Version: 14 March 2013, 12:07 hrs E Hansen comments, mostly regarding conformance.

[[contents](http://www.w3.org/WAI/UA/UAAG20/#toc)] [[implementing](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/)]

[W3C](http://www.w3.org/)

User Agent Accessibility Guidelines (UAAG) 2.0

W3C Editors' Draft 22 February 2013

**This version:**

<http://www.w3.org/WAI/UA/2013/ED-UAAG20-20130222/>

**Latest version:**

<http://www.w3.org/TR/UAAG20/>

**Previous version:**

[http://www.w3.org/WAI/UA/2013/ED-UAAG20-20130108](http://www.w3.org/WAI/UA/2012/ED-UAAG20-20121213/)

**Editors:**

James Allan, Texas School for the Blind and Visually Impaired

Kelly Ford, Microsoft

Kim Patch, Redstart Systems

Jeanne Spellman, W3C/Web Accessibility Initiative

**Previous Editors:**

Jan Richards, Inclusive Design Institute, OCAD University

[Copyright](http://www.w3.org/Consortium/Legal/ipr-notice#Copyright) © 2012 [W3C](http://www.w3.org/)® ([MIT](http://www.csail.mit.edu/), [ERCIM](http://www.ercim.eu/), [Keio](http://www.keio.ac.jp/)), All Rights Reserved. W3C [liability](http://www.w3.org/Consortium/Legal/ipr-notice#Legal_Disclaimer), [trademark](http://www.w3.org/Consortium/Legal/ipr-notice#W3C_Trademarks) and [document use](http://www.w3.org/Consortium/Legal/copyright-documents) rules apply.

Abstract

UAAG 2.0 provides guidelines for designing [user agents](http://www.w3.org/WAI/UA/UAAG20/#def-user-agent) that lower barriers to Web accessibility for people with disabilities. User agents include browsers and other types of software that retrieve and render [Web content](http://www.w3.org/WAI/UA/UAAG20/#def-content). A user agent that [conforms](http://www.w3.org/WAI/UA/UAAG20/#conformance) to these guidelines will promote accessibility through its own user interface and through other internal facilities, including its ability to communicate with other technologies (especially [assistive technologies](http://www.w3.org/WAI/UA/UAAG20/#def-assistive-technology)). Furthermore, all users, not just users with disabilities, should find conforming user agents to be more usable.

In addition to helping developers of browsers and media players, UAAG 2.0 will benefit developers of assistive technologies because it explains what types of information and control an assistive technology may expect from a conforming user agent. Technologies not addressed directly by UAAG 2.0 (e.g. technologies for braille rendering) will be essential to ensuring Web access for some users with disabilities.

The "User Agent Accessibility Guidelines 2.0" (UAAG 2.0) is part of a series of accessibility guidelines published by the W3C [Web Accessibility Initiative](http://www.w3.org/WAI/) (WAI).

Status of this document

May be Superseded

*This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the*[*W3C technical reports index*](http://www.w3.org/TR/)*at http://www.w3.org/TR/.*

Editor's Draft of UAAG 2.0

**This document is the internal working draft used by the UAWG and is updated continuously and without notice. This document has no formal standing within W3C. Please consult the**[**group's home page**](http://www.w3.org/WAI/UA/)**and the**[**W3C technical reports index**](http://www.w3.org/TR/)**for information about the latest publications by this group.**

Web Accessibility Initiative

This document has been produced as part of the W3C [Web Accessibility Initiative](http://www.w3.org/WAI/) (WAI). The goals of the User Agent Working Group (UAWG) are discussed in the [Working Group charter](http://www.w3.org/WAI/UA/2010/uawg_charter.html). The UAWG is part of the [WAI Technical Activity](http://www.w3.org/WAI/Technical/Activity).

No Endorsement

Publication as a Working Draft does not imply endorsement by the W3C Membership. This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this document as other than work in progress.

Patents

This document was produced by a group operating under the [5 February 2004 W3C Patent Policy](http://www.w3.org/Consortium/Patent-Policy-20040205/). W3C maintains a [public list of any patent disclosures](http://www.w3.org/2004/01/pp-impl/36791/status) made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains [Essential Claim(s)](http://www.w3.org/Consortium/Patent-Policy-20040205/#def-essential) must disclose the information in accordance with [section 6 of the W3C Patent Policy](http://www.w3.org/Consortium/Patent-Policy-20040205/#sec-Disclosure).

Table of Contents

* [Abstract](http://www.w3.org/WAI/UA/UAAG20/#abstract)
* [Status of this document](http://www.w3.org/WAI/UA/UAAG20/#status)
* [Introduction](http://www.w3.org/WAI/UA/UAAG20/#introduction)
  + [Overview](http://www.w3.org/WAI/UA/UAAG20/#conceptual_overview)
  + [UAAG 2.0 Layers of Guidance](http://www.w3.org/WAI/UA/UAAG20/#layers_guide)
  + [UAAG 2.0 Supporting Documents](http://www.w3.org/WAI/UA/UAAG20/#supporting_docs)
  + [Components of Web Accessibility](http://www.w3.org/WAI/UA/UAAG20/#intro-components)
  + [Levels of Conformance](http://www.w3.org/WAI/UA/UAAG20/#intro-conf-levels)
  + [Definition of User Agent](http://www.w3.org/WAI/UA/UAAG20/#intro-def-ua)
  + [Modality Independence Principle](http://www.w3.org/WAI/UA/UAAG20/#intro-modality-independence)
  + [Relationship with Authoring Tool Accessibility Guidelines 2.0 (ATAG)](http://www.w3.org/WAI/UA/UAAG20/#intro-atag)
* [UAAG 2.0 Guidelines](http://www.w3.org/WAI/UA/UAAG20/#Guidelines)
  + [PRINCIPLE 1](http://www.w3.org/WAI/UA/UAAG20/#principle-perceivable): Perceivable
    - [Guideline 1.1](http://www.w3.org/WAI/UA/UAAG20/#gl-access-alternative-content): Alternative content
    - [Guideline 1.2](http://www.w3.org/WAI/UA/UAAG20/#gl-missing-alt): Missing content
    - [Guideline 1.3](http://www.w3.org/WAI/UA/UAAG20/#gl-interaction-highlight): Highlighting
    - [Guideline 1.4](http://www.w3.org/WAI/UA/UAAG20/#gl-text-config): Text configuration
    - [Guideline 1.5](http://www.w3.org/WAI/UA/UAAG20/#gl-volume-config): Volume configuration
    - [Guideline 1.6](http://www.w3.org/WAI/UA/UAAG20/#gl-speech-config): Synthesized speech configuration
    - [Guideline 1.7](http://www.w3.org/WAI/UA/UAAG20/#gl-style-sheets-config): User Style sheets
    - [Guideline 1.8](http://www.w3.org/WAI/UA/UAAG20/#gl-viewport-orient): Orientaion in Viewports
    - [Guideline 1.9](http://www.w3.org/WAI/UA/UAAG20/#gl-alternative-views): Alternative Views
    - [Guideline 1.10](http://www.w3.org/WAI/UA/UAAG20/#gl-info-link): Element Information
  + [PRINCIPLE 2](http://www.w3.org/WAI/UA/UAAG20/#principle-operable). Operable
    - [Guideline 2.1](http://www.w3.org/WAI/UA/UAAG20/#gl-keyboard-access): Keyboard access
    - [Guideline 2.2](http://www.w3.org/WAI/UA/UAAG20/#gl-sequential-navigation): Sequential navigation
    - [Guideline 2.3](http://www.w3.org/WAI/UA/UAAG20/#gl-direct-navigation-and-activation): Direct navigation and activation
    - [Guideline 2.4](http://www.w3.org/WAI/UA/UAAG20/#gl-search-text): Search
    - [Guideline 2.5](http://www.w3.org/WAI/UA/UAAG20/#gl-nav-structure): Structured navigation
    - [Guideline 2.6](http://www.w3.org/WAI/UA/UAAG20/#gl-device-independent-handlers): Event handlers
    - [Guideline 2.7](http://www.w3.org/WAI/UA/UAAG20/#gl-store-prefs): Preference settings
    - [Guideline 2.8](http://www.w3.org/WAI/UA/UAAG20/#gl-configure-controls): Toolbar configuration
    - [Guideline 2.9](http://www.w3.org/WAI/UA/UAAG20/#gl-time-independent): Time-independence
    - [Guideline 2.10](http://www.w3.org/WAI/UA/UAAG20/#gl-prevent-flash): Flashing
    - [Guideline 2.11](http://www.w3.org/WAI/UA/UAAG20/#gl-control-inaccessible-content): Media
    - [Guideline 2.12](http://www.w3.org/WAI/UA/UAAG20/#gl-other-devices): Other Input Devices
  + [PRINCIPLE 3](http://www.w3.org/WAI/UA/UAAG20/#principle-understandable): Understandable
    - [Guideline 3.1](http://www.w3.org/WAI/UA/UAAG20/#gl-avoid-unnecessary-messages): Unnecessary messages
    - [Guideline 3.2](http://www.w3.org/WAI/UA/UAAG20/#gl-avoid-mistakes): Mistakes
    - [Guideline 3.3](http://www.w3.org/WAI/UA/UAAG20/#gl-doc-access-features): Documentation
    - [Guideline 3.4](http://www.w3.org/WAI/UA/UAAG20/#gl-predictable-operation): Predictable
  + [PRINCIPLE 4](http://www.w3.org/WAI/UA/UAAG20/#principle-AT-access). Programmatic access
    - [Guideline 4.1](http://www.w3.org/WAI/UA/UAAG20/#gl-AT-access): Assistive technology
  + [PRINCIPLE 5](http://www.w3.org/WAI/UA/UAAG20/#principle-follow-specs): Specifications and conventions
    - [Guideline 5.1](http://www.w3.org/WAI/UA/UAAG20/#gl-obs-env-conventions): Follow specifications
* [Conformance](http://www.w3.org/WAI/UA/UAAG20/#conformance)
* Appendix A: [Glossary](http://www.w3.org/WAI/UA/UAAG20/#glossary)
* Appendix B: [How to refer to UAAG 2.0 from other documents](http://www.w3.org/WAI/UA/UAAG20/#ref-this-doc)
* Appendix C: [References](http://www.w3.org/WAI/UA/UAAG20/#references)
* Appendix D: [Acknowledgements](http://www.w3.org/WAI/UA/UAAG20/#acknowledgments)
* Appendix E: Checklist
* Appendix F: Comparison of UAAG 1.0 guidelines to UAAG 2.0

Introduction

This section is [informative](http://www.w3.org/WAI/UA/UAAG20/#def-informative).

A user agent is any software that retrieves and presents Web content for end users. User agents include Web browsers, media players, and plug-ins that help in retrieving, rendering and interacting with Web content. UAAG 2.0 specifies requirements for [user agent](http://www.w3.org/WAI/UA/UAAG20/#def-user-agent) developers that will lower barriers to accessibility.

Overview

Accessibility involves a wide range of disabilities. These include visual, auditory, physical, speech, cognitive, language, learning, neurological disabilities, and disabilities related to ageing. The goal of UAAG 2.0 is to ensure that all users, including users with disabilities, have control over their environment for accessing the Web. Key methods for achieving that goal include:

* configurability
* device independence
* interoperability
* direct support for both graphical and auditory output
* optional self-pacing
* adherence to published conventions

Some users have more than one disability, and the needs of different disabilities may conflict. Thus, many UAAG 2.0 requirements use configuration to ensure that a functionality designed to improve accessibility for one user does not interfere with accessibility for another. UAAG 2.0 prefers configuration requirements rather than requirements for default settings, because a default user agent setting may be useful for one user but interfere with accessibility for another. For example, a feature required by UAAG 2.0 may be ineffective or cause content to be less accessible, making it imperative that the user be able to turn off the feature. To avoid overwhelming users with an abundance of configuration options, UAAG 2.0 includes requirements that promote documentation and ease of configuration.

Although author preferences are important, UAAG 2.0 includes requirements to override certain author preferences when the user would not otherwise be able to access that content.

Some UAAG 2.0 requirements may have security implications, such as communicating through APIs, or allowing programmatic read and write access to content and [user interface control](http://www.w3.org/WAI/UA/UAAG20/#def-ui-control). UAAG 2.0 assumes that features required by UAAG 2.0 will be built on top of an underlying security architecture. Consequently, UAAG 2.0 grants no conformance exemptions based on security issues, unless permitted explicitly in a success criterion.

The UAWG expects that software that satisfies the requirements of UAAG 2.0 will be more flexible, manageable, extensible, and beneficial for all users.

UAAG 2.0 Layers of Guidance

In order to meet the needs of different audiences using UAAG, several layers of guidance are provided, including overall *principles*, general *guidelines*, testable *success criteria*, and explanatory *intent*, *examples* and *resource* links.

* **Principles** - At the top are five principles that provide the foundation for accessible user agents. Principles 1, 2, and 3 parallel the Web Content Accessibility Guidelines (WCAG) 2.0: *t*o make the user agent *perceivable, operable,*and*understandable*. Principles 4 and 5 are specific to user agents: facilitate*programmatic access* and comply with*specifications and conventions*.
* **Guidelines** - Under the principles are guidelines. The guidelines are goals authors should work toward in order to make user agents more accessible to users with disabilities. The guidelines are not testable, but provide the framework and overall objectives to help authors understand the success criteria and better implement them.
* **Success Criteria** - For each guideline, testable success criteria are provided to allow UAAG 2.0 to be used where requirements and conformance testing are necessary, such as design specification, purchasing, regulation, and contractual agreements. Three levels of conformance meet the needs of different groups and different situations: A (lowest), AA, and AAA (highest). Additional information on UAAG levels can be found in the section on [Conformance](http://www.w3.org/WAI/UA/UAAG20/#conformance).

The principles, guidelines, and success criteria work together to provide guidance on how to make user agents more accessible. Developers are encouraged to use them in order to best address the needs of the widest possible range of users.

Even user agents that conform at the highest level (AAA) may not be accessible to individuals with all types, degrees, or combinations of disability, particularly in the cognitive, language, and learning areas. Developers are encouraged to seek out current best practice to ensure that user agents are as accessible as possible.

UAAG 2.0 Supporting Documents

**A separate document, entitled *Implementing User Agent Accessibility Guidelines 2.0*** (referred to as the "Implementing document" from here on) provides explanations and examples of how each success criteria might be satisfied. It also includes references to other accessibility resources (such as platform-specific software accessibility guidelines) that provide additional information on how a user agent may satisfy each success criteria. The examples in the Implementing document are [informative](http://www.w3.org/WAI/UA/UAAG20/#def-informative) only. Other strategies may be used or required to satisfy the success criteria. The UAWG expects to update the Implementing document more frequently than the current guidelines. Developers, W3C Working Groups, users, and others are encouraged to contribute examples and resources.

Components of Web Accessibility

Web accessibility depends on accessible user agents and accessible content. Accessible content availability is greatly influenced by the accessibility of the authoring tool. For an overview of how these components of Web development and interaction work together, see

* [**Essential Components of Web Accessibility**](http://www.w3.org/WAI/intro/components)
* [**Web Content Accessibility Guidelines (WCAG) Overview**](http://www.w3.org/WAI/intro/wcag.php)
* [**Authoring Tool Accessibility Guidelines (ATAG) Overview**](http://www.w3.org/WAI/intro/atag.php)

Levels of Conformance

User Agents may claim conformance to UAAG 2.0 at one of three conformance levels. The level achieved depends on the level of success criteria satisfied. The conformance levels are

1. **UAAG 2.0 Conformance at Level "A"**  
   The user agent satisfies *all* of the Level A success criteria.
2. **UAAG 2.0 Conformance at Level "Double-A"**  
   The user agent satisfies *all* of the Level A and Level AA success criteria.
3. **UAAG 2.0 Conformance at Level "Triple-A"**   
   The user agent satisfies *all* of the success criteria.

Definition of ***User Agent***

A user agent is any software that retrieves, renders and facilitates end-user interaction with Web content.

A “classic” example of a user agent is a browser. A media player, which only performs these functions for time-based media, is also user agent. Web applications and some mobile apps that render web content are also user agents.

For specific advice in determining if software is a user agent, see [What Qualifies as a User Agent](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#intro-def-useragent) (Implementing UAAG 2.0). User agents may also include authoring tool features: see [Relationship to the Authoring Tool Accessibility Guidelines (ATAG) 2.0](http://www.w3.org/WAI/UA/UAAG20/#intro-atag). For information on the difference between web applications and content see [Relationship to the Web Content Accessibility Guidelines](http://www.w3.org/WAI/UA/UAAG20/#intro-wcag).

Modality Independence Principle

Users interacting with a web browser may do so using one or more input methods including keyboard, mouse, speech, touch, and gesture. It's critical that each user be free to use whatever input method or combination of methods works best for a given situation. Therefore every potential user task must be accessible via modality independent controls that any input technology can access.

For instance, if a user can't use or doesn't have access to a mouse, but can use and access a keyboard, the keyboard can call a modality independent control to activate an OnMouseOver event.

Relationship to the Authoring Tool Accessibility Guidelines (ATAG) 2.0

While it is common to think of user agents retrieving and rendering web content for one group of people (end-users) that was previously authored by another group (authors), user agents are also frequently involved with the process of authoring content.

For these cases, it is important for user agent developers to consider the application of another W3C-WAI Recommendation, the Authoring Tool Accessibility Guidelines (ATAG). ATAG (currently 2.0 is in draft) provides guidance to the developers of tools regarding the accessibility of authoring interfaces to authors (ATAG 2.0 Part A) and ways in which all authors can be supported in producing accessible web content (ATAG 2.0 Part B).

For more information on the role of user agents in web authoring see [Implementing UAAG 2.0](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#intro-atag-role)

Relationship to the Web Content Accessibility Guidelines (WCAG) 2.0

The W3C recommendation, Web Content Accessibility Guidelines (WCAG), applies to all web content; UAAG provides additional advice on the application user interface.

Some user agents are used to package web content into non-web-based applications, especially on mobile platforms. If the finished application is used to retrieve, render, and facilitate end-user interaction with Web content of the end-users choosing, then the application should be considered a stand-alone user agent. If the finished application only renders a constrained set of content specified by the developer, then the application might not be considered a user agent. In both cases, the WCAG 2.0 Guidelines apply.

UAAG 2.0 Guidelines

The success criteria and applicability notes in this section are [normative](http://www.w3.org/WAI/UA/UAAG20/#def-normative). Guideline summaries are informative.

PRINCIPLE 1 - Ensure that the user interface and rendered content are perceivable

Guideline 1.1 - Provide access to alternative content. [[Implementing 1.1](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-access-alternative-content)]

**Summary**: The user can choose to render any type of alternative content available. (1.1.1). The user can also choose at least one alternative such as alt text to be always displayed (1.1.2), but it's recommended that users also be able to specify a cascade (1.1.4), such as alt text if it's there, otherwise longdesc, otherwise filename, etc. It's recommended that the user can configure the caption text and that text or sign language alternative cannot obscure the video or the controls (1.1.3). The user can configure the size and position of media alternatives (1.1.5).

**1.1.1 Render Alternative Content** [was 1.1.3]**:**

For any content element, the user can choose to render any types of [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) [alternative content](http://www.w3.org/WAI/UA/UAAG20/#def-conditional-content) that are present. (Level A) ## DONE 28 March 2012

**Note**: It is recommended that the user agent allow the user to choose whether the alternative content replaces or supplements the original content element.

**1.1.x Replace Non-Text Content:**

The user can have all recognized non-text content replaced by alternative content or placeholders.(Level A)  
## DONE 13 December 2012  
  
**Note**: At Level A, the user agent can choose which type of alternative content or placeholder is used to replace the non-text content. However, at Level AA success criterion 1.1.2 will require that the user be allowed to specify one format or placeholder that will be used, and at Level AAA success criterion 1.1.4 will require the user be allowed to specify a cascade order of formats to be rendered.

**1.1.2 Configurable Alternative Content Defaults** [was 1.1.1]**:**

For each type of non-text content, the user can specify a type of[alternative content](http://www.w3.org/WAI/UA/UAAG20/#def-conditional-content) that, if present, will be rendered by default. (Level AA) ## DONE 28 March 2012

**1.1.3 Display of Time-Based Media Alternatives:**

For [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) on-screen alternatives for time-based media (e.g. captions, sign language video), the following are all true: (Level AA)## DONE 2 August 2012

1. Do not obscure primary media: The user can specify that displaying media alternatives doesn't obscure the primary time-based media; and
2. Do not obscure controls: The user can specify that the displaying media alternatives doesn't obscure recognized controls for the primary time-based media; and
3. Configurable text: The user can configure recognized text within media alternatives (e.g. captions) in conformance with [1.4.1](http://www.w3.org/WAI/UA/UAAG20/#sc-141).

Note: Depending on the screen area available, the display of the primary time-based media may need to be reduced in size to meet this requirement.

**~~1.1.3 [old] Indicate Unrendered Alternative Content:~~**

## DONE 28 March 2012 ## Deleted 4 June 2012

**1.1.4 Default Rendering of Alternative Content (Enhanced):**

For each type of non-text content, the user can specify the cascade order in which to render different types of [alternative content](http://www.w3.org/WAI/UA/UAAG20/#def-conditional-content) when preferred types are not present. (Level AAA) ## DONE 28 March 2012

**1.1.5 Size and Position of Time-Based Media Alternatives:**

The user can configure [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) on-screen alternatives for time-based media (e.g. captions, sign language video) as follows: (Level AAA)

1. The user can *resize* the media alternatives up to the size of the user agent's viewport.
2. The user can *reposition* the media alternatives to at least above, below, to the right, to the left, and overlapping the primary time-based media.

Note 1: Depending on the screen area available, the display of the primary time-based media may need to be reduced in size or hidden to meet this requirement.

Note 2: Implementation may involve displaying media alternatives in a separate window, but this is not required.

## DONE 2 August 2012 and IER on 16 August

Guideline 1.2 - Repair missing content. [[Implementing 1.2](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-missing-alt)]

**Summary**: The user can request useful alternative content when the author fails to provide it. For example, showing metadata in place of missing or empty (1.2.1) alt text. The user can ask the browser to predict missing structural information, such as field labels, table headings or section headings (1.2.2).

**1.2.1 Support Repair by Assistive Technologies:**

If [text alternatives for non-text content](http://www.w3.org/WAI/UA/UAAG20/#def-conditional-content) are missing or empty then both of the following are true: (Level AA)

1. the[user agent](http://www.w3.org/WAI/UA/UAAG20/#def-user-agent) does not attempt to [repair](http://www.w3.org/WAI/UA/UAAG20/#def-repair-text) the text alternatives with text values that are also available to assistive technologies.
2. the user agent makes metadata related to the non-text content available programmatically (and not via fields reserved for text alternatives).

## DONE 19 April

**1.2.2 Repair Missing Structure:**

 The user can specify whether or not the [user agent](http://www.w3.org/WAI/UA/UAAG20/#def-user-agent) should attempt to insert the following types of structural markup on the basis of author-specified presentation attributes (e.g.. position and appearance): (Level AAA) ## DONE 19 April & IER done on 2 August 2012

1. Labels
2. Headers (e.g. heading markup, table headers)

Guideline 1.3 - Provide highlighting for selection, keyboard focus, enabled elements, visited links.[[Implementing 1.3](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-interaction-highlight)]

**Summary**: The user can visually distinguish selected, focused, and enabled items, and recently visited links (1.3.1), with a choice of highighting options that at least include foreground and background colors, and border color and thickness (1.3.2).

**1.3.1 Highlighted Items:**

The user can specify that the following classes be [highlighted](http://www.w3.org/WAI/UA/UAAG20/#def-highlight) so that each is uniquely distinguished: (Level A)## DONE 5 April 2012## DONE 4 June 2012

1. selection
2. [active keyboard focus](http://www.w3.org/WAI/UA/UAAG20/#def-active-input-focus) (indicated by focus cursors and/or text cursors)
3. recognized enabled input elements (distinguished from disabled elements)
4. elements with alternative content
5. recently visited links

**1.3.2 Highlighting Options:**

When highlighting classes specified by [1.3.1 Highlighted Items](http://www.w3.org/WAI/UA/UAAG20/#sc-131), the user can specify highlighting options that include at least: (Level AA) ## DONE 12 April 2012

1. **foreground colors**,
2. **background colors**, and
3. **borders** (configurable color, style, and thickness)

Guideline 1.4 - Provide text configuration. [[Implementing 1.4](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-text-config)]

**Summary**: The user can control text font, color, and size (1.4.1), including whether all text should be the shown the same size (1.4.2).

**1.4.1 Configure Rendered Text:**

The user can globally set any or all of the following characteristics of visually [rendered text](http://www.w3.org/WAI/UA/UAAG20/#def-rendered-text) content,[overriding](http://www.w3.org/WAI/UA/UAAG20/#def-override) any specified by the [author](http://www.w3.org/WAI/UA/UAAG20/#def-author) or [user agent](http://www.w3.org/WAI/UA/UAAG20/#def-user-agent) defaults: (Level A) ## DONE TPAC

1. **text scale** (the general size of text)
2. **font family**
3. **text color** (foreground and background)
4. **line spacing**
5. **character spacing**

**1.4.2 Preserving Size Distinctions:**

 The user can specify whether or not distinctions in the size of [rendered text](http://www.w3.org/WAI/UA/UAAG20/#def-rendered-text) are preserved when that text is rescaled (e.g. headers continue to be larger than body text). (Level A) ## DONE TPAC

Guideline 1.5 - Provide volume configuration. [[Implementing 1.5](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-volume-config)]

**Summary**: The user can adjust the volume of each audio track relative to the global volume level (1.5.1).

**1.5.1 Global Volume:**

The user can independently adjust the volume of all [audio tracks](http://www.w3.org/WAI/UA/UAAG20/#def-audio-track), relative to the global volume level set through[operating environment](http://www.w3.org/WAI/UA/UAAG20/#def-operating-environment) mechanisms. (Level A) ## DONE 3 May 2012

Guideline 1.6 - Provide synthesized speech configuration. [[Implementing 1.6](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-speech-config)]

**Summary**: If synthesized speech is produced, the user can specify speech rate and volume (1.6.1), pitch and pitch range (1.6.2), and synthesizer speech characteristics like emphasis (1.6.3) and features like spelling (1.6.4).

**1.6.1 Speech Rate, Volume, and Voice:**

If synthesized speech is produced, the user can specify the following: (Level A) ## DONE TPAC

1. **speech rate**,
2. **speech volume** (independently of other sources of [audio](http://www.w3.org/WAI/UA/UAAG20/#def-audio)), and
3. **voice**, when more than one voice option is available

**1.6.2 Speech Pitch and Range:**

If synthesized speech is produced, the user can specify the following if offered by the speech synthesizer: (Level AA)## DONE TPAC. Updated 4 October 2012

1. **pitch** (average frequency of the speaking voice), and
2. **pitch range** (variation in average frequency)

Note: Because the technical implementations of text to speech engines vary (e.g., formant-based synthesis or concatenative synthesis), a specific engine may not support varying pitch or pitch range. A user agent will expose the availability of pitch and pitch range control if the currently selected or installed text to speech engine offers this capability.

**1.6.3 Advanced Speech Characteristics:**

The user can adjust all of the speech characteristics offered by the speech synthesizer.(Level AAA)## DONE TPAC

**1.6.4 Synthesized Speech Features:**

If synthesized speech is produced, the following features are provided: (Level AA) ## DONE TPAC

1. **user-defined extensions** to the synthesized speech dictionary,
2. **"spell-out"**, where text is spelled one character at a time, or according to language-dependent pronunciation rules,
3. **at least two ways of speaking numerals:**spoken as individual digits and punctuation (e.g. "one two zero three point five" for 1203.5 or "one comma two zero three point five" for 1,203.5), and spoken as full numbers are spoken (e.g. "one thousand, two hundred and three point five" for 1203.5),
4. **at least two ways of speaking punctuation:** spoken literally, and with punctuation understood from natural pauses.

Guideline 1.7 - Enable Configuration of User Stylesheets. [[Implementing 1.7](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-style-sheets-config)] ## DONE 4 June 2012

**Summary**: The user agent shall support user stylesheets (1.7.1) and the user can choose which if any user-supplied (1.7.2) and author-supplied (1.7.3) stylesheets to use. The user agent will allow users to save user stylesheets (1.7.4).

**1.7.1 Support User Stylesheets:**

If the user agent supports a mechanism for authors to supply [stylesheets](http://www.w3.org/WAI/UA/UAAG20/#def-style-profile), the user agent also provides a mechanism for users to supply stylesheets. (Level A)

**1.7.2 Apply User Stylesheets:**

If [user style sheets](http://www.w3.org/WAI/UA/UAAG20/#def-user-stylesheet) are supported, then the user can enable or disable user stylesheets for: (Level A)## DONE 9 August 2012

1. all pages on specified websites, or
2. all pages

**1.7.3 Author Style Sheets:**

If the user agent supports a mechanism for authors to supply [stylesheets](http://www.w3.org/WAI/UA/UAAG20/#def-style-profile), the user can disable the use of[author style sheets](http://www.w3.org/WAI/UA/UAAG20/#def-author-stylesheet) on the current page. (Level A)## DONE 9 August 2012

**1.7.4 Save Copies of Stylesheets:**

The user can save copies of the stylesheets referenced by the current page, in order to edit and load the copies as user [stylesheets](http://www.w3.org/WAI/UA/UAAG20/#def-stylesheet). (Level AA)

Guideline 1.8 - Help users to use and orient within windows and viewports. [[Implementing 1.8](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-viewport-orient)]

**Summary**: The user agent provides programmatic and visual cues to keep the user oriented. These include highlighting the viewport (1.8.1), keeping the focus within the viewport (1.8.2 & 1.8.7), resizing the viewport (1.8.3), providing scrollbar(s) that identify when content is outside the visible region (1.8.4) and which portion is visible (1.8.5), changing the size of graphical content with zoom (1.8.6 & 1.8.12), restoring the focus and point of regard when the user returns to a previously viewed page (1.8.8). Users can set a preference whether new windows or tabs open automatically (1.8.9) or get focus automatically (1.8.10). Additionally, the user can specify that all view ports have the same user interface elements (1.8.11), if and how new viewports open (1.8.9), and whether the new window automatically gets focus (1.8.10). The user can mark items in a webpage and use shortcuts to navigate back to marked items. (1.8.13).

**1.8.1 Highlight Viewport:**

The [viewport](http://www.w3.org/WAI/UA/UAAG20/#def-viewport) with the [input focus](http://www.w3.org/WAI/UA/UAAG20/#def-input-focus) is highlighted and the user can customize attributes of the [highlighting](http://www.w3.org/WAI/UA/UAAG20/#def-highlight)mechanism (e.g. shape, size, stroke width, color, blink rate). The viewport can include nested viewports and containers. (Level A) ## DONE TPAC - edited by K & J 30 August 2012.

**1.8.2 Move Viewport to Selection and Focus:**

When a viewport's [selection](http://www.w3.org/WAI/UA/UAAG20/#def-selection) or [input focus](http://www.w3.org/WAI/UA/UAAG20/#def-input-focus) changes, the [viewport](http://www.w3.org/WAI/UA/UAAG20/#def-viewport)'s content moves as necessary to ensure that the new selection or input focus location is at least partially in the visible portion of the viewport. (Level A)## DONE TPAC

**1.8.3 Resize Viewport:**

The user can resize graphical [viewports](http://www.w3.org/WAI/UA/UAAG20/#def-viewport) within the limits of the display, [overriding](http://www.w3.org/WAI/UA/UAAG20/#def-override) any values specified by the[author](http://www.w3.org/WAI/UA/UAAG20/#def-author). (Level A) ## DONE 26 April

**1.8.4 Viewport Scrollbars:**

 When the rendered content extends beyond the viewport dimensions, users can have graphical viewports include scrollbars, [overriding](http://www.w3.org/WAI/UA/UAAG20/#def-override) any values specified by the author. (Level A) ## DONE 26 April

**1.8.5 Indicate Viewport Position** [was 1.8.5]**:**

The user can determine the viewport's position relative to the full extent of the [rendered content](http://www.w3.org/WAI/UA/UAAG20/#def-rendered-content). (Level A)## DONE TPAC

**1.8.6: Zoom** [was 1.8.X]**:**

The user can rescale content within graphical viewports as follows: (Level A)## DONE 26 April 

1. Zoom in: to at least 500% of the default size; and
2. Zoom out: to at least 10% of the default size, so the content fits within the height or width of the viewport.

**1.8.7 Maintain point of regard** [was 1.8.Z]

: To the extent possible, the [point of regard](http://www.w3.org/WAI/UA/UAAG20/#def-point-of-regard) remains visible and at the same location within the viewport when the viewport is resized, when content is zoomed or scaled, or when content formatting is changed. (Level A) ## DONE 31 May 2012 

**1.8.8 Viewport History** [was 1.8.5]**:**

For [user agents](http://www.w3.org/WAI/UA/UAAG20/#def-user-agent) that implement a viewport history mechanism (e.g. "back" button), the user can return to any state in the viewport history that is allowed by the content, including a restored [point of regard](http://www.w3.org/WAI/UA/UAAG20/#def-point-of-regard), input focus and selection.(Level AA) ## DONE 5 April 2012

**1.8.9 Open on Request** [was 1.8.6]**:**

The user can specify whether author content can open new [top-level](http://www.w3.org/WAI/UA/UAAG20/#def-viewport-toplevel) [viewports](http://www.w3.org/WAI/UA/UAAG20/#def-viewport-toplevel) (e.g. windows or tabs). (Level AA) ## DONE TPAC

**1.8.10 Do Not Take Focus:**

If new [top-level viewports](http://www.w3.org/WAI/UA/UAAG20/#def-viewport-toplevel) (e.g. windows or tabs) are configured to open without explicit user request, the user can specify whether or not top-level viewports take the active [keyboard focus](http://www.w3.org/WAI/UA/UAAG20/#def-keyboard-focus) when they open. (Level AA) ## DONE TPAC

**1.8.11 Same UI:**

The user can specify that all [top-level viewports](http://www.w3.org/WAI/UA/UAAG20/#def-viewport-toplevel) (e.g. windows or tabs) follow the defined user interface configuration.(Level AA) ## DONE TPAC

**1.8.12: Reflowing Zoom:**

The user can request that when reflowable content in a graphical viewport is rescaled, it is reflowed so that one dimension of the content fits within the height or width of the viewport. (Level AA) ## DONE 26 April 2012 ## IER DONE 4 June 2012

Note: User agents are encouraged to allow users to override author instructions not to wrap content (e.g., nowrap).

**1.8.13 Webpage Bookmarks** [was 1.8.m, 1.8.13]**:**

The user can mark items in a webpage, then use shortcuts to navigate back to marked items. The user can specify whether a navigation mark disappears after a session, or is persistent across sessions. (Level AAA) ## DONE 31 May 2012 

Guideline 1.9 - Provide alternative views. [was 1.10] [[Implementing 1.9](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-alternative-views)]

**Summary**: The user can view the source of content (1.9.2), or an outline view of important elements. (1.9.1).

**1.9.1 Outline View:**

Users can view a navigable outline of [rendered content](http://www.w3.org/WAI/UA/UAAG20/#def-rendered-content) composed of labels for [important elements](http://www.w3.org/WAI/UA/UAAG20/#def-important-elements), and can move focus efficiently to these elements in the main viewport. (Level AA) ## DONE 26 April

**Note:** The important elements depend on the web content technology, but may include headings, table captions, and content sections.

**1.9.2 Source View:**

The user can view all [source text](http://www.w3.org/WAI/UA/UAAG20/#def-text-source) that is available to the [user agent](http://www.w3.org/WAI/UA/UAAG20/#def-user-agent). (Level AAA) ## DONE TPAC

Guideline 1.10 - Provide element information. [was 1.11] [[Implementing 1.10](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-info-link)]

**Summary**:The user agent presents information about content relationships (e.g. form labels, table headers)(1.10.1), and extended link information (e.g. title, internal vs. external) (1.10.2)

**1.10.1 Access Relationships:**

The user can access explicitly-defined relationships based on the user's position in content (e.g. show the label of a form control, show the headers of a table cell). (Level AA)## DONE 31 May 2012

**1.10.2 Access to Element Hierarchy:**

The user can determine the path of element nodes going from the root element of the element hierarchy to the currently focused or selected element. (Level AAA)   
## DONE 5 June 2012

PRINCIPLE 2. Ensure that the user interface is operable

Guideline 2.1 - Ensure full keyboard access. [[Implementing 2.1](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-keyboard-access)]

**Summary**: Users can operate all functions (2.1.1), and move focus (2.1.2) using just the keyboard. Users can activate important or common features with shortcut keys, (2.1.6), override keyboard shortcuts in content and user interface (2.1.4), escape keyboard traps (2.1.3), specify that selecting an item in a dropdown list or menu not activate that item or move to that new web page (2.1.4) and use standard keys for that platform (2.1.5).

**2.1.1 Keyboard Operation:**

All functionality can be operated via the [keyboard](http://www.w3.org/WAI/UA/UAAG20/#def-keyboard) using sequential or direct keyboard commands that do not require specific timings for individual keystrokes, except where the underlying function requires input that depends on the path of the user's movement and not just the endpoints (e.g. free hand drawing). This does not forbid and should not discourage providing other input methods in addition to keyboard operation including mouse, touch, gesture and speech. (Level A) ## DONE 9 April 2012

**2.1.2 Keyboard Focus** (former 1.9.2)**:**

Every [viewport](http://www.w3.org/WAI/UA/UAAG20/#def-viewport) has an active or inactive [keyboard focus](http://www.w3.org/WAI/UA/UAAG20/#def-keyboard-focus) at all times. (Level A) ## DONE 9 April 2012

**2.1.3 No Keyboard Trap** (former 2.1.5)**:**

If keyboard focus can be moved to a component using a keyboard interface (including nested user agents), then focus can be moved away from that component using only a keyboard interface. If this requires more than unmodified arrow or tab keys (or other standard exit methods), users are advised of the method for moving focus away. (Level A)## DONE TPAC

**2.1.4 Separate Selection from Activation** (former 2.1.4)**:**

The user can specify that focus and selection can be moved without causing further changes in focus, selection, or the state of controls, by either the [user agent](http://www.w3.org/WAI/UA/UAAG20/#def-user-agent) or author content. (Level A) ## DONE 8 March 2012

**2.1.5 Follow Text Keyboard Conventions** (former 2.1.7)**:**

The user agent follows keyboard conventions for the [operating environment](http://www.w3.org/WAI/UA/UAAG20/#def-operating-environment).(Level A) ## DONE TPAC ## DONE 4 June 2012

**2.1.6 Efficient Keyboard Access:**

The [user agent](http://www.w3.org/WAI/UA/UAAG20/#def-user-agent) user interface includes mechanisms to make keyboard access more efficient than sequential keyboard access. (Level A)## DONE 9 April 2012

**~~2.1.9 [deleted] Allow Override of User Interface Keyboard Commands~~:**

## DONE TPAC ## Merged with 2.3.5 on 2 August 2012

Guideline 2.2 - Provide sequential navigation [new, includes former 2.1.8 and 1.9.8, and a new SC] [[Implementing 2.2](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-sequential-navigation)]

**Summary**:Users can use the keyboard to navigate sequentially (2.2.3) to all the operable elements (2.2.1) in the viewport as well as between viewports (2.2.2). Users can optionally disable wrapping or request a signal when wrapping occurs (2.2.4).

**2.2.1 Sequential Navigation Between Elements**[replaces 1.9.8 Bi-Directional and 2.1.8 Keyboard Navigation]**:**

The user can move the keyboard focus backwards and forwards through all [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) enabled elements in the current viewport. (Level A)

**2.2.2 Sequential Navigation Between Viewports** [new]**:**

The user can move the keyboard focus backwards and forwards between viewports, without having to [sequentially navigate](http://www.w3.org/WAI/UA/UAAG20/#def-sequential-navigation) all the elements in a viewport. (Level A) ## DONE 8 March 2012

**2.2.3 Default Navigation Order** (former 1.9.9)**:**

If the author has not specified a navigation order, the default [sequential navigation](http://www.w3.org/WAI/UA/UAAG20/#def-sequential-navigation) order is the document order. (Level A) ## DONE 23 March 2012

**2.2.4 Options for Wrapping in Navigation** (new):

The user can prevent sequential navigation from wrapping the focus at the beginning or end of a document, and can request notification when such wrapping occurs. (Level AA) ## DONE 23 March 2012

Guideline 2.3 - Provide direct navigation and activation [includes former 2.1.6, 2.1.7, 2.1.11] [[Implementing 2.3](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-direct-navigation-and-activation)]

**Summary**: Users can navigate directly (e.g. keyboard shortcuts) to important elements (2.3.1) with the option of immediate activation of the operable elements (2.3.3). Display commands with the elements to make it easier for users to discover the commands (2.3.2 & 2.3.4). The user can remap and save direct commands (2.3.5).

**2.3.1 Direct Navigation to Important Elements** (former 2.7.4)**:**

The user can navigate directly to any [important elements](http://www.w3.org/WAI/UA/UAAG20/#def-important-elements) (e.g. structural or operable) in rendered content. (Level AA) ## DONE 23 March 2012 ## DONE 4 June 2012

**2.3.2 Present Direct Commands from Rendered Content (enhanced)** (former 2.1.6)**:**

The user can have any [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) direct commands in rendered content (e.g. accesskey, landmark) be presented with their associated elements (e.g. Alt+R to reply to a web email). (Level AA) ## DONE 28 March 2012

**2.3.3 Direct activation of Enabled Elements** (former 2.7.6)**:**

The user can move directly to and activate any enabled element in rendered content. (Level A) ## DONE 23 March 2012

**2.3.4 Present Direct Commands in User Interface** (former 2.1.7)**:**

The user can have any direct commands in the user agent user interface (e.g. keyboard shortcuts) be presented with their associated user interface controls (e.g. "Ctrl+S" displayed on the "Save" menu item and toolbar button). (Level AA) ## DONE 28 March 2012

**2.3.5 Customize Keyboard Commands:**

The user can override any keyboard shortcut including recognized author supplied shortcuts (e.g. accesskeys) and user agent user interface controls, except for conventional bindings for the operating environment (e.g. arrow keys for navigating within menus). The rebinding options must include single-key and key-plus-modifier keys if available in the operating environment. The user must be able to save these settings beyond the current session. (Level AA)## DONE 8 March 2012 MERGED WITH 2.1.9 on 2 August 2012

Guideline 2.4 (former 2.6) - Provide text search. [[Implementing 2.4](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-search-text)]

**Summary**: Users can search rendered content (2.4.1) forward or backward (2.4.2) and can have the matched content highlighted in the viewport (2.4.3). The user is notified if there is no match (2.4.4). Users can also search by case and for text within alternative content (2.4.5).

**2.4.1 Text Search:**

The user can perform a search within rendered content (e.g. not hidden with a style), including rendered text alternatives and rendered generated content, for any sequence of printing characters from the [document character set](http://www.w3.org/WAI/UA/UAAG20/#def-doc-char-set). (Level A)

**2.4.2 Find Direction:**

 The user can search forward or backward in rendered content. (Level A)## DONE 23 March 2012

**2.4.3 Match Found:**

When a search operation produces a match, the matched content is [highlighted](http://www.w3.org/WAI/UA/UAAG20/#def-highlight), the viewport is scrolled if necessary so that the matched content is within its visible area, and the user can search from the location of the match. (Level A) ## DONE 23 March 2012 ## DONE 5 June 2012

**2.4.4 Alert on Wrap or No Match:**

The user can be [notified](http://www.w3.org/WAI/UA/UAAG20/#def-alert) when there is no match to a search operation. The user can be notified when the search continues from the beginning or end of content. (Level A) ## DONE 23 March 2012

**2.4.5 Search alternative content:**

The user can perform text searches within textual alternative content (e.g. [text alternatives for non-text content](http://www.w3.org/WAI/UA/UAAG20/#def-conditional-content), captions) even when the textual alternative content is not rendered onscreen. (Level AA) ## DONE 23 March 2012

Guideline 2.5 (former 2.7) - Provide structural navigation. [[Implementing 2.5](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-nav-structure)]

**Summary**: Users can view (2.5.1), navigate (2.5.2), and configure the elements used in navigating (2.5.3) content hierarchy.

**2.5.1 Location in Hierarchy:** [was 2.5.3]

When the user agent is presenting hierarchical information, but the hierarchy is not reflected in a standardized fashion in the [DOM](http://www.w3.org/WAI/UA/UAAG20/#def-dom) or [platform accessibility services](http://www.w3.org/WAI/UA/UAAG20/#def-access-platform-arch), the user can view the path of nodes leading from the root of the hierarchy to a specified element. (Level AA) ## DONE 23 August 2012 .

**2.5.2 Navigate by structural element** [was 2.5.5]**:**

The user agent provides at least the following types of structural navigation, where the structure types exist:(Level AA) 

1. by heading
2. within tables

**2.5.3 Configure Elements for Structural Navigation** [was 2.5.7]**:**

The user can configure sets of [important elements](http://www.w3.org/WAI/UA/UAAG20/#def-important-elements) (including element types) for structured navigation and hierarchical/outline view. (Level AAA)

Guideline 2.6 (former 2.2) - Provide access to event handlers [[Implementing 2.6](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-device-independent-handlers)]

**Summary**:Users can interact with web content by mouse, keyboard, voice input, gesture, or a combination of input methods. Users can discover what event handlers (e.g. onmouseover) are available at the element and activate an element's events individually (2.6.1).

**2.6.1 Access to input methods:**

The user can discover [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) input methods explicitly associated with an element, and activate those methods in a modality independent manner. (Level AA) ## DONE 23 March 2012

Guideline 2.7 - Configure and store preference settings [[Implementing 2.7](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-store-prefs)]

**Summary**: Users can restore preference settings to default (2.7.2), and accessibility settings persist between sessions (2.7.1). Users can manage multiple sets of preference settings (2.7.3), and adjust preference setting outside the user interface so the current user interface does not prevent access (2.7.4). It's also recommended that groups of settings can be transported to compatible systems (2.7.5).

**2.7.1 Persistent Accessibility Settings** [was 2.7.2]**:**

User agent accessibility preference settings persist between sessions. (Level A)## DONE TPAC

**2.7.2 Restore all to default** [was 2.7.3]**:**

 The user can restore all preference settings to default values. (Level A)## DONE TPAC

**2.7.3 Multiple Sets of Preference Settings** [was 2.7.4]**:**

 The user can save and retrieve multiple sets of user agent preference settings.(Level AA)## DONE TPAC

**2.7.4 Change preference settings outside the user interface** [was 2.7.6]**:**

The user can adjust any preference settings required to meet the User Agent Accessibility Guidelines (UAAG) 2.0 from outside the user agent user interface. (Level AA)## DONE TPAC## DONE 5 June 2012 

**2.7.5 Portable Preference Settings** [was 2.7.7]**:**

The user can transfer all compatible user agent preference settings between computers.(Level AAA) ## DONE 4 June 2012

Guideline 2.8 - Customize display of GUI controls [[Implementing 2.8](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-configure-controls)]

**Summary**: It's recommended that users can add, remove and configure the position of graphical user agent controls and restore them to their default settings (2.8.1).

**2.8.1 Customize display of controls representing user interface commands, functions, and extensions:**

The user can customize which user agent commands, functions, and extensions are displayed within the user agent's user interface as follows:(Level AA)

1. **Show**: The user can choose to display any controls, which can include user installed extensions, available within the user agent user interface. It is acceptable to limit the total number of controls that are displayed onscreen.
2. **Simplify**: The user can simplify the default user interface by choosing to display only commands essential for basic operation (e.g. by hiding some control).
3. **Reposition**: The user can choose to reposition individual controls within containers (e.g. Toolbars or tool palettes), as well as reposition the containers themselves to facilitate physical access (e.g. To minimize hand travel on touch screens, or to facilitate preferred hand access on handheld mobile devices).
4. **Assign Activation Keystrokes or Gestures**: The user can choose to view, assign or change default keystrokes or gestures used to activate controls.
5. **Reset**: The user has the option to reset the containers and controls to their original configuration.

**2.8.2 Reset Toolbar Configuration:**

## DELETED 15 November 2012

Guideline 2.9 (former 2.3) - Allow time-independent interaction. [[Implementing 2.9](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-time-independent)]

**Summary**: Users can extend the time limit for user input when such limits are controllable by the user agent (2.9.1); by default, the user agent shows the progress of content in the process of downloading (2.9.2).

**2.9.1 Adjustable Timing:**

Where time limits for user input are [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) and controllable by the user agent, the user can extend the time limits. (Level A)## DONE 26 March 2012

**2.9.2 Retrieval Progress:**

By default, the user agent shows the progress of content retrieval. (Level A)## DONE 26 March 2012

Guideline 2.10 (former 2.4) - Help users avoid flashing that could cause seizures. [[Implementing 2.10](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-prevent-flash)]

**Summary**: To help users avoid seizures, the default configuration prevents the browser user interface and rendered content from flashing more than three times a second above a luminescence or color threshold (2.10.1), or does not flash at all (2.10.2).

**2.10.1 Three Flashes or Below Threshold:**

 In its default configuration, the user agent does not display any user interface components or [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) content that flashes more than three times in any one-second period, unless the flash is below the general flash and red flash thresholds. (Level A)## DONE 9 April 2012

**2.10.2 Three Flashes:**

 In its default configuration, the user agent does not display any user interface components or [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize)content that flashes more than three times in any one-second period (regardless of whether not the flash is below the general flash and red flash thresholds). (Level AAA) ## DONE 9 April 2012

Guideline 2.11 - Provide control of content that may reduce accessibility. (former 2.9) [[Implementing 2.11](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-control-inaccessible-content)]

**Summary**: The user can control background images (2.11.1); present placeholders for time-based media (2.11.2) and executable regions (2.11.3), or block all executable content (2.11.4); adjust playback (2.11.5), stop/pause/resume (2.11.6), navigate, (2.11.7) and specify tracks for prerecorded time-based media (2.11.9); and adjust contrast and brightness of visual time-based media (2.11.10).

**Applicability Notes:**

Guideline 2.11 and its success criteria only apply to images, animations, video, audio, etc. that the user agent can [recognize](http://www.w3.org/WAI/UA/UAAG20/#def-recognize).

**2.11.1 Background Image Toggle:**

)## DELETED 13 December 2012 (see 1.1.x)

**2.11.2 Time-Based Media Load-Only:**

The user can override the play on load of [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) time-based media content such that the content is not played until [explicit user request](http://www.w3.org/WAI/UA/UAAG20/#def-user-request). (Level A)## DONE 9 August 2012

**2.11.3 Execution Placeholder:**

 The user can render a [placeholder](http://www.w3.org/WAI/UA/UAAG20/#def-placeholder) instead of executable content that would normally be contained within an on-screen area (e.g. Applet, Flash), until [explicit user request](http://www.w3.org/WAI/UA/UAAG20/#def-user-request) to execute. (Level A)## DONE 9 August 2012

**2.11.4 Execution Toggle:**

 The user can turn on/off the execution of executable content that would not normally be contained within a particular area (e.g. Javascript). (Level A)## DONE TPAC

**2.11.5 Playback Rate Adjustment for Prerecorded Content:**

The user can adjust the playback rate of prerecorded time-based media content, such that all of the following are true: (Level A)

1. The user can adjust the playback rate of the time-based media tracks to between 50% and 250% of real time.
2. Speech whose playback rate has been adjusted by the user maintains pitch in order to limit degradation of the speech quality.
3. Audio and video tracks remain synchronized across this required range of playback rates.
4. The user agent provides a function that resets the playback rate to normal (100%).

**2.11.6 Stop/Pause/Resume Time-Based Media:**

The user can stop, pause, and resume rendered audio and [animation](http://www.w3.org/WAI/UA/UAAG20/#def-animation) content (including video, animated images, and changing text) that last three or more seconds at their default playback rate. (Level A) ## DONE TPAC

**2.11.7 Navigate Time-Based Media:**

 The user can navigate along the timebase using a continuous scale, and by [relative time units](http://www.w3.org/WAI/UA/UAAG20/#def-relative-time-units)within rendered audio and animations (including video and animated images) that last three or more seconds at their default playback rate. (Level A) ## DONE 9 August 2012

**2.11.8 Semantic Navigation of Time-Based Media:**

 The user can navigate by semantic structure within the time-based media, such as by chapters or scenes present in the media (Level AA).## DONE TPAC

**2.11.9 Track Enable/Disable of Time-Based Media:**

During time-based media playback, the user can determine which tracks are available and select or deselect tracks, overriding global default settings for captions, audio descriptions, etc. (Level AA)## DONE TPAC

**2.11.10 Video Contrast and Brightness** [was 2.11.12]**:**

Users can adjust the contrast and brightness of visual time-based media. (Level AAA)

2.11.11 Scale and position visual alternative media tracks:[2.11.11 MOVED to 1.1.3 and 1.1.5 on 2 August 2012]

Guideline 2.12 - Other Input Devices [[Implementing 2.12](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-other-devices)]

**Summary**: For all input devices supported by the platform, the user agents should let the user perform all functions aside from entering text (2.12.2), and enter text with any platform-provided features (2.12.1). If possible, it is also encouraged to let the user enter text even if the platform does not provide such a feature (2.12.3).

**2.12.1 Support Platform Text Input Devices:**

If the [platform](http://www.w3.org/WAI/UA/UAAG20/#def-platform) supports text input using an input device, the user agent is compatible with this functionality. (Level A)

**2.12.2 Operation With Any Device:**

If an input device is supported by the [platform](http://www.w3.org/WAI/UA/UAAG20/#def-platform) , all user agent functionality other than text input can be operated using that device. (Level AA)

**2.12.3 Text Input With Any Device:**

If an input device is supported by the [platform](http://www.w3.org/WAI/UA/UAAG20/#def-platform) , all user agent functionality including text input can be operated using that device. (Level AAA)

PRINCIPLE 3: Ensure that the user interface is understandable

Guideline 3.1 - Help users avoid unnecessary messages. [[Implementing 3.1](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-avoid-unnecessary-messages)]

**Summary**: Users can turn off non-essential messages from the author or user-agent.

**3.1.2 Reduce Interruptions:**

The user can avoid or defer [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) non-essential or low priority messages and updating/changing information in the user agent user interface and rendered content.(Level AA)## DONE 5 June 2012

Guideline 3.2 - Help users avoid and correct mistakes. [[Implementing 3.2](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-avoid-mistakes)]

**Summary**: Users can have form submissions require confirmation (3.2.1), go back after navigating (3.2.2), and have their text checked for spelling errors (3.2.3).

**3.2.1 Form Submission:**

 The user can specify whether or not [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) form submissions must be confirmed. (Level AA)## DONE 15 March 2012

**3.2.2 Back Button**

: The user can reverse [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) navigation between web addresses (e.g. standard "back button" functionality).(Level AA) ## DONE 15 March 2012 & 19 July 2012

**3.2.3 Provide spell checking functionality:**

User agents provide spell checking functionality for text created inside the user agent.(Level AA)## DONE 5 June 2012

**3.2.4 Text Entry Undo:**

The user can reverse [recognized](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) text entry actions prior to submission. (Level A)

**Note**: Submission can be triggered in many different ways, such as clicking a submit button, typing a key in a control with an onkeypress event, or by a script responding to a timer.

## DONE 19 July 2012 (was 3.2.X)

**3.2.5 Settings Change Confirmation:**

If the user agent provides mechanisms for changing its user interface settings, it either allows the user to reverse the setting changes, or the user can require user confirmation to proceed.(Level A)   
## DONE 19 July 2012 (was 3.2.Y)

Guideline 3.3 - Document the user agent user interface including accessibility features.[[Implementing 3.3](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-doc-access-features)]

**Summary**: User documentation is available in an accessible format (3.3.1), it includes accessibility features (3.3.2), delineates differences between versions (3.3.3), provides a centralized views of conformance UAAG2.0 (3.3.4), and is available as context sensitive help in the UA (3.3.5).

**3.3.1 Accessible documentation:**

The product [documentation](http://www.w3.org/WAI/UA/UAAG20/#def-documentation) is available in a format that meets success criteria of WCAG 2.0 Level "A" or greater. (Level A)## DONE TPAC

**3.3.2 Document Accessibility Features:**

All features of the user agent that meet User Agent Accessibility Guidelines 2.0 success criteria are documented. (Level A)## DONE TPAC

**3.3.3 Changes Between Versions:**

Changes to features that meet UAAG 2.0 success criteria since the previous user agent release are documented. (Level AA)## DONE TPAC

**3.3.4 Centralized View:**

There is a dedicated section of the documentation that presents a view of all features of the user agent necessary to meet the requirements of User Agent Accessibility Guidelines 2.0. (Level AAA)## DONE TPAC

Guideline 3.4 - The user agent must behave in a predictable fashion. [[Implementing 3.4](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-predictable-operation)]

**Summary**: Users can prevent non-requested focus changes (3.4.1).

**3.4.1 Avoid unpredictable focus**[formerly 3.4.2, before that 5.4.2, and 1.9.20, broadened]:

The user can prevent focus changes that are not a result of explicit user request. (Level A) ## DONE TPAC

PRINCIPLE 4: Facilitate programmatic access

Guideline 4.1 - Facilitate programmatic access to assistive technology [[Implementing 4.1](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-AT-access)]

**Summary**: Be compatible with assistive technologies by supporting platform standards (4.1.1), including providing information about all menus, buttons, dialogs, etc. (4.1.2, 4.1.6), access to DOMs (4.1.4), and access to structural relationships and meanings, such as what text or image labels a control or serves as a heading (4.1.5). Where something can't be made accessible, provide an accessible alternative version, such as a standard window in place of a customized window (4.1.3). Make sure that that programmatic exchanges are quick and responsive (4.1.7).

**4.1.1 Platform Accessibility Services:**

The user agent supports relevant [platform accessibility services](http://www.w3.org/WAI/UA/UAAG20/#def-access-platform-arch). (Level A) ## DONE TPAC, 19 July 2012

**4.1.2 Name, Role, State, Value, Description:**

For all user interface components including user interface, rendered content, generated content, and alternative content, the user agent makes available the name, role, state, value, and description via a [platform accessibility services](http://www.w3.org/WAI/UA/UAAG20/#def-access-platform-arch). (Level A)## DONE 2 August 2012

**4.1.3 Accessible Alternative:**

If a component of the user agent user interface cannot be exposed through the [platform accessibility services](http://www.w3.org/WAI/UA/UAAG20/#def-access-platform-arch), then the user agent provides an equivalent alternative that is exposed through the platform accessibility service. (Level A)## DONE TPAC

**4.1.4 Programmatic Availability of DOMs:**

 If the user agent implements one or more [DOMs](http://www.w3.org/WAI/UA/UAAG20/#def-dom), they must be made programmatically available to assistive technologies. (Level A) ## DONE TPAC

**4.1.5 Write Access:**

 If the user can modify the state or value of a piece of content through the user interface (e.g., by checking a box or editing a text area), the same degree of write access is available programmatically. (Level A) ## DONE Post TPAC

**4.1.6 Expose Accessible Properties:**

 If any of the following properties are supported by the [platform accessibility services](http://www.w3.org/WAI/UA/UAAG20/#def-access-platform-arch), make the properties available to the accessibility platform architecture: (Level A) ## DONE 8 March 2012

1. the bounding dimensions and coordinates of onscreen elements
2. font family of text
3. font size of text
4. foreground color of text
5. background color of text.
6. change state/value notifications
7. selection
8. highlighting
9. input device focus
10. direct keyboard commands
11. underline of menu items (keyboard command/shortcuts)

**4.1.7 Timely Communication:**

 For APIs implemented to satisfy the requirements of UAAG 2.0, ensure that programmatic exchanges proceed at a rate such that users do not perceive a delay. (Level A) ## DONE TPAC

PRINCIPLE 5: Comply with applicable specifications and conventions

Guideline 5.1 - Comply with applicable specifications and conventions. [[Implementing 5.1](http://www.w3.org/WAI/UA/2013/ED-IMPLEMENTING-UAAG20-20130222/#gl-obs-env-conventions)]

**Summary**: When the browser's menus, buttons, dialogs, etc. are authored in HTML or similar standards, they need to meet W3C's Web Content Accessibility Guidelines.

**5.1.1 WCAG Compliant** [was 5.2.1]**:**

Web-based authoring tool user interfaces meet the WCAG 2.0 success criteria: Level A to meet WCAG 2.0 Level A success criteria; Level AA to meet WCAG 2.0 Level A and AA success criteria; Level AAA to meet all WCAG 2.0 success criteria. (Level AAA) 

Note: This success criterion does not apply to non-Web-based user agent user interfaces, but does include any parts of non-Web-based user agents that are Web-based (e.g. help systems).

**5.1.2 Implement accessibility features of content specs** [was 5.3.1]**:**

Implement and cite in the conformance claim the accessibility features of content specifications. Accessibility features are those that are either (Level A):

1. identified as such in the specification or
2. allow authors to satisfy a requirement of WCAG.

**5.1.3 Implement Accessibility Features of platform:**

If the user agent contains non-web-based user interfaces, then those user interfaces follow user interface accessibility guidelines for the [platform](http://www.w3.org/WAI/UA/UAAG20/#def-platform). (Level A)

**5.1.4 Handle Unrendered Technologies:**

If the [user agent](http://www.w3.org/WAI/UA/UAAG20/#def-user-agent) does not render a technology, the user can choose a way to handle content in that technology (e.g. by launching another application or by saving it to disk). (Level A)

**5.1.5 Alternative content handlers:**

The user can select content elements and have them rendered in alternative viewers. (Level AA)## DONE TPAC## DONE 28 June

**5.1.6 Enable Reporting of User Agent Accessibility Faults:**

The user agent provides a mechanism for users to report user agent accessibility issues.(Level AAA)  
## DONE 12 July 2012

**Applicability Note:**

When a rendering requirement of another specification contradicts a requirement of UAAG 2.0, the user agent may disregard the rendering requirement of the other specification and still satisfy this guideline.

Conformance

This section is normative.

Conformance means that the user agent satisfies the success criteria defined in the guidelines section. This conformance section describes conformance and lists the conformance requirements.

Conformance Requirements @@ [704](https://www.w3.org/WAI/UA/tracker/actions/704), [648](https://www.w3.org/WAI/UA/tracker/actions/648)

In order for a Web page to conform to UAAG 2.0, one of the following levels of conformance is met in full.

* Level A: For Level A conformance (the minimum level of conformance), the user agent satisfies all the Level A success criteria.
* Level AA: For Level AA conformance, the user agent satisfies all the Level A and Level AA Success Criteria.
* Level AAA: For Level AAA conformance, the user agent satisfies all the Level A, Level AA and Level AAA Success Criteria.

Note 1: Although conformance can only be achieved at the stated levels, developers are encouraged to report (in their claim) any progress toward meeting success criteria from all levels beyond the achieved level of conformance.

Conformance Claims (Optional)

If a conformance claim is made, the conformance claim must meet the following conditions and include the following information (user agents can conform to UAAG 2.0 without making a claim):

Conditions on Conformance Claims

* At least one version of the conformance claim must be published on the web as a document meeting level "A" of WCAG 2.0. A suggested metadata description for this document is "UAAG 2.0 Conformance Claim".
* Whenever the claimed conformance level is published (e.g. product information website), the URI for the on-line published version of the conformance claim must be included.
* The existence of a conformance claim does not imply that the W3C has reviewed the claim or assured its validity.
* Claimants may be anyone (e.g. user agent developers, journalists, other third parties).
* Claimants are solely responsible for the accuracy of their claims (including claims that include products for which they are not responsible) and keeping claims up to date.
* Claimants are encouraged to claim conformance to the most recent version of the User Agent Accessibility Guidelines Recommendation.

Required Components of an UAAG 2.0 Conformance Claim

1. Claimant name and affiliation.
2. Date of the claim.
3. Conformance level satisfied.
4. User agent information: The name of the user agent and sufficient additional information to specify the version (e.g. vendor name, version number (or version range), required patches or updates, human language of the user interface or documentation).   
   Note: If the user agent is a collection of software components (e.g. a browser and extensions or plugins), then the name and version information must be provided separately for each component, although the conformance claim will treat them as a whole. As stated above, the Claimant has sole responsibility for the conformance claim, not the developer of any of the software components.
5. Included Technologies: A list of the web content technologies (including version numbers) rendered by the user agent that the claimant is including in the conformance claim. By including a web content technology, the Claimant is claiming that the user agent meets the requirements of UAAG 2.0 during the rendering of web content using that web content technology.   
   Note 1: Web content technologies may be a combination of constituent web content technologies. For example, an image technology (e.g. PNG) might be listed together with a markup technology (e.g. HTML) since web content in the markup technology is used make web content in the image technology accessible (e.g. a PNG graph is made accessible using an HTML table).
6. Excluded Technologies: A list of any web content technologies produced by the user agent that the Claimant is excluding from the conformance claim. The user agent is not required to meet the requirements of UAAG 2.0 during the production of the web content technologies on this list.
7. Declarations: For each success criterion: A declaration of whether or not the success criterion has been satisfied; or a declaration that the success criterion is not applicable and a rationale for nonapplicability.
8. Platform(s): The [platform](http://www.w3.org/WAI/UA/UAAG20/#def-platform) (s) upon which the user agent was evaluated: For user agent platform(s) (used to evaluate web-based user agent user interfaces): provide the name and version information of the user agent(s). For platforms that are not user agents (used to evaluate non-web-based user agent user interfaces) provide: The name and version information of the platform(s) (e.g..[operating system](http://www.w3.org/WAI/UA/UAAG20/#def-operating-system) , etc.) and the name and version of the platform accessibility service(s) employed.

Optional Components of an UAAG 2.0 Conformance Claim

A description of how the UAAG 2.0 success criteria were met where this may not be obvious.

"Progress Towards Conformance" Statement

Developers of user agents that do not conform fully to a particular UAAG 2.0 conformance level are encouraged to publish a statement on progress towards conformance. The progress statement is the same as a conformance claim except an UAAG 2.0 conformance level that is being progressed towards, rather than one already satisfied, and report progress on success criteria not yet met. Authors of "Progress Towards Conformance" statement are solely responsible for the accuracy of their statements. Developers are encouraged to provide expected timelines for meeting outstanding success criteria within the Statement.

Disclaimer

Neither W3C, WAI, nor UAWG are not responsible for any aspect or result of any UAAG 2.0 conformance claim that has not been published under the authority of the W3C, WAI, or UAWG.

Appendix A: Glossary

This glossary is [normative](http://www.w3.org/WAI/UA/UAAG20/#def-normative).

[a](http://www.w3.org/WAI/UA/UAAG20/#a) · b · [c](http://www.w3.org/WAI/UA/UAAG20/#c) · [d](http://www.w3.org/WAI/UA/UAAG20/#d) · [e](http://www.w3.org/WAI/UA/UAAG20/#e) · [f](http://www.w3.org/WAI/UA/UAAG20/#f) · [g](http://www.w3.org/WAI/UA/UAAG20/#g) · [h](http://www.w3.org/WAI/UA/UAAG20/#h) · [i](http://www.w3.org/WAI/UA/UAAG20/#i) · j · k · l · m · [n](http://www.w3.org/WAI/UA/UAAG20/#n) · [o](http://www.w3.org/WAI/UA/UAAG20/#o) · [p](http://www.w3.org/WAI/UA/UAAG20/#p) · q · [r](http://www.w3.org/WAI/UA/UAAG20/#r) · [s](http://www.w3.org/WAI/UA/UAAG20/#s) · [t](http://www.w3.org/WAI/UA/UAAG20/#t) · [u](http://www.w3.org/WAI/UA/UAAG20/#u) · [v](http://www.w3.org/WAI/UA/UAAG20/#v) · [w](http://www.w3.org/WAI/UA/UAAG20/#w) · x · y · z

***accelerator key***

see [***keyboard command***](http://www.w3.org/WAI/UA/UAAG20/#def-accelerator-key)

***activate***

To carry out the behaviors associated with an [enabled element](http://www.w3.org/WAI/UA/UAAG20/#def-enabled-element) in the rendered content or a component of the [user agent user interface](http://www.w3.org/WAI/UA/UAAG20/#def-ua-ui).

***active input focus***

see [***focus***](http://www.w3.org/WAI/UA/UAAG20/#def-focus)

***active selection***

see [***focus***](http://www.w3.org/WAI/UA/UAAG20/#def-focus)

***alternative content***

[Content](http://www.w3.org/WAI/UA/UAAG20/#def-content) ot placeholder that can be used in place of default content that may not be universally accessible. Alternative content fulfills the same purpose as the original content. Examples include text alternatives for non-text content, captions for audio, audio descriptions for video, sign language for audio, media alternatives for time-based media. See [WCAG](http://www.w3.org/TR/WCAG20/) for more information.

***alternative content stack***

A set of [alternative content](http://www.w3.org/WAI/UA/UAAG20/#def-conditional-content) items. The items may be mutually exclusive (e.g. regular contrast graphic vs. high contrast graphic) or non-exclusive (e.g. caption track that can play at the same time as a sound track).

***animation***

Graphical [content](http://www.w3.org/WAI/UA/UAAG20/#def-content) rendered to automatically change over time, giving the user a visual perception of movement. Examples include video, animated images, scrolling text, programmatic animation (e.g. moving or replacing rendered objects).

***application programming interface (API)*,*****(conventional input/output/device API*)**

An application programming interface (API) defines how communication may take place between applications.

***assistive technology***

An assistive technology:

1. relies on services (such as retrieving [Web resources](http://www.w3.org/WAI/UA/UAAG20/#def-web-resource) and parsing markup) provided by one or more other "host" user agents. Assistive technologies communicate data and messages with host user agents by using and monitoring [APIs](http://www.w3.org/WAI/UA/UAAG20/#def-api).
2. provides services beyond those offered by the host user agents to meet the requirements of users with disabilities. Additional services include alternative renderings (e.g. as synthesized speech or magnified content), alternative input methods (e.g. voice), additional navigation or orientation mechanisms, and content transformations (e.g. to make tables more accessible).

Examples of assistive technologies that are important in the context of UAAG 2.0 include the following:

* screen magnifiers, which are used by people with visual disabilities to enlarge and change colors on the screen to improve the visual readability of rendered text and images.
* screen readers, which are used by people who are blind or have reading disabilities to read textual information through synthesized speech or braille displays.
* voice recognition software, which are used by some people who have physical disabilities to simulate the keyboard and mouse.
* alternative keyboards, which are used by some people with physical disabilities to simulate the keyboard and mouse.
* alternative pointing devices, which are used by some people with physical disabilities to simulate mouse pointing and button activations.

Beyond UAAG 2.0, assistive technologies consist of software or hardware that has been specifically designed to assist people with disabilities in carrying out daily activities. These technologies include wheelchairs, reading machines, devices for grasping, text telephones, and vibrating pagers. For example, the following very general definition of "assistive technology device" comes from the (U.S.) Assistive Technology Act of 1998 [*[AT1998]*](http://www.w3.org/WAI/UA/UAAG20/#ref-AT1998):

Any item, piece of equipment, or product system, whether acquired commercially, modified, or customized, that is used to increase, maintain, or improve functional capabilities of individuals with disabilities.

***audio***

The technology of sound reproduction. Audio can be created synthetically (including speech synthesis), streamed from a live source (such as a radio broadcast), or recorded from real world sounds.

***audio description* - (described video, video description or descriptive narration)**

An equivalent alternative that takes the form of narration added to the [audio](http://www.w3.org/WAI/UA/UAAG20/#def-audio) to describe important visual details that cannot be understood from the main soundtrack alone. Audio description of video provides information about actions, characters, scene changes, on-screen text, and other visual content. In standard audio description, narration is added during existing pauses in dialogue. In ***extended audio description***, the video is paused so that there is time to add additional description.

***authors***

The people who have worked either alone or collaboratively to create the content (e.g. content authors, designers, programmers, publishers, testers).

***author styles***

See [Style properties](http://www.w3.org/WAI/UA/UAAG20/#def-style-properties)

**background images**

Images that are rendered on the [base background](http://www.w3.org/WAI/UA/UAAG20/#def-base-background).

***base background***

The background of the content as a whole, such that no content may be layered behind it. In graphics applications the base background is often referred to as the canvas.

***blinking text***

Text whose visual rendering alternates between visible and invisible at any rate of change.

***captions* (caption)**

An equivalent alternative that takes the form of text presented and synchronized with *time-based* media to provide not only the speech, but also non-speech information conveyed through sound, including meaningful sound effects and identification of speakers. In some countries, the term "subtitle" is used to refer to dialogue only and "captions" is used as the term for dialogue plus sounds and speaker identification. In other countries, "subtitle" (or its translation) is used to refer to both. ***Open captions***are captions that are always rendered with a visual track; they cannot be turned off. ***Closed captions*** are captions that may be turned on and off. The captions requirements of UAAG 2.0 assume that the user agent can [recognize](http://www.w3.org/WAI/UA/UAAG20/#def-recognize) the captions as such.  
**Note:** Other terms that include the word "caption" may have different meanings in UAAG 2.0. For instance, a "table caption" is a title for the table, often positioned graphically above or below the table. In UAAG 2.0, the intended meaning of "caption" will be clear from context.

***collated text transcript***

A collated text transcript is a [text equivalent](http://www.w3.org/WAI/UA/UAAG20/#def-text-eq) of a movie or other animation. It is the combination of the [text transcript](http://www.w3.org/WAI/UA/UAAG20/#def-text-transcript) of the [audio track](http://www.w3.org/WAI/UA/UAAG20/#def-audio-track) and the text equivalent of the [visual track](http://www.w3.org/WAI/UA/UAAG20/#def-visual-track). For example, a collated text transcript typically includes segments of spoken dialogue interspersed with text descriptions of the key visual elements of a presentation (actions, body language, graphics, and scene changes). See also the definitions of [text transcript](http://www.w3.org/WAI/UA/UAAG20/#def-text-transcript) and [audio description](http://www.w3.org/WAI/UA/UAAG20/#def-audio-description). Collated text transcripts are essential for people who are deaf-blind.

***command*,*****direct command*,*****direct navigation command*,*****direct activation command*,*****sequential navigation command* ,*****spacial (directional) command*,*****structural navigation command***

**direct commands** apply to a specified item (e.g., button) or action (e.g. save function), regardless of the current focus location

**direct navigation commands** move focus to a specified item

**direct activation commands** activate the specified item (and may also move focus to it) or action

**sequential navigation commands** (sometimes called logical or linear navigation commands) move focus forwards and backwards through a list of items. The element list being navigated may be the list of all elements or just a subset (e.g. the list of headers, the list of links, etc.)

**structural navigation commands** move forwards, backwards, up and down a hierarchy

**spatial commands** (sometimes called directional commands), require the user to be cognizant of the spatial arrangement of items on the screen:

* spatial navigation commands move from one item to another based on direction on the screen
* spatial manipulation commands resize or reposition an item on the screen

***content (web content)*, empty content, reflowable content**

Information and sensory experience to be communicated to the user by means of a user agent, including code or markup that defines the content's structure, presentation, and interactions [adapted from [WCAG 2.0](http://www.w3.org/TR/WCAG20/)]

***empty content*** (which may be [alternative content](http://www.w3.org/WAI/UA/UAAG20/#def-conditional-content)) is either a null value or an empty string (e.g. one that is zero characters long). For instance, in HTML, alt="" sets the value of the alt attribute to the empty string. In some markup languages, an element may have empty content (e.g. the HR element in HTML).

***reflowable content*** is content that can be arbitrarily wrapped over multiple lines. The primary exceptions to reflowable content are graphics and video.

***continuous scale***

When interacting with a time-based media presentation, a continuous scale allows user (or programmatic) action to set the active playback position to any time point on the presentation timeline. The granularity of the positioning is determined by the smallest resolvable time unit in the media timebase.

***cursor***

see [***focus***](http://www.w3.org/WAI/UA/UAAG20/#def-focus)

***default***

see [***properties***](http://www.w3.org/WAI/UA/UAAG20/#def-Properties-and-Values)

***direct command*, *direct navigation command*, *direct activation command*, *linear navigation command* , *spacial (directional) command*, *structural navigation command***

see [**command**](http://www.w3.org/WAI/UA/UAAG20/#def-command)

***dimensions***

A viewport may also have temporal dimensions, for instance when audio, speech, animations, and movies are rendered. When the dimensions (spatial or temporal) of rendered content exceed the dimensions of the viewport, the user agent provides mechanisms such as scroll bars and advance and rewind controls so that the user can access the rendered content "outside" the viewport. Examples include: when the user can only view a portion of a large document through a small graphical viewport, or when audio content has already been played.

***document character set***

The internal representation of data in the source content by a user agent.

***document object*,*****(Document Object Model*, DOM)**

The Document Object Model is a platform- and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure and style of documents. The document can be further processed and the results of that processing can be incorporated back into the presented page. This is an overview of DOM-related materials here at W3C and around the web: <http://www.w3.org/DOM/#what>.

***document source*,*****(text source)***

Text the user agent renders upon user request to view the source of specific viewport content (e.g. selected content, frame, page).

***documentation***

Any information that supports the use of a user agent. This information may be found, for example, in manuals, installation instructions, the help system, and tutorials. Documentation may be distributed (e.g. as files installed as part of the installation, some parts may be delivered on CD-ROM, others on the Web). See [guideline 5.3](http://www.w3.org/WAI/UA/UAAG20/#gl-doc-access-features) for information about documentation.

***element*,*****element type***

UAAG 2.0 uses the terms "element" and "element type" primarily in the sense employed by the XML 1.0 specification ([*[XML]*](http://www.w3.org/WAI/UA/UAAG20/#ref-XML), section 3): an element type is a syntactic construct of a document type definition (DTD) for its application. This sense is also relevant to structures defined by XML schemas. UAAG 2.0 also uses the term "element" more generally to mean a type of content (such as video or sound) or a logical construct (such as a header or list).

***empty content***

see [***content***](http://www.w3.org/WAI/UA/UAAG20/#def-content)

***enabled element*,*****disabled element***

An [element](http://www.w3.org/WAI/UA/UAAG20/#def-element) with associated behaviors that can be activated through the user interface or through an [API](http://www.w3.org/WAI/UA/UAAG20/#def-api). The set of elements that a user agent enables is generally derived from, but is not limited to, the set of [elements](http://www.w3.org/WAI/UA/UAAG20/#def-element) defined by implemented markup languages. A disabled element is a potentially enabled element that is not currently available for activation (e.g. a "grayed out" menu item).

***equivalent alternative***

Acceptable substitute content that a user may not be able to access. An equivalent alternative fulfills essentially the same function or purpose as the original content upon presentation:

* text alternative: text that is available via the [operating environment](http://www.w3.org/WAI/UA/UAAG20/#def-operating-environment) that is used in place of non-text content (e.g. text equivalents for images, text transcripts for audio tracks, or collated text transcripts for a movie). [from [WCAG 2.0](http://www.w3.org/TR/WCAG20/)]
* full text alternative for synchronized media including any interaction: document including correctly sequenced text descriptions of all visual settings, actions, speakers, and non-speech sounds, and transcript of all dialogue combined with a means of achieving any outcomes that are achieved using interaction (if any) during the synchronized media. [from[WCAG 2.0](http://www.w3.org/TR/WCAG20/)]
* synchronized alternatives: present essential audio information visually (i.e. captions) and essential video information in an auditory manner (i.e. audio descriptions). [from [ATAG 2.0](http://www.w3.org/TR/ATAG20/)]

***events and scripting, event handler, event type***

User agents often perform a task when an event having a particular "event type" occurs, including a user interface event, a change to content, loading of content, or a request from the [operating environment](http://www.w3.org/WAI/UA/UAAG20/#def-operating-environment). Some markup languages allow authors to specify that a script, called an ***event handler***, be executed when an event of a given type occurs. An event handler is ***explicitly associated with an element*** through scripting, markup or the [DOM](http://www.w3.org/WAI/UA/UAAG20/#def-dom).

***explicit user request***

An interaction by the user through the [user agent user interface](http://www.w3.org/WAI/UA/UAAG20/#def-ua-ui), the [focus](http://www.w3.org/WAI/UA/UAAG20/#def-focus), or the [selection](http://www.w3.org/WAI/UA/UAAG20/#def-selection). User requests are made, for example, through [user agent user interface controls](http://www.w3.org/WAI/UA/UAAG20/#def-ui-control) and [keyboard commands](http://www.w3.org/WAI/UA/UAAG20/#def-keyboard-command). Some examples of explicit user requests include when the user selects "New viewport," responds "yes" to a prompt in the user agent's user interface, configures the user agent to behave in a certain way, or changes the selection or focus with the keyboard or pointing device. **Note:** Users can make errors when interacting with the user agent. For example, a user may inadvertently respond "yes" to a prompt instead of "no." This type of error is considered an explicit user request.

***focus* (active input focus, active selection, cursor, focus cursor, focusable element, highlight, inactive input focus, inactive selection, input focus, keyboard focus, pointer, pointing device focus, selection, split focus, text cursor)**

**Hierarchical Summary of some focus terms**

* Input Focus (active/inactive)
  + Keyboard Focus (active/inactive)
    - Cursor (active/inactive)
      * Focus cursor (active/inactive)
      * Text cursor (active/inactive)
  + Pointing device focus (active/inactive)
    - Pointer

***active input focus***

The [input focus](http://www.w3.org/WAI/UA/UAAG20/#def-input-focus) location in the active [viewport](http://www.w3.org/WAI/UA/UAAG20/#def-viewport). The active focus is in the active viewport, while the inactive input focus is the inactive viewport. The [active input focus](http://www.w3.org/WAI/UA/UAAG20/#def-active-input-focus) is usually visibly indicated. In UAAG 2.0 "active input focus" generally refers to the active keyboard input focus. @@ Editors' Note: this term is not used in the document other than the glossary.@@

***active selection***

The selection that will currently be affected by a user command, as opposed to selections in other viewports, called inactive selections, which would not currently be affected by a user command. @@ Editors' Note: this term is not used in the document other than the glossary.@@

***conform***

see [***support***](http://www.w3.org/WAI/UA/UAAG20/#def-support)

***cursor***

Visual indicator showing where keyboard input will occur. There are two types of cursors: [focus cursor](http://www.w3.org/WAI/UA/UAAG20/#def-focus-cursor) (e.g. the dotted line around a button) and [text cursor](http://www.w3.org/WAI/UA/UAAG20/#def-text-cursor) (e.g. the flashing vertical bar in a text field, also called a 'caret'). Cursors are active when in the active viewport, and inactive when in an inactive viewport.

***focus cursor***

Indicator that [highlights](http://www.w3.org/WAI/UA/UAAG20/#def-highlight) a user interface element to show that it has [keyboard focus](http://www.w3.org/WAI/UA/UAAG20/#def-keyboard-focus), e.g. a dotted line around a button, or brightened title bar on a window. There are two types of cursors: focus cursor (e.g. the dotted line around a button) and [text cursor](http://www.w3.org/WAI/UA/UAAG20/) (e.g. the flashing vertical bar in a text field).

***focusable element***

Any element capable of having [input focus](http://www.w3.org/WAI/UA/UAAG20/#def-input-focus), e.g. link, text box, or menu item. In order to be accessible and fully usable, every focusable element should take [keyboard focus](http://www.w3.org/WAI/UA/UAAG20/#def-keyboard-focus), and ideally would also take [pointer focus](http://www.w3.org/WAI/UA/UAAG20/#def-pointing-device-focus).

***highlight, highlighted, highlighting***

Emphasis indicated through the user interface. For example, user agents highlight content that is selected, focused, or matched by a search operation. Graphical highlight mechanisms include dotted boxes, changed colors or fonts, underlining, adjacent icons, magnification, and reverse video. Synthesized speech highlight mechanisms include alterations of voice pitch and volume ("speech prosody"). User interface items may also be highlighted, for example a specific set of foreground and background colors for the title bar of the active window. Content that is highlighted may or may not be a [selection](http://www.w3.org/WAI/UA/UAAG20/#def-selection).

***inactive input focus***

An input focus location in an inactive viewport such as a background window or pane. The inactive input focus location will become the active input focus location when input focus returns to that viewport. An inactive input focus may or may not be visibly indicated.

***inactive selection***

A selection that does not have the [input focus](http://www.w3.org/WAI/UA/UAAG20/#def-input-focus) and thus does not take input events.

***input focus***

The place where input will occur if a viewport is active. Examples include keyboard focus and pointing device focus. Input focus can also be active (in the active viewport) or inactive (in an inactive viewport).

***keyboard focus***

The screen location where keyboard input will occur if a viewport is active. Keyboard focus can be active (in the active viewport) or inactive (in an inactive viewport). See keyboard interface definition for types of keyboards included and what constitutes a keyboard.

***keyboard interface***

Keyboard interfaces are programmatic services provided by many platforms that allow operation in a device independent manner. A keyboard interface can allow keystroke input even if particular devices do not contain a hardware keyboard (e.g., a touchscreen-controlled device can have a keyboard interface built into its [operating system](http://www.w3.org/WAI/UA/UAAG20/#def-operating-system) to support onscreen keyboards as well as external keyboards that may be connected).   
*Note:* Keyboard-operated mouse emulators, such as MouseKeys, do not qualify as operation through a keyboard interface because these emulators use pointing device interfaces, not keyboard interfaces. [from ATAG 2.0]

***pointer***

Visual indicator showing where pointing device input will occur. The indicator can be moved with a pointing device or emulator such as a mouse, pen tablet, keyboard-based mouse emulator, speech-based mouse commands, or 3-D wand. A pointing device click typically moves the [input focus](http://www.w3.org/WAI/UA/UAAG20/#def-input-focus) to the pointer location. The indicator may change to reflect different states.When touch screens are used, the "pointing device" is a combination of the touch screen and the user's finger or stylus. On most systems there is no pointer (on-screen visual indication) associated with this type of pointing device.

***pointing device focus***

The screen location where pointer input will occur if a viewport is active. There can be multiple pointing device foci for example when using a screen sharing utility there is typically one for the user's physical mouse and one for the remote mouse. @@ Editors' Note: this term is not used in the document other than the glossary.@@

***selection***

A user agent mechanism for identifying a (possibly empty) range of content that will be the implicit source or target for subsequent operations. The selection may be used for a variety of purposes, including for cut-and-paste operations, to designate a specific element in a document for the purposes of a query, and as an indication of [point of regard](http://www.w3.org/WAI/UA/UAAG20/#def-point-of-regard) (e.g. the matched results of a search may be automatically selected). The selection should be [highlighted](http://www.w3.org/WAI/UA/UAAG20/#def-highlight) in a distinctive manner. On the screen, the selection may be highlighted in a variety of ways, including through colors, fonts, graphics, and magnification. When rendered using synthesized speech, the selection may be highlighted through changes in pitch, speed, or prosody.

***split focus***

A state when the user could be confused because the input focus is separated from something it is usually linked to, such as being at a different place than the selection or similar highlighting, or has been scrolled outside of the visible portion of the viewport. @@ Editors' Note: this term is not used in the document other than the glossary.@@

***text cursor***

Indicator showing where keyboard input will occur in text (e.g. the flashing vertical bar in a text field, also called a caret).

**@@ Editor's Note: Need to find the hrefs to these definitions and fix them. @@**

***globally, global configuration***

a global setting is one that applies to the entire user agent or all content being rendered by it, rather than to a specific feature within the user agent or a specific document being viewed.

***graphical***

Information (e.g. text, colors, graphics, images, and animations) rendered for visual consumption.

***highlight, highlighted, highlighting***

see [***focus***](http://www.w3.org/WAI/UA/UAAG20/#def-highlight-focus)

***image***

Pictorial content that is static (i.e. not moving or changing). See also the definition of [animation](http://www.w3.org/WAI/UA/UAAG20/#def-animation).

***implement***

see [***support***](http://www.w3.org/WAI/UA/UAAG20/#def-support)

***important elements***

This specification intentionally does not identify which "important elements" must be navigable because this will vary by specification. What constitutes "efficient navigation" may depend on a number of factors as well, including the "shape" of content (e.g. sequential navigation of long lists is not efficient) and desired granularity (e.g. among tables, then among the cells of a given table). Refer to the Implementing document [Implementing UAAG 2.0] for information about identifying and navigating important elements.

***inactive input focus***

see [***focus***](http://www.w3.org/WAI/UA/UAAG20/#def-inactive-input-focus)

***inactive selection***

see [***focus***](http://www.w3.org/WAI/UA/UAAG20/#def-inactive-selection)

***informative (non-normative)***

see [***normative***](http://www.w3.org/WAI/UA/UAAG20/#def-informative)

***input configuration***

The set of bindings between user agent functionalities and [user interface](http://www.w3.org/WAI/UA/UAAG20/#def-user-interface) input mechanisms (e.g. menus, buttons, keyboard keys, and voice commands). The default input configuration is the set of bindings the user finds after installation of the software. Input configurations may be affected by author-specified bindings (e.g. through the accesskey attribute of HTML 4 [*[HTML4]*](http://www.w3.org/WAI/UA/UAAG20/#ref-HTML4)).

***input focus***

see [***focus***](http://www.w3.org/WAI/UA/UAAG20/#def-input-focus)

***keyboard (keyboard emulator, keyboard interface)***

The letter, symbol and command keys or key indicators that allow a user to control a computing device. Assistive technologies have traditionally relied on the keyboard interface as a universal, or modality independent interface. In this document references to keyboard include keyboard emulators and keyboard interfaces that make use of the keyboard's role as a modality independent interface (see [Modality Independence Principle](http://www.w3.org/WAI/UA/UAAG20/#intro-modality-independence)). Keyboard emulators and interfaces may be used on devices which do not have a physical keyboard, such as mobile devices based on touchscreen input.

***keyboard command* (*****keyboard binding*,*****keyboard shortcuts* or*****accelerator keys*)**

A key or set of keys that are tied to a particular UI control or application function, allowing the user to navigate to or activate the control or function without traversing any intervening controls (e.g. CTRL+"S" to save a document). It is sometimes useful to distinguish keyboard commands that are associated with controls that are rendered in the current context (e.g. ALT+"D" to move focus to the address bar) from those that may be able to activate program functionality that is not associated with any currently rendered controls (e.g. "F1" to open the Help system). Keyboard commands can be triggered using a physical keyboard or keyboard emulator (e.g. on-screen keyboard or speech recognition). (See [Modality Independence Principle](http://www.w3.org/WAI/UA/UAAG20/#intro-modality-independence)).

***keyboard focus***

see [***focus***](http://www.w3.org/WAI/UA/UAAG20/#def-keyboard-focus)

***natural language***

Natural language is spoken, written, or signed human language such as French, Japanese, and American Sign Language. On the Web, the natural language of [content](http://www.w3.org/WAI/UA/UAAG20/#def-content) may be specified by markup or HTTP headers. Some examples include the [langattribute in HTML 4](http://www.w3.org/TR/1999/REC-html401-19991224/struct/dirlang.html#adef-lang) ([*[HTML4]*](http://www.w3.org/WAI/UA/UAAG20/#ref-HTML4) section 8.1), the [xml:lang attribute in XML 1.0](http://www.w3.org/TR/1998/REC-xml-19980210#sec-lang-tag) ([*[XML]*](http://www.w3.org/WAI/UA/UAAG20/#ref-XML), section 2.12), the [hreflang attribute](http://www.w3.org/TR/1999/REC-html401-19991224/struct/links.html#adef-hreflang) for links in HTML 4 ([*[HTML4]*](http://www.w3.org/WAI/UA/UAAG20/#ref-HTML4), section 12.1.5), the HTTP Content-Language header ([*[RFC2616]*](http://www.w3.org/WAI/UA/UAAG20/#ref-RFC2616), section 14.12) and the Accept-Language request header ([*[RFC2616]*](http://www.w3.org/WAI/UA/UAAG20/#ref-RFC2616), section 14.4). See also the definition of [script](http://www.w3.org/WAI/UA/UAAG20/#def-script).

***navigation***

see [***command***](http://www.w3.org/WAI/UA/UAAG20/#def-command)

***non-text content (non-text element, non-text equivalent)***

see [***text***](http://www.w3.org/WAI/UA/UAAG20/#def-text)

***normative*,*****informative (non-normative)***[[WCAG 2.0](http://www.w3.org/TR/WCAG20/#glossary), ATAG 2.0]

What is identified as "normative" is required for [conformance](http://www.w3.org/WAI/UA/UAAG20/#conformance) (noting that one may conform in a variety of well-defined ways to UAAG 2.0). What is identified as "informative" (or, "non-normative") is never required for conformance.

***notify***

To make the user aware of events or status changes. Notifications can occur within the user agent user interface (e.g. a status bar) or within the content display. Notifications may be passive and not require user acknowledgment, or they may be presented in the form of a prompt requesting a user response (e.g. a confirmation dialog).

***operating system (OS)***

Software that supports a computer's basic functions, such as scheduling tasks, executing applications, and managing hardware and peripherals.   
**Note**: Many operating systems mediate communication between executing applications and assistive technology via a platform accessibility service.

***override***

In UAAG 2.0, the term "override" means that one configuration or behavior preference prevails over another. Generally, the requirements of UAAG 2.0 involve user preferences prevailing over author preferences and user agent default settings and behaviors. Preferences may be multi-valued in general (e.g. the user prefers blue over red or yellow), and include the special case of two values (e.g. turn on or off blinking text content).

***placeholder***

A placeholder is content generated by the user agent to replace author-supplied content. A placeholder may be generated as the result of a user preference (e.g. to not render images) or as [repair content](http://www.w3.org/WAI/UA/UAAG20/#def-repair-content) (e.g. when an image cannot be found). A placeholder can be any type of content, including text, images, and audio cues. A placeholder should identify the technology of the replaced object. Placeholders appear in the alternative content stack.

***platform***

The software environment(s) within which the authoring tool operates. Platforms provide a consistent operational environment. There may be layers of software in an hardware architecture and that each layer may be considered a platform. \*Non-web-based\* platforms include desktop [operating system](http://www.w3.org/WAI/UA/UAAG20/#def-operating-system) (e.g. Linux, MacOS, Windows, etc.), mobile operating systems (e.g. Android, Blackberry, iOS, Windows Phone, etc.), and cross-OS environments (e.g. Java). \*Web-based\* platforms are other user agents. User agents may employ server-based processing, such as web content transformations, text-to-speech production, etc..  
**Note 1**: A user agent may include functionality hosted on multiple platforms (e.g. a browser running on the desktop may include server-based pre-processing and web-based documentation).  
**Note 2**: Accessibility guidelines for developers exist for many platforms.

***platform accessibility service***

A programmatic interface that is engineered to enhance communication between mainstream software applications and assistive technologies (e.g. MSAA, UI Automation, and IAccessible2 for Windows applications, AXAPI for MacOSX applications, Gnome Accessibility Toolkit API for Gnome applications, Java Access for Java applications). On some platforms it may be conventional to enhance communication further via implementing a [DOM](http://www.w3.org/WAI/UA/UAAG20/#def-dom).

***plug-in***[ATAG 2.0]

A plug-in is a program that runs as part of the user agent and that is *not* part of [content](http://www.w3.org/WAI/UA/UAAG20/#def-content). Users generally choose to include or exclude plug-ins from their user agents.

***point of regard***

The point of regard is the position in [rendered content](http://www.w3.org/WAI/UA/UAAG20/#def-rendered-content) that the user is presumed to be viewing. The dimensions of the point of regard may vary. For example,it may be a two-dimensional area (e.g. content rendered through a two-dimensional graphical viewport), or a point (e.g. a moment during an audio rendering or a cursor position in a graphical rendering), or a range of text (e.g. focused text), or a two-dimensional area (e.g. content rendered through a two-dimensional graphical viewport). The point of regard is almost always within the viewport, but it may exceed the spatial or temporal [dimensions](http://www.w3.org/WAI/UA/UAAG20/#viewport-dimension) of the viewport (see the definition of [rendered content](http://www.w3.org/WAI/UA/UAAG20/#def-rendered-content) for more information about viewport dimensions). The point of regard may also refer to a particular moment in time for content that changes over time (e.g. an audio-only presentation). User agents may determine the point of regard in a number of ways, including based on viewport position in content, [keyboard focus](http://www.w3.org/WAI/UA/UAAG20/#def-content-focus), and [selection](http://www.w3.org/WAI/UA/UAAG20/#def-selection). The stability of the point of regard is addressed by [success criterion 1.8.7](http://www.w3.org/WAI/UA/UAAG20/#sc-187)

***pointer***

see [***focus***](http://www.w3.org/WAI/UA/UAAG20/#def-pointer)

***pointing device focus***

see [***focus***](http://www.w3.org/WAI/UA/UAAG20/#def-pointing-device-focus)

***profile***

A profile is a named and persistent representation of user preferences that may be used to configure a user agent. Preferences include input configurations, style preferences, and natural language preferences. In [operating environments](http://www.w3.org/WAI/UA/UAAG20/#def-operating-environment) with distinct user accounts, profiles enable users to reconfigure software quickly when they log on. Users may share their profiles with one another.Platform-independent profiles are useful for those who use the same user agent on different devices.

***prompt***[ATAG 2.0]

Any user-agent-initiated request for a decision or piece of information from a user.

***properties, values, and defaults***

A user agent renders a document by applying formatting algorithms and style information to the document's elements. Formatting depends on a number of factors, including where the document is rendered (e.g. on screen, on paper, through loudspeakers, on a braille display, on a mobile device). Style information (e.g. fonts, colors, synthesized speech prosody) may come from the elements themselves (e.g. certain font and phrase elements in HTML), from style sheets, or from user agent settings. For the purposes of these guidelines, each formatting or style option is governed by a property and each property may take one value from a set of legal values. Generally in UAAG 2.0, the term "[property](http://www.w3.org/TR/1998/REC-CSS2-19980512/conform.html)" has the meaning defined in CSS 2 ([*[CSS2]*](http://www.w3.org/WAI/UA/UAAG20/#ref-CSS2), section 3). A reference to "styles" in UAAG 2.0 means a set of style-related properties. The value given to a property by a user agent at installation is the property's default value.

***recognize***

Authors encode information in many ways, including in markup languages, style sheet languages, scripting languages, and protocols. When the information is encoded in a manner that allows the user agent to process it with certainty, the user agent can "recognize" the information. For instance, HTML allows authors to specify a heading with the H1 element, so a user agent that implements HTML can recognize that content as a heading. If the author creates a heading using a visual effect alone (e.g. just by increasing the font size), then the author has encoded the heading in a manner that does not allow the user agent to recognize it as a heading. Some requirements of UAAG 2.0 depend on content roles, content relationships, timing relationships, and other information supplied by the author. These requirements only apply when the author has encoded that information in a manner that the user agent can recognize. See the section on [conformance](http://www.w3.org/WAI/UA/UAAG20/#conformance) for more information about applicability. User agents will rely heavily on information that the author has encoded in a markup language or style sheet language. Behaviors, style, meaning encoded in a [script](http://www.w3.org/WAI/UA/UAAG20/#def-script), and markup in an unfamiliar XML namespace may not be recognized by the user agent as easily or at all.

***relative time units***

Relative time units define time intervals for navigating media relative to the current point (e.g. move forward 30 seconds). When interacting with a time-based media presentation, a user may find it beneficial to move forward or backward via a time interval relative to their current position. For example, a user may find a concept unclear in a video lecture and elect to skip back 30 seconds from the current position to review what had been described. Relative time units may be preset by the user agent, configurable by the user, and/or automatically calculated based upon media duration (e.g. jump 5 seconds in a 30-second clip, or 5 minutes in a 60-minute clip). Relative time units are distinct from absolute time values such as the 2 minute mark, the half-way point, or the end.

***rendered content*,*****rendered text***

Rendered content is the part of [content](http://www.w3.org/WAI/UA/UAAG20/#def-content) that the user agent makes available to the user's senses of sight and hearing (and only those senses for the purposes of UAAG 2.0). Any content that causes an effect that may be perceived through these senses constitutes rendered content. This includes text characters, images, style sheets, scripts, and any other content that, once processed, may be perceived through sight and hearing.

The term "rendered text" refers to [text](http://www.w3.org/WAI/UA/UAAG20/#def-text) content that is rendered in a way that communicates information about the characters themselves, whether visually or as synthesized speech.

In the context of UAAG 2.0, ***invisible content*** is content that is not rendered but that may influence the graphical rendering (i.e. layout) of other content. Similarly, ***silent content*** is content that is not rendered but that may influence the audio rendering of other content. Neither invisible nor silent content is considered rendered content.

***repair content*,*****repair text***

Content generated by the user agent to correct an error condition. "Repair text" refers to the [text](http://www.w3.org/WAI/UA/UAAG20/#def-text) portion of repair content. Error conditions that may lead to the generation of repair content include:

* Erroneous or incomplete content (e.g. ill-formed markup, invalid markup, or missing [alternative content](http://www.w3.org/WAI/UA/UAAG20/#def-conditional-content) that is required by format specification);
* Missing resources for handling or rendering content (e.g. the user agent lacks a font family to display some characters, or the user agent does not implement a particular scripting language).

UAAG 2.0 does not require user agents to include repair content in the [document object](http://www.w3.org/WAI/UA/UAAG20/#def-document-object). Repair content inserted in the document object should conform to the Web Content Accessibility Guidelines 1.0 [*[WCAG10]*](http://www.w3.org/WAI/UA/UAAG20/#ref-WCAG10). For more information about repair techniques for Web content and software, refer to "Techniques for Authoring Tool Accessibility Guidelines 1.0" [*[ATAG10-TECHS]*](http://www.w3.org/WAI/UA/UAAG20/#ref-ATAG10-TECHS).

***script***

In UAAG 2.0, the term "script" almost always refers to a scripting (programming) language used to create dynamic Web content. However, in guidelines referring to the written (natural) language of content, the term "script" is used as in Unicode [*[UNICODE]*](http://www.w3.org/WAI/UA/UAAG20/#ref-UNICODE)to mean "A collection of symbols used to represent textual information in one or more writing systems."

Information encoded in (programming) scripts may be difficult for a user agent to [recognize](http://www.w3.org/WAI/UA/UAAG20/#def-recognize). For instance, a user agent is not expected to recognize that, when executed, a script will calculate a factorial. The user agent will be able to recognize some information in a script by virtue of implementing the scripting language or a known program library (e.g. the user agent is expected to recognize when a script will open a viewport or retrieve a resource from the Web).

***selection*,*****current selection***

see [***focus***](http://www.w3.org/WAI/UA/UAAG20/#def-selection)

***style properties*,*****user agent default styles*,*****author styles*,*****user styles***

Properties whose values determine the presentation (e.g., font, color, size, location, padding, volume, synthesized speech prosody) of content elements as they are rendered (e.g. onscreen, via loudspeaker, via braille display) by user agents. Style properties can have several origins:   
***user agent default styles***: The default style property values applied in the absence of any author or user styles. Some [web content technologies](http://www.w3.org/WAI/UA/UAAG20/#def-Web-Content-Technology) specify a default rendering; others do not.   
***author styles***: Style property values that are set by the author as part of the content (e.g., in-line styles, author style sheets).   
***user styles***: Style property values that are set by the user (e.g., via user agent interface settings, user style sheets).

***style sheet*,*****user style sheet*,*****author style sheet***

A mechanism for communicating [style property](http://www.w3.org/WAI/UA/UAAG20/#def-style-properties) settings for web content, in which the style property settings are separable from other content resources. This separation is what allows author style sheets to be toggled or substituted, and user style sheets defined to apply to more than one resource. Style sheet [web content technologies](http://www.w3.org/WAI/UA/UAAG20/#def-Web-Content-Technology) include Cascading Style Sheets (CSS) and Extensible Stylesheet Language (XSL). ***User style sheet***: Style sheets specified by the user, resulting in [user styles](http://www.w3.org/WAI/UA/UAAG20/#def-user-styles). ***Author style sheet***: Style sheets specified by the author, resulting in [author styles](http://www.w3.org/WAI/UA/UAAG20/#def-author-styles).

***support*,*****implement*,*****conform***

Support, implement, and conform all refer to what a developer has designed a user agent to do, but they represent different degrees of specificity. A user agent "supports" general classes of objects, such as "images" or "Japanese." A user agent "implements" a specification (e.g. the PNG and SVG image format specifications or a particular scripting language), or an [API](http://www.w3.org/WAI/UA/UAAG20/#def-api)(e.g. the DOM API) when it has been programmed to follow all or part of a specification. A user agent "conforms to" a specification when it implements the specification *and* satisfies its conformance criteria.

***synchronize***

The act of time-coordinating two or more presentation components (e.g. a [visual track](http://www.w3.org/WAI/UA/UAAG20/#def-visual-track) with captions, or several tracks in a multimedia presentation). For Web content developers, the requirement to synchronize means to provide the data that will permit sensible time-coordinated rendering by a user agent. For example, Web content developers can ensure that the segments of caption text are neither too long nor too short, and that they map to segments of the visual track that are appropriate in length. For user agent developers, the requirement to synchronize means to present the content in a sensible time-coordinated fashion under a wide range of circumstances including technology constraints (e.g. small text-only displays), user limitations (e.g. slow reading speeds, large font sizes, high need for review or repeat functions), and content that is sub-optimal in terms of accessibility.

***technology (web content technology)***[[WCAG 2.0](http://www.w3.org/TR/WCAG20/#glossary), ATAG 2.0]

A mechanism for encoding instructions to be rendered, played or executed by [user agents](http://www.w3.org/WAI/UA/UAAG20/#def-user-agent). Web Content technologies may include markup languages, data formats, or programming languages that [authors](http://www.w3.org/WAI/UA/UAAG20/#def-author) may use alone or in combination to create end-user experiences that range from static Web pages to multimedia presentations to dynamic Web applications. Some common examples of Web content technologies include HTML, CSS, SVG, PNG, PDF, Flash, and JavaScript.

***text* (*text content*, *non-text content*, *text element*, *non-text element*, *text equivalent*, *non-text equivalent*)**

Text used by itself refers to a sequence of characters from a markup language's [document character set](http://www.w3.org/WAI/UA/UAAG20/#def-doc-char-set). Refer to the "Character Model for the World Wide Web" [*[CHARMOD]*](http://www.w3.org/WAI/UA/UAAG20/#ref-CHARMOD) for more information about text and characters. **Note:** UAAG 2.0 makes use of other terms that include the word "text" that have highly specialized meanings: [collated text transcript](http://www.w3.org/WAI/UA/UAAG20/#def-collated-text-transcript), [non-text content](http://www.w3.org/WAI/UA/UAAG20/#def-non-text-content), [text content](http://www.w3.org/WAI/UA/UAAG20/#def-text-content), [non-text element](http://www.w3.org/WAI/UA/UAAG20/#def-non-text-element), [text element](http://www.w3.org/WAI/UA/UAAG20/#def-text-element), [text equivalent](http://www.w3.org/WAI/UA/UAAG20/#def-text-eq), and [text transcript](http://www.w3.org/WAI/UA/UAAG20/#def-text-transcript).

A***text element*** adds [text characters](http://www.w3.org/WAI/UA/UAAG20/#def-text) to either [content](http://www.w3.org/WAI/UA/UAAG20/#def-content) or the [user interface](http://www.w3.org/WAI/UA/UAAG20/#def-user-interface). Both in the Web Content Accessibility Guidelines 2.0[*[WCAG20]*](http://www.w3.org/WAI/UA/UAAG20/#ref-WCAG20) and in UAAG 2.0, text elements are presumed to produce text that can be understood when rendered visually, as synthesized speech, or as Braille. Such text elements benefit at least these three groups of users:

1. visually-displayed text benefits users who are deaf and adept in reading visually-displayed text;
2. synthesized speech benefits users who are blind and adept in use of synthesized speech;
3. braille benefits users who are blind, and possibly deaf-blind, and adept at reading braille.

A text element may consist of both text and non-text data. For instance, a text element may contain markup for style (e.g. font size or color), structure (e.g. heading levels), and other semantics. The essential function of the text element should be retained even if style information happens to be lost in rendering. A user agent may have to process a text element in order to have access to the text characters. For instance, a text element may consist of markup, it may be encrypted or compressed, or it may include embedded text in a binary format (e.g. JPEG).

***Text content*** is content that is composed of one or more text elements. A ***text equivalent*** (whether in content or the user interface) is an [equivalent](http://www.w3.org/WAI/UA/UAAG20/#def-equivalent) composed of one or more text elements. Authors generally provide text equivalents for content by using the [alternative content](http://www.w3.org/WAI/UA/UAAG20/#def-conditional-content) mechanisms of a specification.

A ***non-text element*** is an element (in content or the user interface) that does not have the qualities of a text element. ***Non-text content*** is composed of one or more non-text elements. A ***non-text equivalent*** (whether in content or the user interface) is an[equivalent](http://www.w3.org/WAI/UA/UAAG20/#def-equivalent) composed of one or more non-text elements.

***text decoration***

Any stylistic effect that the user agent may apply to visually [rendered text](http://www.w3.org/WAI/UA/UAAG20/#def-rendered-text) that does not affect the layout of the document (i.e. does not require reformatting when applied or removed). Text decoration mechanisms include underline, overline, and strike-through.

***text format***

Any media object given an Internet media type of "text" (e.g. "text/plain", "text/html", or "text/\*") as defined in RFC 2046[*[RFC2046]*](http://www.w3.org/WAI/UA/UAAG20/#ref-RFC2046), section 4.1, or any media object identified by Internet media type to be an XML document (as defined in [*[XML]*](http://www.w3.org/WAI/UA/UAAG20/#ref-XML), section 2) or SGML application. Refer, for example, to Internet media types defined in "XML Media Types" [*[RFC3023]*](http://www.w3.org/WAI/UA/UAAG20/#ref-RFC3023).

***text transcript***

A text equivalent of audio information (e.g. an audio-only presentation or the [audio track](http://www.w3.org/WAI/UA/UAAG20/#def-audio-track) of a movie or other animation). A text transcript provides text for both spoken words and non-spoken sounds such as sound effects. Text transcripts make audio information accessible to people who have hearing disabilities and to people who cannot play the audio. Text transcripts are usually created by hand but may be generated on the fly (e.g. by voice-to-text converters). See also the definitions of [captions](http://www.w3.org/WAI/UA/UAAG20/#def-captions)and [collated text transcripts](http://www.w3.org/WAI/UA/UAAG20/#def-collated-text-transcript).

***time-based***

defining a common time scale for all components of a time-based media presentation. For example, a media-player will expose a single timebase for a presentation composed of individual video and audio tracks, for instance allowing users or technology to query or alter the playback rate and position.

***track* (*audio track* or**[**visual track**](http://www.w3.org/WAI/UA/UAAG20/#def-visual-track)**)**

[Content](http://www.w3.org/WAI/UA/UAAG20/#def-content) rendered as sound through an audio [viewport](http://www.w3.org/WAI/UA/UAAG20/#def-viewport). The audio track may be all or part of the audio portion presentation (e.g. each instrument may have a track, or each stereo channel may have a track). Also see definition of [visual track](http://www.w3.org/WAI/UA/UAAG20/#def-visual-track)

***toolbar***

A collection of commonly used controls presented in a region that can be configured or navigated separately from other regions. Such containers may be docked or free-floating, permanent or transient, integral to the application or add-ons. Variations are often called toolbars, palettes, panels, or inspectors.

***user agent***

A user agent is any software that retrieves, renders and facilitates end user interaction with Web content.

***user agent extension (add-in)***

Software installed into a user agent that adds one or more additional features that modify the behavior of the user agent. Two common capabilities for user agent extensions are the ability to \*modify the content\* before the user agent renders it (e.g., to add highlights if certain types of alternative content are present) and to \*modify the user agent's own user interface \* (e.g. add a headings view).

***user agent default styles***

User agent default styles are [style property values](http://www.w3.org/WAI/UA/UAAG20/#def-Properties-and-Values) applied in the absence of any author or user styles. Some markup languages specify a default rendering for content in that markup language; others do not. For example, XML 1.0 [*[XML]*](http://www.w3.org/WAI/UA/UAAG20/#ref-XML) does not specify default styles for XML documents. HTML 4 [*[HTML4]*](http://www.w3.org/WAI/UA/UAAG20/#ref-HTML4) does not specify default styles for HTML documents, but the CSS 2 [*[CSS2]*](http://www.w3.org/WAI/UA/UAAG20/#ref-CSS2)specification suggests a [sample default style sheet for HTML 4](http://www.w3.org/TR/1998/REC-CSS2-19980512/sample.html) based on current practice.

***user interface*,*****user interface control***

For the purposes of UAAG 2.0, user interface includes both:

1. the ***user agent user interface***, i.e. the controls (e.g. menus, buttons, prompts, and other components for input and output) and mechanisms (e.g. selection and focus) provided by the user agent ("out of the box") that are not created by [content](http://www.w3.org/WAI/UA/UAAG20/#def-content).
2. the "content user interface," i.e. the [enabled elements](http://www.w3.org/WAI/UA/UAAG20/#def-enabled-element) that are part of content, such as form controls, links, and applets.

The document distinguishes them only where required for clarity. For more information, see the section on requirements for content, for user agent features, or both @@.

The term "user interface control" refers to a component of the user agent user interface or the content user interface, distinguished where necessary.

***user styles***

User styles are [style property values](http://www.w3.org/WAI/UA/UAAG20/#def-Properties-and-Values) that come from user interface settings, user style sheets, or other user interactions.

***values***

see [***properties***](http://www.w3.org/WAI/UA/UAAG20/#def-Properties-and-Values)

***view***

A user interface function that lets users interact with web content. UAAG 2.0 recognizes a variety of approaches to presenting the content in a view, such as:

* ***rendered view***: Views in which content is presented such that it is rendered, played or executed. There are several sub-types:
  + In conventionally rendered views the content is rendered, played or executed according to the web content technology specification. This is the default view of most user agents.
  + In unconventionally rendered views the content is rendered quite differently than specified in the technology specification (e.g., rendering an audio file as a graphical wavefront); or
* ***source view***: Views in which the web content is presented without being rendered, played or executed. The source view may be plain text (i.e., "View Source") or it may include some other organization (e.g., presenting the markup in a tree).
* ***outline view***: Views in which only a subset of the rendered content is presented, usually composed of labels or placeholders for important structural elements. The important structural elements will depend on the web content technology, but may include headings, table captions, and content sections. Note: Views can be visual, audio, or tactile.  
  ***top-level viewports*** are viewports that are not contained within other user agent viewports.

***viewport***

The part of an onscreen view that the user agent is currently presenting onscreen to the user, such that the user can attend to any part of it without further action (e.g. scrolling). There may be multiple viewports on to the same view (e.g. when a split-screen is used to present the top and bottom of a document simultaneously) and viewports may be nested (e.g. a scrolling frame located within a larger document). When the viewport is smaller in extent than the content it is presenting, user agents typically provide mechanisms to bring the occluded content into the viewport (e.g., scrollbars).

***visual-only***

A visual-only presentation is content consisting exclusively of one or more [visual tracks](http://www.w3.org/WAI/UA/UAAG20/#def-visual-track) presented concurrently or in series. A silent movie is an example of a visual-only presentation.

***visual track***

A visual object is content rendered through a graphical [viewport](http://www.w3.org/WAI/UA/UAAG20/#def-viewport). Visual objects include graphics, text, and visual portions of movies and other animations. A visual track is a visual object that is intended as a whole or partial presentation. A visual track does not necessarily correspond to a single physical object or software object.

***voice browser***

From "Introduction and Overview of W3C Speech Interface Framework" [*[VOICEBROWSER]*](http://www.w3.org/WAI/UA/UAAG20/#ref-VOICEBROWSER): "A voice browser is a device (hardware and software) that interprets voice markup languages to generate voice output, interpret voice input, and possibly accept and produce other modalities of input and output."

***web resource***

Anything that can be identified by a Uniform Resource Identifier (URI).

Definitions No Longer Used (to be removed before publishing)

***operating environment***

The term "operating environment" refers to the environment that governs the user agent's operation, whether it is an [operating system](http://www.w3.org/WAI/UA/UAAG20/#def-operating-system) or a programming language environment such as Java.

Appendix B: How to refer to UAAG 2.0 from other documents

This section is [informative](http://www.w3.org/WAI/UA/UAAG20/#def-informative).

There are two recommended ways to refer to the "User Agent Accessibility Guidelines 2.0" (and to W3C documents in general):

1. References to a specific version of "User Agent Accessibility Guidelines 2.0." For example, use the "this version" URI to refer to the current document:  
   <http://www.w3.org/TR/2010/WD-UAAG20-20100617/>
2. References to the latest version of "User Agent Accessibility Guidelines 2.0." Use the "latest version" URI to refer to the most recently published document in the series:   
   <http://www.w3.org/TR/UAAG20/>.

In almost all cases, references (either by name or by link) should be to a specific version of the document. W3C will make every effort to make UAAG 2.0 indefinitely available at its original address in its original form. The top of UAAG 2.0 includes the relevant catalog metadata for specific references (including title, publication date, "this version" URI, editors' names, and copyright information).

An XHTML 1.0 paragraph including a reference to this specific document might be written:

<p>  
<cite><a href="http://www.w3.org/TR/2010/WD-UAAG20-20100617/">  
"User Agent Accessibility Guidelines 2.0,"</a></cite>  
J. Allan, K. Ford, J. Spellman, eds.,  
W3C Recommendation, http://www.w3.org/TR/ATAG20/.  
The <a href="http://www.w3.org/TR/ATAG20/">latest version</a> of this document is available at http://www.w3.org/TR/ATAG20/.</p>

For very general references to this document (where stability of content and anchors is not required), it may be appropriate to refer to the latest version of this document. Other sections of this document explain how to build a [conformance claim](http://www.w3.org/WAI/UA/UAAG20/#conformance).

Appendix C: References

@@ add [Evaluating Web Sites for Accessibility: Overview](http://www.w3.org/WAI/eval/Overview.html) @@

This section is [informative](http://www.w3.org/WAI/UA/UAAG20/#def-informative).

For the **latest version** of any W3C specification please consult the list of [W3C Technical Reports](http://www.w3.org/TR/) at http://www.w3.org/TR/. Some documents listed below may have been superseded since the publication of UAAG 2.0.

**Note:** In UAAG 2.0, bracketed labels such as "[WCAG20]" link to the corresponding entries in this section. These labels are also identified as references through markup.

**[CSS1]**

[*"Cascading Style Sheets (CSS1) Level 1 Specification,"*](http://www.w3.org/TR/1999/REC-CSS1-19990111) B. Bos, H. Wium Lie, eds., 17 December 1996, revised 11 January 1999. This W3C Recommendation is http://www.w3.org/TR/1999/REC-CSS1-19990111.

**[CSS2]**

[*"Cascading Style Sheets, level 2 (CSS2) Specification,"*](http://www.w3.org/TR/1998/REC-CSS2-19980512/) B. Bos, H. Wium Lie, C. Lilley, and I. Jacobs, eds., 12 May 1998. This W3C Recommendation is http://www.w3.org/TR/1998/REC-CSS2-19980512/.

**[DOM2CORE]**

[*"Document Object Model (DOM) Level 2 Core Specification,"*](http://www.w3.org/TR/2000/REC-DOM-Level-2-Core-20001113/) A. Le Hors, P. Le Hégaret, L. Wood, G. Nicol, J. Robie, M. Champion, S. Byrne, eds., 13 November 2000. This W3C Recommendation is http://www.w3.org/TR/2000/REC-DOM-Level-2-Core-20001113/.

**[DOM2STYLE]**

[*"Document Object Model (DOM) Level 2 Style Specification,"*](http://www.w3.org/TR/2000/REC-DOM-Level-2-Style-20001113/) V. Apparao, P. Le Hégaret, C. Wilson, eds., 13 November 2000. This W3C Recommendation is http://www.w3.org/TR/2000/REC-DOM-Level-2-Style-20001113/.

**[INFOSET]**

[*"XML Information Set,"*](http://www.w3.org/TR/2001/REC-xml-infoset-20011024/) J. Cowan and R. Tobin, eds., 24 October 2001. This W3C Recommendation is http://www.w3.org/TR/2001/REC-xml-infoset-20011024/.

**[RFC2046]**

[*"Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types,"*](http://www.ietf.org/rfc/rfc2046.txt) N. Freed, N. Borenstein, November 1996.

**[WCAG10]**

[*"Web Content Accessibility Guidelines 1.0,"*](http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505/) W. Chisholm, G. Vanderheiden, and I. Jacobs, eds., 5 May 1999. This W3C Recommendation is http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505/.

**[XML]**

[*"Extensible Markup Language (XML) 1.0 (Second Edition),"*](http://www.w3.org/TR/1998/REC-xml-19980210) T. Bray, J. Paoli, C.M. Sperberg-McQueen, eds., 6 October 2000. This W3C Recommendation is http://www.w3.org/TR/2000/REC-xml-20001006.

**[AT1998]**

The [*Assistive Technology Act of 1998*](http://www.section508.gov/508Awareness/html/at1998.html).

**[ATAG10]**

[*"Authoring Tool Accessibility Guidelines 1.0,"*](http://www.w3.org/TR/2000/REC-ATAG10-20000203/) J. Treviranus, C. McCathieNevile, I. Jacobs, and J. Richards, eds., 3 February 2000. This W3C Recommendation is http://www.w3.org/TR/2000/REC-ATAG10-20000203/.

**[ATAG10-TECHS]**

[*"Techniques for Authoring Tool Accessibility Guidelines 1.0,"*](http://www.w3.org/TR/2002/NOTE-ATAG10-TECHS-20021029/) J. Treviranus, C. McCathieNevile, J. Richards, eds., 29 Oct 2002. This W3C Note is http://www.w3.org/TR/2002/NOTE-ATAG10-TECHS-20021029/.

**[CHARMOD]**

[*"Character Model for the World Wide Web,"*](http://www.w3.org/TR/2002/WD-charmod-20020430/) M. Dürst and F. Yergeau, eds., 30 April 2002. This W3C Working Draft is http://www.w3.org/TR/2002/WD-charmod-20020430/. The [latest version](http://www.w3.org/TR/charmod/) is available at http://www.w3.org/TR/charmod/.

**[DOM2HTML]**

[*"Document Object Model (DOM) Level 2 HTML Specification,"*](http://www.w3.org/TR/2002/PR-DOM-Level-2-HTML-20021108/) J. Stenback, P. Le Hégaret, A. Le Hors, eds., 8 November 2002. This W3C Proposed Recommendation is http://www.w3.org/TR/2002/PR-DOM-Level-2-HTML-20021108/. The [latest version](http://www.w3.org/TR/DOM-Level-2-HTML/) is available at http://www.w3.org/TR/DOM-Level-2-HTML/.

**[HTML4]**

[*"HTML 4.01 Recommendation,"*](http://www.w3.org/TR/1999/REC-html401-19991224/) D. Raggett, A. Le Hors, and I. Jacobs, eds., 24 December 1999. This W3C Recommendation is http://www.w3.org/TR/1999/REC-html401-19991224/.

**[RFC2616]**

[*"Hypertext Transfer Protocol — HTTP/1.1,"*](http://www.ietf.org/rfc/rfc2616.txt) J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, June 1999.

**[RFC3023]**

[*"XML Media Types,"*](http://www.ietf.org/rfc/rfc3023.txt) M. Murata, S. St. Laurent, D. Kohn, January 2001.

**[SMIL]**

[*"Synchronized Multimedia Integration Language (SMIL) 1.0 Specification,"*](http://www.w3.org/TR/1998/REC-smil-19980615/) P. Hoschka, ed., 15 June 1998. This W3C Recommendation is http://www.w3.org/TR/1998/REC-smil-19980615/.

**[SMIL20]**

[*"Synchronized Multimedia Integration Language (SMIL 2.0) Specification,"*](http://www.w3.org/TR/2001/REC-smil20-20010807/) J. Ayars, et al., eds., 7 August 2001. This W3C Recommendation is http://www.w3.org/TR/2001/REC-smil20-20010807/.

**[SVG]**

[*"Scalable Vector Graphics (SVG) 1.0 Specification,"*](http://www.w3.org/TR/2001/REC-SVG-20010904/) J. Ferraiolo, ed., 4 September 2001. This W3C Recommendation is http://www.w3.org/TR/2001/REC-SVG-20010904/.

**[UAAG10]**

"[User Agent Accessibility Guidelines 1.0](http://www.w3.org/TR/2002/REC-UAAG10-20021217/)," I. Jacobs, J. Gunderson, E. Hansen, eds.17 December 2002. This W3C Recommendation is available at http://www.w3.org/TR/2002/REC-UAAG10-20021217/.

**[UAAG10-CHECKLIST]**

An appendix to UAAG 2.0 lists all of the checkpoints, sorted by priority. The checklist is available in either [tabular form](http://www.w3.org/TR/2002/REC-UAAG10-20021217/uaag10-chktable) or [list form](http://www.w3.org/TR/2002/REC-UAAG10-20021217/uaag10-chklist).

**[UAAG10-ICONS]**

Information about [UAAG 1.0 conformance icons](http://www.w3.org/WAI/UAAG10-Conformance) and their usage is available at http://www.w3.org/WAI/UAAG10-Conformance.

**[UAAG10-SUMMARY]**

An appendix to UAAG 2.0 provides a [summary](http://www.w3.org/TR/2002/REC-UAAG10-20021217/uaag10-summary) of the goals and structure of User Agent Accessibility Guidelines 1.0.

**[UAAG10-TECHS]**

[*"Techniques for User Agent Accessibility Guidelines 1.0,"*](http://www.w3.org/TR/UAAG10-TECHS/) I. Jacobs, J. Gunderson, E. Hansen, eds. The latest draft of the techniques document is available at http://www.w3.org/TR/UAAG10-TECHS/.

**[UNICODE]**

The Unicode Consortium. [*The Unicode Standard, Version 6.1.0*](http://www.unicode.org/versions/Unicode6.1.0/), (Mountain View, CA: The Unicode Consortium, 2012. ISBN 978-1-936213-02-3)   
<http://www.unicode.org/versions/Unicode6.1.0/>

**[VOICEBROWSER]**

[*"Introduction and Overview of W3C Speech Interface Framework,"*](http://www.w3.org/TR/2000/WD-voice-intro-20001204/) J. Larson, 4 December 2000. This W3C Working Draft is http://www.w3.org/TR/2000/WD-voice-intro-20001204/. The [latest version](http://www.w3.org/TR/voice-intro/) is available at http://www.w3.org/TR/voice-intro/. UAAG 2.0 includes references to additional W3C specifications about voice browser technology.

**[W3CPROCESS]**

[*"World Wide Web Consortium Process Document,"*](http://www.w3.org/Consortium/Process-20010719/) I. Jacobs ed. The 19 July 2001 version of the Process Document is http://www.w3.org/Consortium/Process-20010719/. The [latest version](http://www.w3.org/Consortium/Process/) is available at http://www.w3.org/Consortium/Process/.

**[WCAG20]**

[*"Web Content Accessibility Guidelines (WCAG) 2.0"*](http://www.w3.org/TR/WCAG20/) B. Caldwell, M. Cooper, L. Guarino Reid, G. Vanderheiden, eds., 8 December 2008. This W3C Recommendation is http://www.w3.org/TR/2008/REC-WCAG20-20081211/. The [latest version](http://www.w3.org/TR/WCAG20/) is available at http://www.w3.org/TR/WCAG20/. Additional format-specific techniques documents are available from this Recommendation.

**[WCAG20-TECHS]**

[*"Techniques for Web Content Accessibility Guidelines 2.0,"*](http://www.w3.org/TR/WCAG10-TECHS/) B. Caldwell, M. Cooper, L. Guarino Reid, G. Vanderheiden, eds., 8 December 2008. This W3C Note is http://www.w3.org/TR/2010/NOTE-WCAG20-TECHS-20101014/. The [latest version](http://www.w3.org/TR/WCAG20-TECHS/) is available at http://www.w3.org/TR/WCAG20-TECHS/. Additional format-specific techniques documents are available from this Note.

**[WEBCHAR]**

[*"Web Characterization Terminology and Definitions Sheet,"*](http://www.w3.org/1999/05/WCA-terms/01) B. Lavoie, H. F. Nielsen, eds., 24 May 1999. This is a W3C Working Draft that defines some terms to establish a common understanding about key Web concepts. This W3C Working Draft is http://www.w3.org/1999/05/WCA-terms/01.

**[XAG10]**

[*"XML Accessibility Guidelines 1.0,"*](http://www.w3.org/TR/2002/WD-xag-20021003) D. Dardailler, S. Palmer, C. McCathieNevile, eds., 3 October 2001. This W3C Working Draft is http://www.w3.org/TR/2002/WD-xag-20021003. The [latest version](http://www.w3.org/TR/xag) is available at http://www.w3.org/TR/xag.

**[XHTML10]**

[*"XHTML[tm] 1.0: The Extensible HyperText Markup Language,"*](http://www.w3.org/TR/2000/REC-xhtml1-20000126/) S. Pemberton, et al., 26 January 2000. This W3C Recommendation is http://www.w3.org/TR/2000/REC-xhtml1-20000126/.

**[XMLDSIG]**

[*"XML-Signature Syntax and Processing,"*](http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/) D. Eastlake, J. Reagle, D. Solo, eds., 12 February 2002. This W3C Recommendation is http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/.

**[XMLENC]**

[*"XML Encryption Syntax and Processing,"*](http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/) D. Eastlake, J. Reagle, eds., 10 December 2002. This W3C Recommendation is http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/.

Appendix D: Acknowledgments

Participants active in the UAWG prior publication:

* Jim Allan (Co-Chair, Texas School for the Blind and Visually Impaired)
* Wayne Dick (Invited Expert)
* Kelly Ford (Co-Chair, Microsoft)
* Mark Hakkinen (Invited Expert)
* Simon Harper (University of Manchester)
* Greg Lowney (Invited Expert)
* Kimberly Patch (Invited Expert)
* Jan Richards (Adaptive Technology Resource Centre, University of Toronto)
* Jeanne Spellman (W3C Staff Contact)

Other previously active UAWG participants and other contributors to UAAG 2.0:

* Judy Brewer (W3C)
* Alan Cantor (Invited Expert)
* Bim Egan (Royal National Institute of Blind People)
* Sean Hayes (Microsoft)
* Dean Hudson (Apple)
* Patrick Lauke (Opera Software)
* Cathy Laws (IBM)
* Peter Parente (IBM)
* David Poehlman (Invited Expert)
* Simon Pieters (Opera Software)
* Henny Swan (Opera Software)
* Gregory Rosmaita (Invited Expert)
* David Tseng (Apple)

UAAG 2.0 would not have been possible without the work of [those who contributed to UAAG 1.0](http://www.w3.org/TR/WAI-USERAGENT/acks.html#Acknowledgments).

This publication has been funded in part with Federal funds from the U.S. Department of Education, National Institute on Disability and Rehabilitation Research (NIDRR) under contract number ED-OSE-10-C-0067. The content of this publication does not necessarily reflect the views or policies of the U.S. Department of Education, nor does mention of trade names, commercial products, or organizations imply endorsement by the U.S. Government.

Appendix E: Checklist

@@ Editors' Note: This section is still under development@@

Appendix F: Comparison of UAAG 1.0 guidelines to UAAG 2.0

@@ Editors' Note: This section is still under development@@