
HTML 4 is available at <http://www.w3.org/TR/html4>.

**[HTML4-ACCESS]**

"WAI Resources: HTML 4.0 Accessibility Improvements," I. :acobs, J. Brewer,  
and D. Dardailier, eds. This document describes accessibility features in HTML  
4.0.

**[IBM-ACCESS]**

"Software Accessibility," IBM Special Needs Systems.

**[CCCM]**

"The Inter-Client communication conventions manual." A protocol for



MathML 1.0

**[TRACE-REF]**

"Application Software Design Guidelines," compiled by G. Vanderheiden. A thorough reference work.

**[UAAG10]**

"User Agent Accessibility Guidelines," J. Gunderson and I. Jacobs, eds. The latest version of the User Agent Accessibility Guidelines is available at <http://www.w3.org/WAI/UA/UAAG10>.

**[UAAG10-TECHS]**

"Techniques for User Agent Accessibility Guidelines 1.0," J. Gunderson, and I. Jacobs, eds. The latest version of Techniques for User Agent Accessibility Guidelines 1.0 is available at <http://www.w3.org/TR/UAAG10-TECHS/>.

**[WAI-ER]**

The Web Accessibility Initiative Evaluation and Repair Tools Working Group tracks and develops tools that can help repair accessibility errors.

**[WCAG10]**

"Web Content Accessibility Guidelines 1.0," W. Chisholm, G. Vanderheiden, and I. Jacobs, eds., 5 May 1999. This Recommendation is <http://www.w3.org/TR/1999/WAI-WEBCONTENT-199904s5>. The latest version of the Web Content Accessibility Guidelines 1.0" is available at <http://www.w3.org/TR/WCAG10/>.

**[WCAG10-TECHS]**

"Techniques for Web Content Accessibility Guidelines 1.0," W. Chisholm, G. Vanderheiden, and I. Jacobs, eds. The latest version of Techniques for Web Content Accessibility Guidelines 1.0 is available at <http://www.w3.org/TR/WCAG10-TECHS/>.

**[WHAT-IS]**

"What is Accessible Software," James W. Thatcher, Ph.D., IBM, 1997. This paper gives a short example-based introduction to the difference between software that is accessible, and software that can be used by some assistive technologies.

**[XHTML10]**

"XHTML<sup>(TM)</sup>

Level Double-A conformance icon, W3C-WAI Web Content Accessibility Guidelines 1.0