Guidelines for Designing and Developing Accessible Web Sites by Using Microsoft Solutions

Version 2.0

Published: January 2007

Summary

This white paper is intended to help readers better understand the issues involved in making Web sites accessible to people with disabilities, a legal obligation that also benefits all users of the Web. We will deal with the current issues in accessibility, as well as the various initiatives, norms, and standards implemented with respect to Web-enabled applications. In this regard, we will examine using Microsoft products and technologies to design and develop Web sites that comply with accessibility guidelines.

More specifically, this document describes a set of guidelines that enable you to produce accessible Web sites by using the new Microsoft Office SharePoint Server (MOSS) 2007 technologies as part of Microsoft Office system 2007 or previous solutions such as Content Management Server 2002 and SharePoint 2003 technologies. It defines the extent to which these technologies offer the best possible compliance with, as a minimum, the priority 1 checkpoints of Web Content Accessibility Guidelines (WCAG) 1.0.

Status

This document is the second release of the technical white paper on the accessibility of Web sites that are based on Microsoft solutions; it references the standards that were available at the date of publication and the features available in current versions of the above-mentioned products and technologies at the date of publication. This document will be updated as required based on product modifications.

The information contained in this document represents the current point of view of Microsoft Corporation concerning the subjects dealt with at the date of publication. Given that Microsoft must adapt to ever-changing market conditions, any and all information contained herein should not be construed in any way as a commitment on the part of Microsoft. Microsoft makes no guarantee as to the accuracy of the contents of this work with respect to any changes or developments occurring after the date of publication.

MICROSOFT PROVIDES NO EXPLICIT OR IMPLICIT WARRANTY OF ANY KIND IN THIS DOCUMENT.

The names of products or companies cited in this document may be trademarks of their respective owners.

Microsoft Corporation • One Microsoft Way • Redmond, WA 98052-6399 • USA

Table of Contents

1. Introduction

The notion of digital accessibility, or "eAccessibility", appeared with the emergence of new information and communication technologies and, specifically, with the advent of the Internet.

The World Wide Web (generally considered to be the most rapidly adopted technology in history) is now part of our everyday life. Its use is rapidly spreading to all sectors of our society. As such, the Internet is the most global source of information.

However, people with disabilities might find it difficult to use this medium, as well as other interactive online services in general.

As Tim Berners-Lee, director of the World Wide Web Consortium (W3C) and inventor of the World Wide Web, says:

> *"The power of the Web resides in its universality. Access by everyone, regardless of disability, is an essential aspect"*

The problem does not reside in physical access to computing equipment (for example, based on certain studies in France, the visually impaired are twice as likely as the national average to have a computer). The real problem lies in the use of the Internet: Web sites browsed by the visually impaired are not always adapted to the assistive devices used, nor do they provide even elementary accessibility features.

Accessibility takes on additional significance because new technologies represent both an incredible potential and, paradoxically, an additional risk of exclusion for the disabled.

Accessibility is currently at the forefront of the political scene. It has become a priority — and a legal obligation.

As an indication of the changing landscape in Europe, we can mention the eEurope 2002 Action Plan concerning the accessibility of public Web sites and their content, which enjoins member states to make their public Web sites accessible to the disabled. Subsequently, at the European Council held in Seville in June 2002, the eEurope 2005 Action Plan confirmed the work of the Web Accessibility Initiative (WAI) launched by the W3C for Web site accessibility (see section 1.3 "W3C recommendations").

As an example of a recent French application, in February 2005 a law was passed (law No. 2005-102) in defense of the "equality of rights and opportunities, participation and citizenship of the disabled," mandating on 11 February 2005 (article 47) that online accessibility is an obligation, just as providing physical access to buildings has been for some time:

All online public communication services of the government and territorial communities and the public organizations under the responsibility of said authorities must be accessible to the disabled.

France has joined other European countries, such as Spain, Italy, Portugal, Great Britain and Sweden that have already passed similar legislation in compliance with European directives and the various action plans and initiatives arising from them. European institutions have committed to comply with current standards in terms of accessibility, and many other organizations within the European Union share this objective.

In the United Kingdom, the Disabilities Discrimination Act 1999 extends this obligation to commercial private Web sites. Since 2004 the Disability Rights Commission has been given a mandate to initiate public investigations, which can lead to litigations. Courts of laws can order the shutdown of a site until compliance is reached. Spain is taking a similar path with article 10 of the Law on Equality of Opportunities and Against Discrimination (2003), which is being reexamined in Parliament, and Italy has already included all educational products and services in its Stanca Law on Equal Access to Information.

Most work regulations also require that disabled employees be able to make full use of their employers' intranets and information systems; it is considered to be discriminatory not to enable access to those key productivity tools.

Additionally, within the framework of the eEurope 2005 Action Plan, the Support-EAM project (Supporting the creation of an e-Accessibility Quality Mark) is intended to standardize the methods used to evaluate accessibility of Web sites, to create a Quality Mark to certify that Internet services are accessible, and to broadly disseminate the use of the Quality Mark throughout Europe. This project makes direct reference to the Council of Europe Resolution 5165/03 concerning eAccessibility, aimed at improving the access of people with disabilities to the knowledge-based society, inviting the Commission and the Member States to "*consider the provision of an 'eAccessibility mark' for goods and services which comply with relevant standards for eAccessibility.*"

Support-EAM launched and conducted a CEN/ISS WAC workshop which produced a CEN Workshop Agreement (CWA n° 15554:2006 Specifications for a complete European Web Accessibility Certification Scheme and a Quality Mark). This agreement has been published by the CEN as a first level European agreement on how standard conformity assessment schemes commonly used in Europe can apply to Web accessibility conformity assessment.

As an illustration, a recent survey by Forrester Research, Inc. (see "Research Studies About Accessible Technology") shows that, by the year 2010, more than half of the population in European countries, such as Germany, Belgium, and France, will be 45 years of age or older. By the year 2050, the population of seniors will be more than double the population of those under 35.

We now enter the "i2010 – European Information Society in 2010" initiative that proposes to provide an integrated approach to the information society and audiovisual policy as implemented within the European Union. The scope of this action includes regulatory initiatives, research, promotional activities, and deployment of cultural diversity initiatives. The purpose is to make the information society accessible to all and to make eInclusion@EU a key contributor to the process of inclusion.

Last on March 27, 2006, Ms Viviane Reding, Commissioner for Information Society and Media, chaired the first meeting of the High Level Expert Group in charge of the i2010 digital libraries initiative which aims at making Europe's diverse cultural and scientific heritage easier and more interesting to use online for work, leisure and/or study. It builds on Europe's rich heritage combining multicultural and multilingual environments with technological advances and new business models.

Within this European context, this white paper is intended to help the reader better understand the issues involved in eAccessibility. We will deal with the current issues in accessibility, as well as the various initiatives and standards implemented with respect to Web-enabled applications. In this regard, we will examine the use of Microsoft products and technologies to design and produce Web sites that comply with accessibility guidelines.

1.1 Defining eAccessibility

Although the problem does not reside in physical access to computing equipment (based on certain studies in France and in the UK, the disabled are twice as likely as the national average to have a computer), people with disabilities encounter real problems when they try to use the Internet and Internet-based services.

Tim Berners-Lee, director of the W3C and inventor of the World Wide Web, describes eAccessibility in this way:

> *“Making the Web and Web services available to all individuals, whatever their hardware or software, whatever their network infrastructure, their language, their culture, their geographic location, or their physical or mental aptitude.”*

In other words, eAccessibility can be defined simply as "enabling any person — whatsoever — to access the Web." A Web site is accessible if it can be used similarly by anyone — with or without a disability. The content, features and services provided by a Web site must be accessible to as broad a public as possible, regardless of age, disability, or limitations in the technology or user environment.

To illustrate the magnitude of the issue, more than 750 million people in the world suffer from disabilities. More than two-thirds of them are in the developing world. In the EU alone, 12% to 15% of the population have disabilities. This data, however, does not include impairments such as color blindness (11 out of 100 men), arthritis (110 million sufferers), dyslexia, or temporary impairments due to accident or surgery[1]. Altogether, 330 million people in the EU, which is almost one in two Europeans, need assistive technology for using information technology.

This data confirms findings from the Forrester research that was commissioned by Microsoft in 2004. The findings from that research stated that 57% of computer users would benefit from assistive technology and 44% of computer users already use one or more accessibility features without necessarily suffering from impairments, simply for ease of use or reading comfort.

Although some people do not use the services available on the Web because they simply do not have the financial, technical, or other means to do so[2], many people do attempt to benefit from such services. It thus seems fair and equitable to remove as many of the obstacles to browsing on the Web as possible.

eAccessibility, in fact, corresponds to a set of technical guidelines that take into account the diversity of these needs and, more specifically, all aspects that might constitute obstacles for the disabled.

In practice, sometimes very little effort is required to solve accessibility issues. Unfortunately such requirements are often not considered early enough during Web site deployment projects.

Developers simply might be unaware of what is required in this area. In addition, they might not have access to guidelines that they can use to ensure that their site is accessibility compliant.

Making a Web site accessible is often perceived as merely providing the disabled with access to digital content. However, Web accessibility provides advantages that go beyond simply facilitating access and ensuring compatibility with assistive technologies for the disabled (see section 1.2.2 "Assistive technologies"). The advantages of eAccessibility extend well beyond the disabled.

When Web-enabled data is designed to be accessible by the disabled, the resulting Web sites are generally more logical and tend to emphasize content rather than merely providing information. A gain of 35% in usability has been noted in a 2004 survey carried out by the Disabilities Rights Commission in the UK. In addition, maintenance and hosting of content-centric Web sites is generally less expensive.

eAccessible Web sites can be accessed by using entry-level computers with slow Internet connections, and so forth. They also facilitate portability to other devices, such as Web-enabled mobile phones and PDAs. This advantage is a significant factor, since already in 2004 there were 720 million users of mobile devices.

1.1.1 Accessibility or usability?

Although these two notions are closely linked, there are a few essential differences.

Even if some Web sites are technically accessible, they are not always very easy to use. An accessible Web site is thus not always a usable Web site.

Whereas the purpose of accessibility is to "open up" a Web site to as broad a public as possible, the purpose of usability is to improve user efficiency, effectiveness, and satisfaction when accessing the site.

Accessibility covers technical aspects and considerations, whereas usability concerns the experience of the user when accessing a given Web site. Usability can be compared to the notion of quality in that it is most often noticed when it is unfortunately absent!

[1] Source : Commission des communautés européennes; European Disability Forum, Disability and social exclusion in the European Union, 2002, VISCHECK, World Dyslexia Network Foundation, PARE Manifesto

[2] This related topic is discussed in the digital inclusion (or eInclusion) section below

Usability can thus be defined as "the ease with which visitors can use a Web site." Web usability is not just a matter of making sure that everything works on the site; developers also need to measure how quickly and easily visitors can use the site.

Jakob Nielsen's law (see useit.com: Jakob Nielsen's Web site) concerning the Internet user's experience probably provides the best definition of the lack of Web site usability:

> *"It is when users spend most of their time on other Web sites"*

Web site usability involves such issues as user-friendliness and intuitive browsing, as well as download times, page layout, the graphical charter, animations, browsing experience, data architecture, search engine design, and so forth.

Usability can be evaluated only by the actual experience and satisfaction of the end-user:

- Can the user easily find information she or he is seeking?
- Are the offered services easy to access, and is the provided information easy to understand?
- Is the Web site content presented in a consistent fashion?
- Is the structure of the Web site logical and easy to understand? Can the user browse through the site easily and effectively?
- Does the site provide a clear explanation of how it is organized and how the user is intended to browse through it?
- Is the user able to work interactively with the site and provide feedback? Does the user obtain and provide rapid and appropriate responses?
- Is a search function available on the site?

As illustrated by this partial list of potential issues, the notion of Web usability is extremely far-reaching, and we could not hope to cover all aspects of it in detail.

1.1.2 Typical misconceptions and misunderstandings

Certain preconceived ideas hinder the advancement of widespread accessibility. Some people are convinced that:

- *The disabled do not have access to computers, so why worry about facilitating Internet access for them?* This is untrue; a variety of hardware and software assistive products are available (see section 1.2.2 "Assistive technologies"). Moreover, certain surveys in the USA indicate that the hearing impaired, for example, are more than twice as likely as the national average to have a computer.
- The Internet is based on the display of graphical images and various other media that cannot be made accessible to everyone —the visually impaired, for example. All that is required is a text alternative to pictures, which makes the information readable by a Braille device or text-to-speech software.
- *Accessibility kills creativity: Accessible Web sites contain only text and, for this reason, they are boring and monotonous.* Fortunately, this is untrue. As we have just seen, Web accessibility is not a question of restrictions leading to a Spartan interface, but rather a question of substantial overall improvement. One of the concerns is that non-text content should be accessible to people with disabilities. Major companies provide guidance on how to make animated content accessible. Creating accessible Web 2.0 sites is equally possible.
- *Basic accessibility features are difficult and very costly to implement.* This is untrue; elementary directives exist that, with just a slight effort, can greatly improve the accessibility and usability of your Web site. In the following chapters of this white paper, we attempt to illustrate some of these ideas as implemented by using Microsoft products and technologies.

1.1.3 Accessibility starts at the design phase

Web accessibility starts right from the design phase. In other words, you must keep accessibility in mind when designing a Web site. "Accessible design" does not mean "minimalist design," but rather "well-thought-out design!"

Accessibility is not a constraint, but rather a guarantee that is part of the quality assurance process.

Each step of the project is involved, from the first brainstorming session, through site development, to maintenance and site monitoring.

Taking accessibility into account during the design phase has an impact on all levels: ease of use, graphical design, and technical design. The experience of the project team and the solutions, tools, and technologies envisioned should enable the implementation of Web site interfaces that provide high levels of performance and are suited to the purpose, while considering the recommendations of the WAI that deal with accessibility (see section 1.3 "W3C recommendations").

Thus, considering accessibility-related recommendations should come into play right from the design phase, as follows:

- Design a graphical charter that supports access for a broad range of disabilities (see section 1.2 "Disabilities").
- Check whether certain standard user-friendly interfaces are appropriate, and adapt as necessary.
- Set up a storyboard taking into account specific criteria, such as data table management.
- Define and draft an editorial charter by implementing systematic rules (length of content, use of acronyms, content hierarchy, and so forth).
- Select content management tools to ensure that future content will also be accessible.
- If experienced screen reader users can be involved, define a testing platform that implements browser testing tools, such as Lynx or JAWS, in addition to the usual Web browsers. Accessibility validation testing must always be included in the quality assurance process used during the Web development project.
- And so forth.

Each deliverable produced must be assessed to ensure that accessibility criteria are addressed.

1.1.4 Accessibility is an ongoing process

The production phase must ensure strict compliance with standards and rules drawn up during the design phase. Rigor, consistency, and reinforced testing procedures are the key to success in this approach.[3]

The use of a flexible and dynamic methodology such as Microsoft Solutions Framework (MSF), which is the result of years of experience acquired by Microsoft development teams and Microsoft consultants throughout the world, makes it possible to maintain full control over the management processes and techniques in projects of this type.

Before describing the current standards dealing with accessibility, we first discuss the various disabilities and the relevant adaptive strategies.

[3] The development of Internet services that are designed for specific categories of disabled people, such as online translation into sign language or digital libraries, are not directly part of digital accessibility, but they do share a complementary and similar approach

1.2 Disabilities

As we stated at the beginning of this white paper, the Internet can present some significant obstacles to the disabled. To better understand Web site accessibility, we first need to identify the various types of disabilities.

1.2.1 Types of disabilities

Disabilities can be broken down into the following types:

- **Visual disabilities** - This type of disability includes:
- *Blindness* - By blindness we mean a substantial and incurable loss of vision in both eyes.
- *Weak vision* - (This is also sometimes referred to as "poor vision" in some parts of Europe.) Weak vision can include weakened visual acuity (imprecise vision), tunnel vision (vision limited to the very center of the visual field), loss of central vision (vision limited to the periphery of the visual field), or blurred vision.
- *Color blindness* - This condition is characterized by a lack of sensitivity to certain colors. Among the most common types of color blindness are the inability to distinguish between red and green or between yellow and blue. Color blindness can also include the inability to perceive colors.
- **Hearing disabilities** - This type of disability includes:
- *Deafness* - By deafness we mean a substantial and incurable loss of hearing in both ears. Sometimes the mother tongue of a deaf person is sign language, and the person cannot fluently read or write any other language.
- *Loss of hearing acuity* – This condition is characterized by slight to moderate hearing loss. People with loss of hearing acuity are sometimes considered to be "hard of hearing."
- **Physical disabilities** - Physical disabilities include conditions related to impaired motor functions. These conditions might involve weakness or loss of muscular control (such as uncontrolled movements, loss of coordination, or paralysis), loss of the sense of touch, joint problems, or missing members. Some physical handicaps might also include pain that restricts mobility. These disabilities might involve the hands, the arms, or other parts of the body. With 100 million arthritis sufferers in Europe and an increasingly aging population, the number of people with this type of disabilities is likely to grow.
- **Speech-related disabilities** - Speech disabilities or impediments might include difficulties in producing speech that is recognizable by voice recognition software, whether due to weakness of the voice or to lack of clarity in the sound produced.
- **Cognitive or neurological disabilities** - These types of disabilities include:
- *Dyslexia* - People suffering from dyslexia or "math dyslexia" (sometimes referred to as learning disabilities in the United States) might have difficulty understanding written language or pictures (if reading is performed visually), they might have difficulty understanding spoken language, or they might have difficulty understanding figures, either spoken or read.
- *Attention deficit disorders* - People suffering from attention deficit disorders may experience difficulty concentrating on the information presented. Looped animations that cannot be paused will be an issue for them.
- *Intellectual disorders* - People suffering from intellectual disorders (sometimes referred to as learning disabilities in Europe) might need more time to learn or might experience difficulties understanding complex concepts.

- *Memory disorders* - People suffering from memory disorders might have problems with either short-term or long-term memory, or they might lose the use of speech.
- *Mental health disorders* - People suffering from mental or emotional health disorders might have difficulty concentrating on the information provided by a Web site. They might be troubled by blurred vision or trembling hands, which can be side effects of medication they are taking.
- *Seizure disorders* - Some people are subject to seizures that can be triggered by a flickering screen or by a sound of a certain frequency.
- **Multiple disabilities**- People suffering from multiple disabilities might have reduced capabilities for accessing and interacting with information online.
- **Problems linked to aging** - Aging can cause discrete or progressive change in functional abilities or a combination of changes linked to vision, hearing, mobility, and memory. The likelihood of experiencing one or more impairments is three times greater after age 50 than before age 40.

1.2.2 Assistive technologies

Assistive technologies (AT) are products used by the disabled to accomplish everyday tasks that would be difficult or impossible to accomplish without such assistance. AT devices are designed to improve accessibility for individuals who have physical or cognitive difficulties, impairments, and disabilities.

When linked to a computer, assistive technologies include hardware and software solutions providing so-called "disability compensating technologies."

Some disability compensating technologies depend on other hardware or software, such as a graphical browser (as opposed to a text-based browser), CD-ROM or DVD-ROM, and so forth.

There are many types of assistive technology products and hundreds of products available. Here are a few examples of AT devices:

- Specially adapted keyboard[4]
- Virtual keyboard[5]
- Braille pad
- Magnification software (screen zoom)
- Audible alarms
- Screen or paragraph reader
- Voice recognition
- Optical character recognition
- Text-based browsers
- Voice-based browsers

For additional information and user scenario, please refer to the W3C article How People with Disabilities use the Web, which additionally provides links to various specialized browsers, screen readers, voice browsers, and other access facilitating solutions.

Microsoft works closely with assistive technology manufacturers in the Microsoft Assistive Technology Vendor Program (MATvp). These partners have a proven track record of designing, building, and supporting assistive technology products. According to the right type of assistive

[4] This type of keyboard has smaller or larger keys than a standard keyboard, can have different key configurations, and can sometimes be used with a single hand.

[5] This type of keyboard displays directly on the user's screen, and can support the use of a laser pointer, for example.

technology that might be useful, you can search the catalog of assistive technology products for products compatible with the Windows operating system.

An AT device sits between a user and an application and allows the user to interact more successfully with that application. The devices translate the application data into a format that the user can access and interact with and, in turn, render the user's input into a format that the application can interpret.

To function effectively, AT devices must be compatible with the computer operating system and programs on the particular computer being used. Fortunately, developers do not need to understand each AT device. AT devices use a standard object model, such as the Document Object Model (DOM), or a set of interfaces, such as Microsoft Active Accessibility (MSAA), to communicate with a client application running on Windows or with an application running on the Web. A wide variety of AT devices are built on this common base (see above).

MSAA is a set of COM interfaces and application program interfaces (APIs) that provides a reliable way to expose and collect information about Microsoft Windows-based UI elements and, in the present context, Web content. AT devices can then use this information to communicate the UI in alternative formats, such as voice or Braille, and voice command and control applications can remotely manipulate the interface. MSAA comes with its in-depth documentation and a set of tools. Indeed, you might not have AT devices available to you when you design, develop, and test your applications (see chapter 2 Recommended approach for accessible Web content).

However, the MSAA tools can serve as proxies for actual assistive technology devices. These tools are free and easy to use, and can help you uncover the vast majority of accessibility issues.

1.2.3 Adaptive strategies

Adaptive strategies are techniques used to facilitate browsing through Web sites.

A person who is visually impaired or has restricted mobility can use certain keys on the keyboard as tab keys to jump from link to link and from tab to tab, such as for an online form. These techniques can be used with or without the help of the disability compensating technologies mentioned above.

These issues are developed in the working document from the W3C entitled How People with Disabilities Use the Web dated 5 May 2005. This document introduces the use of the Web by the disabled, as well as describes some of the requirements disabled people have when they use Web sites and Web-based applications. Lastly, this document provides information about the work of the WAI of the W3C consortium, which we discuss in the following section.

We also want to mention here the availability on the Microsoft Accessibility Web site of several guides organized around the various types of difficulties and impairments. These guides are available at http://www.microsoft.com/enable/guides/default.aspx . Each guide provides a list of assistive technology products and links to Step by Step Tutorials for accessibility features that are helpful for specific types of difficulties and impairments.

In the context of this white paper, the Step by Step Tutorials for Microsoft Internet Explorer 7 and Microsoft Internet Explorer 6 Accessibility Options http://www.microsoft.com/enable/training/ie7/default.aspx might also be helpful.

1.3 W3C recommendations

The Web Accessibility Initiative (WAI) is a special task force that was set up within W3C in 1996 to deal with accessibility issues. The purpose of WAI was to provide the basis for the development of eAccessibility standards through the drafting of a series of recommendations.

1.3.1 WCAG 1.0

1.3.1.1 Overview

One of the recommendations published by the W3C/WAI on 5 May 1999 was release 1.0 of the Web Content Accessibility Guidelines (WCAG).

These guidelines explain how to make Web content accessible to the disabled. They were specifically written for all Web content creators (that is, those involved in creating Web pages or designing Web sites) and developers of tools for creating content.

The main purpose of the guidelines is to promote accessibility for the disabled. However, when a Web site complies with these guidelines, the content is optimized for all users, independent of the program used to access the site, whether classical browser, text browser, voice browser, smart phone, and so forth, and regardless of the constraints imposed by the environment, such as surrounding noise, excessive or inadequate lighting, or hands-free use.

These guidelines can enable users to find information on the Web more rapidly. The guidelines are not intended to discourage content creators from using multimedia content, but rather to explain how to make that content more accessible to a broader audience.

The WCAG 1.0 guidelines establish a stable framework for developing accessible Web sites and help people with disabilities to access such Web sites.

Today, these guidelines are generally considered to be the standard reference for creating accessible Web content. They have been recognized by most countries since 1999. They have been recognized on the European level since 2002.

Generally speaking, WAI guidelines are considered to be "law," whereas the WCAG 1.0 guidelines are the actual practical implementation thereof, that is, the "decrees."

The WCAG 1.0 guidelines include 14 guidelines that were designed to support forward compatibility to keep up with evolving Web technology, while reducing processing overhead for legacy (older generation) browsers.

These guidelines are intended to ensure that (X)HTML code is proper and complete; they also state several common sense rules. The (X)HTML code must be compliant from a syntax and grammatical point of view: elements must be properly closed and nested, attributes must be enclosed in quotes, and so forth. Anyone using any type of software should be able to access and use the information provided. For example, images should present alternate text, and data tables should provide a summary, a title, and meaningful headers. A given Web page should be useful even if the browser used does not understand JavaScript, Flash or multimedia elements. Thus, the portions of a Web site that use these technologies should also have a pure HTML equivalent on the same page. However, the code used should be as lightweight is possible and should contain only minimal amounts of unnecessary HTML code to avoid penalizing those who use a text-based browser or assistive technologies (see section 1.2.2 "Assistive technologies"). These few examples serve to illustrate the principles of accessibility addressed in the guidelines.

Each of these guidelines explains how accessibility principles can be applied to the characteristics of a Web site. The guidelines make it possible for the disabled to view Web pages by using existing technical aids (Braille pads, text-to-speech, laser pointing devices, magnification software, and so forth). For example, one of the guidelines indicates that any image should present alternate text so that a visually impaired person can access the information provided by the image. Only the alternate text is readable on a Braille pad or is spoken by using text-to-speech software. It is particularly important that this guideline is strictly followed for Web sites in which only images are used for browsing. Another guideline concerns captions for sound files that provide information to people who are hearing impaired.

WCAG 1.0 guidelines take all disabilities into consideration and are based on standard Internet technology. However, following these guidelines also ensures that Web sites are usable for everyone, regardless of the computer configuration. For example, some browsers might be configured to display pages without images to speed up surfing. In such cases, alternate text appears in the place of the images and enables the user to obtain the appropriate information. This approach also reduces maintenance costs and improves portability of the Web site to a variety of devices. This advantage is significant, given that already in 2004 there were 720 million users of mobile devices.

Each guideline in WCAG 1.0 includes a list of checkpoints that apply to it. Each checkpoint describes how the guideline applies to a typical Web development situation and links to techniques that describe, with detailed examples and explanations, how to implement the checkpoint. The 65 checkpoints with their associated techniques can be found at Techniques for Web Content Accessibility Guidelines 1.0. The techniques might (depending upon the topic and the technical area involved) refer to one of the following documents:

- Core Techniques for Web Content Accessibility Guidelines 1.0, which discusses general accessibility techniques that apply to all technologies.
- HTML Techniques for Web Content Accessibility Guidelines 1.0, which provides examples and strategies to help define accessible content by using Hypertext Markup Language (HTML).
- CSS Techniques for Web Content Accessibility Guidelines 1.0, which provides examples and strategies for writing Cascading Styles Sheets (CSS) as one component in the design of accessible content.

To accommodate the rapid pace of technological change, these documents are regularly updated.

The WCAG 1.0 guidelines are part of a series of guidelines on Web accessibility published by the W3C/WAI at WAI Guidelines and Techniques. It should be noted that the W3C/WAI also provides a home page in other languages.

This series includes:

- Authoring Tool Accessibility Guidelines (ATAG) 1.0 - These guidelines describe how the various content creation tools can be used to create accessible Web pages and how to make a Web site accessible.
- User Agent Accessibility Guidelines (UAAG) 1.0 - These guidelines explain how to improve accessibility for browsers, multimedia drives, and assistive technologies (see section 1.2.2 "Assistive technologies," which is closely related to these guidelines).
- XML Accessibility Guidelines Working Draft - These guidelines explain how to ensure that XML-based applications support and enable accessibility.

To help developers find their way amidst all these guidelines, the W3C/WAI also provides an introductory text on the basics of making a Web site accessible: WAI Resources on Introducing Web Accessibility.

1.3.1.2 Priorities and levels of compliance

In section 5 on Conformance, the WCAG 1.0 document identifies three levels of priorities corresponding to three levels of compliance with the guidelines:

- **Priority 1** - Web content developers must satisfy checkpoints assigned Priority 1. If a Web site does not comply with these checkpoints, some users will find it impossible to access the content on the site. This checkpoint is a basic requirement to enable some groups to use Web documents.
- **Priority 2** - Web content developers should satisfy checkpoints assigned Priority 2 to reduce obstacles in accessing Web sites. If a Web site does not comply with these checkpoints, some users will find it difficult to access the content the site. This checkpoint removes significant obstacles to accessing Web content.
- **Priority 3** - Web content developers may satisfy checkpoints assigned Priority 3 to improve ease of use. If a Web site does not comply with these checkpoints, some users will find it somewhat difficult to access the content on the site. This checkpoint improves access to Web content.

A list of the relevant checkpoints for each guideline in priority order is provided at Checklist of Checkpoints for Web Content Accessibility Guidelines 1.0.

The guidelines also define three levels of compliance to simplify comparisons:

- W3C WAI-A WCAG 1.0 **Level of compliance "A"** - A Web site is Level A compliant if it satisfies all Priority 1 checkpoints.
- W3C WAI-AA WCAG 1.0 **Level of compliance "AA"** - A Web site is Level AA compliant if it satisfies all Priority 1 and 2 checkpoints.
- W3C WAI-AAA WCAG 1.0 **Level of compliance "AAA"** - A Web site is Level AAA compliant if it satisfies all Priority 1, 2, and 3 checkpoints.

1.3.2 WCAG 2.0

On 8 January 2003, the W3C/WAI published an editor's draft of the WCAG, which was referred to as release 2.0 of the WCAG.

The latest official release of the proposal[6] as of the writing of this white paper, the Web Content Accessibility Guidelines 2.0 - W3C Working Draft 27 April 2006, is largely based on WCAG 1.0 and shares the same objectives: explaining how to make Web content accessible to people with disabilities and how to define target thresholds of accessibility.

Release 2.0, however, has been enriched by all the feedback and experience arising out of WCAG 1.0 and deals specifically with checkpoints. It attempts to apply checkpoints to a broader range of technologies and uses a language that is more accessible to a heterogeneous readership.

The W3C/WAI takes great care to ensure that the organizations and people currently using WCAG 1.0 will later be able to move easily to release 2.0 of these guidelines. **WCAG 1.0 is the last approved stable release and, as such, is considered to be the reference, until release 2.0 has been approved.**

1.3.2.1 Main differences between release 1.0 and the proposed release 2.0

Since the publication in May 1999 of WCAG 1.0, the W3C/WAI has obtained a great deal of feedback and information with respect to priorities, notably with respect to checkpoints and the usability of the documents provided, as well as requests for clarification about the meaning of certain checkpoints and the appropriate way to comply with certain checkpoint requirements.

The proposed release 2.0, when it becomes an official W3C recommendation (probably during the year 2007), should, in the light of available documents, accomplish the following:

- Be organized more effectively
- Adapt the priority of some checkpoints
- Modify, remove, and add requirements to reflect changes in Web technology since the publication of WCAG 1.0
- Correct errata in WCAG 1.0
- Reflect the experience acquired in the implementation of WCAG 1.0

The organization of WCAG 1.0 is by guideline, each of which links to checkpoints of priority 1, 2, or 3. The checkpoints are the criteria for determining compliance with WCAG 1.0. The most recent version of the proposed 2.0 release is organized by four main principles of Web accessibility design. Each of these principles has its own guidelines, and each guideline has criteria for success corresponding to levels 1, 2, and 3. The criteria for success are used to determine compliance with WCAG 2.0.

[6] Please note that a more recent release of the proposal, the Web Content Accessibility Guidelines 2.0 – Editor's Draft dated August-September 2006, is also available. This release is probably closer to w hat the final draft will look like.

1.3.2.2 Improvements in proposed release 2.0

The proposed release 2.0 of the guidelines for accessibility of Web content provides several improvements over the preceding version. Although the main objective of release 2.0 is the same as WCAG 1.0, that is, to promote Web content accessibility, release 2.0 includes additional objectives, including:

- Ensure that requirements can be applied to all technologies;
- Ensure that requirements for compliance are clear;
- Ensure product ease of use;
- Draft guideline text using a language that is accessible to a more heterogeneous readership;
- Clearly identify the beneficiaries of accessible content;
- Ensure that the new release provides for backward compatibility with WCAG 1.0.

1.4 EU Unified Web Evaluation Methodology (UWEM 1.0)

In Europe, the publication, last on July 5, 2006, of the version 1.0 of the EU Unified Web Evaluation Methodology (UWEM) is the result of a work undertaken jointly by 3 European projects combined in a cluster called the Web Accessibility Benchmarking (WAB) Cluster, a cluster of European projects to develop a harmonized European methodology for evaluation and benchmarking of websites:

- European Internet Accessibility Observatory (EIAO) project, which aims at establishing, as its name indicates, the technical basis for a European Internet Accessibility Observatory,
- Supporting the creation of an eAccessibility Mark (Support-EAM) project, which aims at harmonizing the Web accessibility in Europe, more particularly on the basis of a unified and harmonized methodology, and at creating a European Web Accessibility Quality Mark for Web sites,
- Benchmarking Tools for the Web (BenToWeb) project, which aims at constituting benchmarking modules for testing the Web accessibility.

23 European organizations contribute to these projects.

The publication of UWEM 1.0 constitutes along with the publication of a Web accessibility certification scheme the major fact of the year 2006. These two publications make it possible today to organize the effective harmonization of the accessibility of the Web at the European level.

The UWEM 1.0 methodology is conformant with WCAG 1.0 directives (see section 1.3.1 WCAG 1.0) and will be synchronized with the foreseen migration from WCAG 1.0 to the proposed 2.0 release (see section 1.3.2 WCAG 2.0).

The UWEM 1.0 methodology offers an interpretation of WCAG 1.0. This interpretation is agreed among stakeholders within the above projects. Practical reasons have conditioned this choice. Indeed, WCAG 1.0 are largely accepted and constitute a stable factor of accessibility since May 1999. In 2002 already, the European Union recommended that they are adopted by the public sector of the Member States. In several countries, they are integrated like reference to the national legislation making compulsory the accessibility of public sector Web sites.

Even the evaluation of UWEM is also designed to contribute to the work of the W3C/WAI on the evolution of the WCAG directives. It should be noted that, if the W3C/WAI has evaluated and provided a contribution to former outlines of this methodology, that does not imply in anyway the approval by the W3C/WAI of whole or part of this methodology.

The objective of UWEM 1.0 primarily consists in providing a single base allowing at the same time the evaluation of the Web accessibility by a human expert and in an automatic way by machine interfaces.

UWEM 1.0 primarily covers methods to evaluate documents based on the following technologies: HTML 4.01, XHTML 1.0 and 1.1, CSS 2.x, and other embedded objects in (X)HTML resources.

UWEM 1.0 is declined in two levels of conformance:

- Priority 1 of WCAG 1.0 - a Web site has the level of priority 1 (P1) if the 54 tests of this level passed successfully;
- Priority 2 of WCAG 1.0 - a Web site has the level of priority 2 (P2) if the 92 tests of this level passed successfully

The French AccessiWeb Quality Label is the first label in Europe to organize its migration towards the European methodology UWEM and the future European Quality Mark. Indeed, the French organization Braillenet.org has released last July 2006 the French translation of UWEM1.0 and a mapping to its existing AccessiWeb guidelines (already conformant with WCAG 1.0).

1.5 Microsoft's commitments

Microsoft takes into account the requirements of the ill and the disabled in the field of computing to enable them to benefit from new technologies, both professionally and personally.

Microsoft recognized early on that information technologies are important and empowering tools for people who suffer from disabilities or whose faculties are impaired in some way. For two decades, Microsoft has explored and improved the accessibility features integrated in its products. Microsoft is committed to innovation in this area (see Microsoft's Corporate Mission and Accessibility Strategy) and, with a view to furthering advances in the field, has created the Accessible Technology Group (ATG), staffed by 50 professionals who manage and coordinate all related Microsoft projects, and work with nearly 300 assistive technology vendors worldwide through the Microsoft Assistive Technology Vendor program (MATvp).

The Microsoft Accessibility Web site and the free *Accessibility Update* e-newsletter (both at www.microsoft.com/enable) provide in-depth information about the accessibility of Microsoft products.

Microsoft's commitment to accessibility empowers public authorities by providing opportunities and choice in accessible technologies. This improves opportunities for employment of people suffering from a disability or incapacity and offers better access to public services. Microsoft supports public initiatives, guidelines, and regulations in the drive to encourage eInclusion by providing information about Microsoft accessibility products and the ways in which these products respond to a variety of accessibility standards.

In the United States, section 508 of the Rehabilitation Act requires that federal agencies make information and electronics technology accessible to the disabled. Section 508 creates a strong motivation for software publishers to integrate more and better accessibility features into their products. This reinforces and extends the work that Microsoft has already done to make technology universally accessible.

For more information about Microsoft's public commitment, see Microsoft Actively Supports Section 508.

In Europe, Microsoft supports the eInclusion initiative of the European Community mentioned earlier and participates in the trans-Atlantic i2010 support initiative.

Within this framework, the purpose of this white paper is to:

- Indicate the level of native compliance of the main Microsoft solutions available today for designing and publishing Internet, intranet, and extranet Web sites, referring to access criteria in the accessibility guidelines.
- This is based, among other things, on the Voluntary Product Accessibility Template (VPAT) documents for the corresponding Microsoft products. The VPAT documents, which are available at http://www.microsoft.com/resources/government/section508.aspx, address the Section 508 standards. Each VPAT document describes how a software product does or does not comply with the standards specified under Section 508.
- Identify the necessary recommendations for the design and production of accessible Web content.
- Propose an approach to assessment and validation using specific tools, before beginning the certification process for the resulting Web site.

This white paper is concerned with the various Microsoft solutions available today for the design, management, and publication of Internet, intranet, and extranet Web sites.

These solutions not only include:

- The new Windows SharePoint Services (WSS) 3.0 and Microsoft Office SharePoint Server (MOSS) 2007;

But also:

- Microsoft Content Management Server (MCMS) 2002;
- Windows SharePoint Services (WSS) 2.0 and Microsoft SharePoint Portal Server (SPS) 2003;

The wording "MOSS 2007 technologies" used in the rest of this document covers both Windows SharePoint Services 3.0 and Microsoft Office SharePoint Server 2007.

Likewise, the wording "SharePoint 2003 technologies" used in the rest of this document covers both Windows SharePoint Services 2.0 and Microsoft SharePoint Portal Server 2003.

Finally, MOSS 2007 technologies supersede both MCMS 2002 and SharePoint 2003 technologies.

As we will see, these products and technologies supports the Visual Studio 2005 development environment, and thus supports .NET Framework 2.0/3.0[7] and, in part, ASP.NET 2.0 technology.

ASP.NET 2.0 technologies offer significant improvements for eAccessibility in six areas: accessible images, accessible forms, accessible browsing, accessible data, master pages, and XHTML. All these elements are developed and illustrated in the MSDN article Building ASP.NET 2.0 Web Sites Using Web Standards.

ASP.NET 2.0 technologies are designed to facilitate compliance with all priority 1 and 2 checkpoints in WCAG 1.0, as well as with all section 508 standards. All these guidelines are taken extremely seriously, and it is the responsibility of all developers working with ASP.NET 2.0 Framework to review and test each ASP.NET control with respect to eAccessibility.

This document is intended to provide technical guidelines for eAccessibility and is to be broadly available to all stakeholders —- whether during design, acceptance, or periodic checks on Web sites.

The framework and the general context of this white paper have now been outlined. We will now outline key concepts to adopt throughout the lifecycle of any web site development project in order to maximize the ability to comply with the WCAG 1.0 checkpoints and will then discuss the

[7] The .NET Framework 3.0 adds new technologies to the .NET Framework 2.0, which makes the .NET Framework 3.0 a superset of the .NET Framework 2.0.

Microsoft solutions mentioned previously, starting with Microsoft Content Management Server 2002.

2. Recommended approach for accessible Web content

The followings sections:

- outline key concepts that must be consider throughout the lifecycle of any web site development project in order to maximize the ability to comply with the WCAG 1.0 checkpoints,
- and suggest various manners to asses and validate the accessibility functions implemented on the Web site.

Most of the points are based on the article Quick Tips to Make Accessible Web Sites, published by the W3C/WAI. This article summarizes key concepts to address when you design and develop Web sites for accessibility. These key concepts are taken from Web Content Accessibility Guidelines 1.0 (WCAG 1.0*)*.

2.1 Design checklist for accessible Web content

It is important that Web application architects and designers consider the needs of all users during the initial planning and design phases of development. You should do this early because it might affect your design in ways that would be difficult to change after your code is written.

This section identifies a series of questions for defining an editorial charter that implements systematic design rules. It is based on Quick Tips to Make Accessible Web Sites, published by W3C/WAI. The article summarizes key concepts to address when you design Web sites for accessibility. These key concepts are taken from the Web Content Accessibility Guidelines 1.0.

Key concepts to consider include:

- **Colors** - Is information that is provided or rendered by using color still readable if colors are disabled? (WCAG 1.0 checkpoint 2.1) Is there enough contrast between the colors to be distinguishable by users with impaired color vision? (WCAG 1.0 checkpoint 2.2)
- **Multimedia** - Can the information that is conveyed by multimedia be provided another way? (WCAG 1.0 checkpoint 1.1) Is the Multimedia content synchronized with the alternate support? (WCAG 1.0 checkpoint 1.4).
- **Tables** - Is the content in formatted tables in correct sequence? (WCAG 1.0 checkpoint 5.3)
- **Links (hypertext)** - Do all identical link titles lead to the same target? (WCAG 1.0 checkpoint 13.1)
- **Scripts (ActiveX controls, and so forth)** - If a script requires alternate text to be accessible, is the information provided by the alternate text equivalent to the information provided by the script? (WCAG 1.0 checkpoints 6.2, 6.3, 8.1) It must also be possible to browse through Web site using just a keyboard (WCAG 1.0 checkpoints 6.4, 9.2, 9.3)
- **Mandatory elements** - Is the content of the TITLE tag different from one page to the next? (WCAG 1.0 checkpoint 13.2)
- **Information structuring (Organization)** – Is information structured consistently for the general context of the site? (WCAG 1.0 checkpoint 14.1) Is the Web page presented in a consistent fashion? (WCAG 1.0 checkpoints 3.5, 13.8)
- **Presentation of information** - Is the order in which information appears the same with or without the style sheet (WCAG 1.0 checkpoint 6.1)
- **Browsing aids** – Is the main navigation menu on the Web site located in the same place on all pages? (WCAG 1.0 checkpoints 13.4, 14.3) Is the user informed when a

new window is opened? The use of JavaScript pop-ups is not recommended (WCAG 1.0 checkpoint 10.1)

- **Ability to test content accessibility** – Each deliverable must be checked and assessed to ensure that the project accessibility objectives have been reached. To aid in this endeavor, define a set of testing procedures that, depending on project scope, include a project test plan, testing tools to be selected, testing scripts to be written, and so forth.
- The tools selection section on the W3C/WAI Web site, Selecting Web Accessibility Evaluation Tools, describes what tools can and cannot do to assess a Web site, as well as describes the various issues involved in selecting software tools:
- Some tools make it possible to test all or part of the accessibility features on a Web site based on the level of compliance with WCAG 1.0 priority 1, 2, and 3 checkpoints, section 1194.22 of section 508 Voluntary Product Accessibility Template (VPAT), color contrasts used in the Web site, and so forth.
- Other tools can be used to repair existing pages and to code for accessibility; they do, however, still require human intervention.
- Lastly, filtering and transformation tools can help simulate certain disabilities (see section 1.2 "Disabilities" in this document) or refer to the various browsing methods available.
- A list of more than 30 tools that can be used to assess, reconfigure, repair, filter or transform Web sites is available on the W3C/WAI Web site at Web Accessibility Evaluation Tools: Overview.

2.2 Coding checklist for accessible Web content

The development phase must ensure strict compliance with standards and rules drawn up during the design phase (see preceding section). Rigor, consistency, and reinforced testing procedures are the key to success in this approach.

Key concepts to consider during the production phase include the following:

- **Graphics** – The ALT attribute should be used systematically in (X)HTML to describe the function of each graphic (IMG, SHAPE, MAP) with alternate text (WCAG 1.0 checkpoint 1.1).
- **Images and animations** - The following rules should be observed:
- The alternate text should be set to null or no value (ALT="") for images that are solely decorative (WCAG 1.0 checkpoint 1.1).
- The alternate text should be appropriate for the context in which the image is viewed (WCAG 1.0 checkpoint 1.1).
- The alternate text should be no more than 60 characters long (WCAG 1.0 checkpoint 1.1).
- For images that contain text, the alternate text should contain the same text (WCAG 1.0 checkpoint 1.1).
- For images used as a hyperlink, the alternate text should indicate the function of the link (WCAG 1.0 checkpoint 13.1).
- Use the LONGDESC attribute to provide a detailed description (WCAG 1.0 checkpoint 1.1).
- This detailed description should be relevant (WCAG 1.0 checkpoint 1.1).
- **Clickable images** - Use the MAP element and describe the active areas in a relevant fashion (WCAG 1.0 checkpoints 1.1, 1.2, 9.1).

- **Figures and diagrams** - Describe these elements within the page itself, or use the LONGDESC attribute to provide a detailed description (WCAG 1.0 checkpoint 1.1).
- **Frames** - Use NOFRAMES and meaningful titles (WCAG 1.0 checkpoints 6.5, 12.1.
- **Multimedia** - Provide captions and subtitles for audio and descriptions for video. (WCAG 1.0 checkpoint 1.1)
- **Tables** - Facilitate line-by-line reading and summarize. The following rules should be observed:
- Use a relevant SUMMARY attribute (information about the function of the table) (WCAG 1.0 checkpoint 5.5).
- Assign a CAPTION attribute to the data table (WCAG 1.0 checkpoint 5.5).
- Use headers in a data table. Use a TH tag for each column (WCAG 1.0 checkpoints 5.1, 5.4).
- Assign a HEADER attribute to each cell in a data table (WCAG 1.0 checkpoint 5.2).
- **Links (hypertext)** - Use relevant descriptive text when the context is not apparent. The following rules should be observed:
- Link titles should be no more than 80 characters long and should be explicit (do not use "Click here") (WCAG 1.0 checkpoint 13.1).
- Use the TITLE attribute to provide additional information, and keep the text to 80 characters or less (WCAG 1.0 checkpoint 13.1).
- **Scripts (ActiveX controls, etc.)** - Provide alternate text if the active context is inaccessible or is not processed. The following rules should be observed:
- If a script requires alternate text to make it accessible, is the information provided by the alternate text equivalent to the information provided by the script? If so, use validation controls on the client side and on the server side (WCAG 1.0 checkpoints 6.2, 6.3, 8.1).
- In general, the use of scripts should be avoided on the client side. If the use of scripts is unavoidable, an alternative solution should be made available. The information should remain available even if scripts are disabled.
- **Mandatory elements** - The following elements might need be present, depending on context:
- The DOCTYPE tag should be at the beginning of the page source code (WCAG 1.0 checkpoint 3.2). Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

- The LANG attribute in the HTML tag must indicate the language used in the text (WCAG 1.0 checkpoint 4.3). Example:

```
<html lang="EN">
```

- The TITLE tag in the header must be present and explicit on all pages (WCAG 1.0 checkpoint 13.2). Example:

```
<title>Explicit title ofcontent</title>
```

- Indicate language changes in text by using the LANG attribute (WCAG 1.0 checkpoint 4.1). Example:

```
<SPAN lang="en">something in english</SPAN>
```

- **Information structuring (Organization)** - Use section headers and a consistent structure (WCAG 1.0 checkpoint 3.5). Page content should be separated from content introduction: No HTML formatting tags should be used (for example, BGCOLOR, FONT, B, U, ALIGN, VALIGN). All layout elements should be defined by using CSS style sheets as much as possible (WCAG 1.0 checkpoint 3.3).

The page should be readable without the CSS style sheet. (WCAG 1.0 checkpoint 6.1).

- **Presentation of information -** The following rules should be observed for forms:
- The LABEL tag should be used along with ID and FOR attributes (WCAG 1.0 checkpoints 10.2, 12.4).
- The button (or the image placeholder) used for validation purposes must have explicit alternate text (WCAG 1.0 checkpoint 1.1).
- Verify that all mandatory fields are clearly identified, and inform the user about all the data that is being sent to the server (even if scripts are disabled) (WCAG 1.0 checkpoints 6.3, 6.5).
- **Browsing aids -** If keyboard shortcuts are set up for a page, these shortcuts should be activated and homogeneous throughout the pages of the Web site (WCAG 1.0 checkpoint 13.4).
- **Accessible content -** Do not refresh screen content automatically (WCAG 1.0 checkpoint 7.4).
- **Content accessibility check -** Validate. This issue is developed in the next section below.

Beyond this list of key concepts to obverse to write good core (X)HTML, we advice the reader to consult the Writing Accessible Web Applications whitepaper that gives recommendations based on W3C recommendations (see eponym section 1.3 W3C recommendations), Microsoft guidelines, and user feedback as well as an understanding the requirements of AT devices (see section 1.2.2 Assistive technologies).

Interestingly enough, the specific information and examples can be applied to simple (X)HTML, complex DHTML, and Asynchronous JavaScript + XML (AJAX) applications. This information can be seen as a cascading or layered approach, with the more fundamental requirements discussed first. The guidelines fall into the following general categories:

- *(X)HTML issues* – Use semantically correct HTML. As previously mentioned, you need to focus on correct use of roles and names, and always supply alternative text for visual elements. Many examples are given in particular with respect to the menus.
- *Dynamic HTML (DHTML) issues* – You need to maintain source order and use elements and events with built-in action semantics.
- *Asynchronous JavaScript And XML (AJAX) issues* – AJAX is DHTML with asynchronous updates. When you use AJAX, you need to notify AT that the page has been updated.

The related information provided in this guide can used in conjunction with ASP.NET AJAX, to create accessible pages with responsive UI and more efficient client-server communication.

ASP.NET AJAX integrates cross-browser client script libraries with the ASP.NET 2.0 development framework. Developers targeting other server frameworks can take advantage of the Microsoft AJAX Library that is supported by most popular browsers and can be used to build client-centric Web applications that integrate with any backend data provider.

Furthermore, during the course of the development of the accessible Web content, you can use the Internet Explorer Developer Toolbar to confirm that the appropriate HTML element type, for example, <H3>, <UL>, and so on, is used.

If you select *View Source* from Internet Explorer, you can view the HTML as it was downloaded to the browser. However, if you use the developer toolbar, you can view the DOM after it has been manipulated by script.

This toolbar is also valuable as a debugging and testing tool. You can use it to inspect the tag name, HTML attributes, and cascading style sheet (CSS) properties of elements. The toolbar has other useful features for testing your work, such the ability to quickly view CSS or alternate text for images and validators. As far as the latter point is concerned, it provides an automated access to online line validation services discussed in the next section: Markup Validation Service, CSS Validation Service, and Link Checker from W3C for respectively HTML, CSS and Link validation. Content accessibility checking is also provided through the use of the HiSoftware Cynthia Says portal.

2.3 Assessment and validation of accessible Web Content

As indicated in the preceding sections, validation of Web pages and Web site content should begin very early in the project, right from the design phase. Assessment and validation can and should take place as soon as accessibility functions are implemented on the Web site.

Existing Web pages can also be assessed for accessibility errors.

2.3.1 Approach and assessment methods

- Assessment methods and tools need to be identified in the testability framework defined during the project design phase. The methods adopted must provide not only the best possible coverage of WCAG 1.0 checkpoints, but also the best level of automation.
- As discussed in the W3C/WAI article on tool selection, Selecting Web Accessibility Evaluation Tools, Web accessibility evaluation tools can ease the burden of assessing compliance with accessibility guidelines. However, tools cannot replace human judgment, and some validations require manual evaluation by an expert.

The following WCAG 1.0 checkpoints can be fully automated: 1.1, 1.3, 1.4, 2.1, 5.5, 6.1, 6.3, 6.5, 7.1, 8.1, 9.1, 10.1, 12.1, 13.2, 13.4

The following WCAG 1.0 checkpoints can be partially automated and then validated by an expert: 1.2, 4.1, 5.1, 5.2, 5.3, 5.4, 6.2, 6.4, 7.2, 9.2, 11.4, 13.1

Appropriate tools can be selected based on the article mentioned above, Selecting Web Accessibility Evaluation Tools. Also available on the W3C/WAI Web site are the following:

- A list of more than 30 tools that can be used to assess, reconfigure, repair, filter or transform Web sites, at Web Accessibility Evaluation Tools: Overview
- The WAI HTML Table Linearizer Entry Form, from which you can run the *tablin* tool to linearize HTML tables and render them according to preferences set by the presentation layer (screen reader)

We reference several of these tools for the purpose of illustration in the following sections.

2.3.2 W3C/WAI approach to Web site accessibility evaluation

This section provides a high-level overview of accessibility evaluation activities based on the W3C/WAI approach.

The W3C/WAI approach does not describe how to systematically evaluate every checkpoint, but it does provide some general procedures and methodologies for determining the general level of conformance of a Web site.

Accessibility evaluation can be approached at two levels: a preliminary review, which can quickly locate possible accessibility issues, and a conformance evaluation, which evaluates a Web site against standards to determine the level of conformance. These two approaches are described briefly here. For more in-depth information, see the Evaluating Web Sites for Accessibility: Overview section of the W3C/WAI Web site.

2.3.2.1 Preliminary Review

This approach does not look at every accessibility issue or specific standards, but it provides a way to quickly get a sense of the nature of possible problems on an existing Web site. The steps involved in this approach are as follows:

1. Select a sample page that includes representative elements from your Web site.
2. Use a graphical browser to evaluate the page, modifying various settings.
3. Use a specialty browser, such a voice or text-based browser, to view the page.
4. Run two or more automated tools against the selected pages to find out what type of problems are uncovered.
5. Summarize and report your findings with recommendations for next steps.

Some toolbars available for download offers many functions to help you manually assess your Web pages. We already have mentioned the Internet Explorer Developer Toolbar.

You can also download the free Web Accessibility Toolbar, developed by the Accessible Information Solutions (AIS) team of Vision Australia, and use its many functions to help you manually assess your Web pages.

The toolbar provides features that can be used in Internet Explorer to help you identify structural and semantic components in a Web page, including, but not limited to, the following:

- Presence or absence of alternate text linked to graphical elements
- Availability of information about links (title, URL, content of TITLE attribute, and so forth)
- Accessibility of information about frames
- Presence of event handlers
- Page structure using H tags
- Linear structure of page

The toolbar includes documentation that describes how to use it for assessing WCAG 1.0 compliance. It can be used to check for many of the 65 checkpoints covered in the WCAG 1.0 guidelines. The functions do not explicitly indicate whether a Web page complies with a specific WCAG checkpoint, but they provide information you can use to assess the compliance of the page.

The toolbar is also based on the Job Access With Speech (JAWS) screen reader, which makes it possible to read information displayed on the screen by using text-to-voice software or a Braille terminal. A demo version of JAWS is available. This is an unlimited version and provides access to all software features. It can be used in 40 minutes sessions. After 40 minutes of usage, you must restart the computer to continue using JAWS.

2.3.2.2 Conformance Evaluation

This approach assesses a Web site against specific accessibility standards, such as the WCAG 1.0 checkpoints. The approach uses automated, partially automated, and manual methods. The general steps involved in this approach are as follows:

1. Determine the scope of your evaluation: identify sample Web pages and the level of conformance to be evaluated.
2. Run markup validation tools and accessibility evaluation tools against the selected sample pages and identify problems. For more information about markup validation and accessibility evaluation tools, see Conformance Evaluation of Web Sites for Accessibility.
3. Using the checklist of WCAG 1.0 checkpoints and various types of browsers, manually review the sample Web pages. Include a review of page content.
4. Summarize and report your findings, including the problems uncovered, recommendations, and best practices.

2.3.2.3 Contextual spot checks: images and colors

- Verify that a user can interpret the content on the Web page even if the browser or screen reader cannot interpret images. To perform this test, disable image display in the browser. In Internet Explorer, on the *Tools* menu, click *Internet Options*. In the *Internet Options* dialog box, click the *Advanced* tab. Scroll to the *Multimedia* group, and clear *Show Pictures*. After the system is configured, test the Web site to make sure that it can still be browsed.
- Check that the colors on the Web page have enough contrast to be distinguishable by users who have impaired color vision. The Color Contrast Analyser 1.0, which is a free download from Vision Australia, can verify foreground and background color combinations to find out whether the colors provide good color visibility, as determined by algorithms provided by the W3C.

2.3.2.4 Links and scripts validation

- Check that all links are valid. The online Link Checker tool from W3C, for example, can check that all links on a Web site home page are active to a depth of several levels.
- Check that scripts do not cause navigation problems. Although various types of scripts (JavaScript, VBScript, ActiveX controls, and so forth) improve interactivity on Web sites, these elements sometimes are not visible to the user (for example, the appropriate plug-in is not installed, scripts are disabled, or a legacy browser is used). Verify that scripts do not interfere with browsing if they are disabled. A tool such as Snoop, an online tool from the Bartimeus Accessibility Foundation, can simulate a browser that has functions such as scripts, style sheets, images, and frames disabled so that you can determine how the page works without them.

2.3.2.5 Unplug your mouse

- Check keyboard accessibility. You should be able to access all pages on the Web site by using only a keyboard. To perform this test, navigate through the site by using only the keyboard and the main shortcuts (such as TAB, SHIFT+TAB) and without using the mouse. This test also verifies the tab order and verifies that all links are working correctly.

2.3.3 Testing platform and common scenarios

To validate a Web site for the accessibility issues covered in this document, you need an accessibility testing platform that includes:

- Syntax validation (for example, (X)HTML and XML). Use a tool such as Markup Validation Service from W3C, or use the verification features included in the Visual Studio 2005 environment.
- Style sheet validation (including CSS). Use a tool such as CSS Validation Service from W3C.
- Contrast validation. Use a tool such as the Colour Contrast Analyser 1.0 from Vision Australia.
- Automatic validation of Web pages and browsing features. Use an automated validation tool that is based on accessibility standards and a browsing assessment tool.
- Please note that software tools do not provide assessment capabilities for some types of accessibility criteria (for example, relevance of text used in links or relevance of alternate text).
- Assessment tools typically support testing for WCAG 1.0 checkpoints (priorities 1, 2, and 3) or section 1194.22 of section 508 Voluntary Product Accessibility Template (VPAT).

The accessibility assessment tools that are included in the Visual Studio 2005 development environment provide Web site validation for WCAG 1.0 priority 1 and 2 checkpoints and section 508 requirements.

Microsoft Expression Web and Microsoft Office SharePoint Designer 2007 both provide the same level of validation. These are new products both based on Microsoft Office FrontPage 2003 technologies. On one hand, Expression Web is a professional design tool to create sophisticated standards-based broad reach HTML Web sites that deliver compelling user experiences. On the other hand, Office SharePoint Designer 2007 enables solution creator and content author to develop applications and solutions on top of the SharePoint platform.

- We recommend the joint use of both Visual Studio 2005 and Microsoft Expression Web/Office SharePoint Designer 2007, because their functional coverage is slightly different. By using both tools, you can combine and compare the assessment results.
- In addition to the verification features that are included with Visual Studio 2005 and Microsoft Expression Web/Office SharePoint Designer 2007, you can use tools such as the following:
- The WebXACT online assessment platform from Watchfire (or its purchasable, downloadable equivalent Bobby).
- The free downloadable tool A-Prompt, or ATRC's new online replacement for A-Prompt, Web Accessibility Checker.

For validating links, you can use online tools such as Link Checker from W3C or Link Validation Utility from HiSoftware.

- Use both recent and legacy releases of popular graphical browsers, such as Internet Explorer and Firefox, to assess and validate Web pages using the following scenarios:
- Disable graphics and check that the information is presented in an appropriate sequence for the graphical layout of the site.
- Disable multimedia elements, such as sound, and ensure that the audio content is available through text equivalents.
- Disable style sheets, and ensure that the information is presented in an appropriate sequence for the graphical layout of the site.
- Modify font sizes (both larger and smaller), and check that the page is still readable.
- Set the screen resolution to 640 x 480, and verify that the page can be displayed by using horizontal scrolling.
- Switch the screen to black and white (or print the page in black and white), and check that the color contrast is appropriate.
- Use tabs and keyboard shortcuts to navigate through the links and form controls, and check that all the links and controls are accessible and that each link target is clearly indicated.
- Disable scripts, ActiveX controls, and so forth, and ensure that equivalent navigation facilities are provided.
- Use a text-only browser, such as Lynx Viewer, to check the following issues:
- Verify that the same information is available with the text-only browser as with a graphical browser.
- Verify that the information is presented in the same logical order with the text browser as with a graphical browser.

Note: An experienced tester can test with a "self-voicing" browser or a screen reader instead of a text-only browser. A visually impaired tester requires the presence of a sighted assistant to compare the graphics on the Web site to the text equivalents. Sighted testers need to listen to text with their eyes closed, and then check that the information provided is equivalent.

- Use a grammar and spell checker. A person who uses text-to-voice software to access a Web page might not be able to select the best version provided by the synthesizer if the text has spelling or grammatical errors. Eliminate all spelling and grammar errors to improve comprehension.

- Integrate content review by testers who have disabilities.

To improve the accessibility features of future Web sites, consolidate all the assessment results, summarize the problems encountered during testing (include the test method that identified each accessibility problem), and list the best practices acquired during the testing process. You can then apply these lessons learned on an ongoing basis and build on them during future testing. Maintaining these results makes it possible to develop an Accessibility knowledge base.

3. Accessibility guidelines for using Microsoft Content Management Server 2002 for Web implementations

Microsoft Content Management Server (MCMS) 2002 is a Web content management solution that provides a full range of dynamic and customizable features for implementing Internet, intranet, and extranet Web sites. For example, you can do the following by using MCMS:

- Build dynamic and scalable Web sites that use MCMS controls (based on the MCMS guidelines) by using the power of the Visual Studio 2005 development environment[8] (and, in part, ASP.NET 2.0 technology[9]), Microsoft SQL Server 2005[10], and standards such as SOAP and XML.
- Deploy Web sites and Web content from one server to another by means of a flexible and extendable content object packaging mechanism that uses XML.
- Enable users who have a minimal understanding of the technical aspects of Web site development to create, modify, publish, and manage their own content by using user-friendly tools that are based on an Internet browser or Microsoft Word.
- Enable authorized content creators to create or modify multiple parts of a single Web site simultaneously by separating content from format.

For maximum flexibility, MCMS stores content in XML, HTML, and binary objects in a Microsoft SQL Server repository. Content is thus managed separately from design elements or templates that are used to produce the Web site. Web site templates (ASPX, ASP, or ASCX source files) can be used to define the overall appearance for a set of pages in a MCMS Web site and can contain executable code.

These templates and the executable code are stored at the file system level, and can be managed by using a source code management system, such as Microsoft Visual Source Safe or Microsoft Visual Studio Team System (Team Foundation Source Control).

Because MCMS integrates code dynamically in a Web site through template objects, Web pages can be produced and published dynamically. In this way, content can be customized as a function of user profile and behavior, Internet browser, or default language. Managing objects separately makes it possible to rapidly adapt content as a function of a variety of applications employed by the organization and outside partners.

Lastly, a connector, the Microsoft Content Management Server 2002 Connector for SharePoint Technologies, makes it possible to integrate SharePoint technologies (see chapter 3 "Accessibility guidelines for using SharePoint 2003 technologies for Web implementations") with MCMS 2002 to provide an end-to-end solution for document publication.

For additional information about MCMS, see http://www.microsoft.com/cmserver/default.mspx.

3.1 Evaluation of MCMS 2002 Web sites for WCAG 1.0 checkpoints

This white paper evaluates the use of MCMS for creating Web sites that comply with the priority 1 checkpoints defined for the 14 guidelines in the Web Content Accessibility Guidelines (WCAG) 1.0, which were developed by the W3C/WAI and published 5 May 1999 (see section

[8] MCMS 2002 Service Pack 2 supports Visual Studio 2005.

[9] MCMS 2002 Service Pack 2 supports the following ASP.NET 2.0 features: Authorization Providers, Navigation Providers, and Master Pages. The other features of ASP.NET 2.0 are not supported.

[10] MCMS 2002 Service Pack 2 supports Microsoft SQL Server 2005.

1.3.1 "WCAG 1.0"). Sometime after WCAG 2.0 is approved and released, a future version of this document will evaluate compliance with the guidelines in the new release.

To evaluate the use of MCMS 2002 for creating compliant Web sites, this document considers the following two areas:

- Template creation by using a template designer. This area includes the browsing mechanisms that are set up for the Web site.
- Content, that is created by users who have been granted appropriate author or editor permissions. The Web content is created for each page by using *placeholders*.

MCMS 2002 is evaluated for each checkpoint with respect to its content authoring tools (as they exist without development) and, if applicable, the possibility of workarounds to comply with the checkpoints.

As mentioned in the introduction, this assessment takes into consideration the Voluntary Product Accessibility Template (VPAT) for MCMS 2002.

The MCMS 2002 release examined is the latest available as of the publication date of this white paper: Microsoft Content Management Server 2002 with Service Pack 2.

Service Pack 2 requires prior installation of Service Pack 1a, which in turn requires prior installation of Internet Explorer Web Controls for Content Management Server 2002 with Service Pack 1a. We suggest that you begin by reading the following articles:

- Content Management Server 2002 Service Pack 2 installation information
- Rational Guide to installing MCMS 2002 Service Pack 2

MCMS 2002 Service Pack 2 supports .NET Framework 2.0 and the Visual Studio 2005 development environment. It should be noted that Visual Studio can be used to develop XHTML-compliant templates that are automatically validated during the design process.

By default, Visual Studio 2005 is configured to check Web pages for Internet Explorer 6.0 requirements. If you wish to check Web pages for a specific XHTML scheme, you need to select one of the XHTML schemes by selecting *Tools* ⇨ *Options* ⇨ *Validation*.

Validation errors, such as unclosed statements, are indicated in the Source view by a red wavy line under the content where the problem occurs. Syntax errors, such as the use of an obsolete marker, are underlined with a green wavy line. Explanatory tips are displayed when the mouse pointer hovers over a syntax error. These messages can also be displayed in the error list (*View* ⇨ *Other Windows* ⇨ *Error List*).

By using the new Accessibility Validation tool in Visual Studio 2005, you can validate your Web pages against several different accessibility standards. Access the tool by selecting *Tools* ⇨ *Check Accessibility*. You can use this function to validate your Web pages against WCAG Priorities 1 and 2 and Section 508 directives. You can also display the validation results in the error list.

See the MSDN article Building ASP.NET 2.0 Web Sites Using Web Standards, which describes these features in detail.

3.1.1 Graphics

1.1: Does each graphic have text to display as an alternative to the graphic?

In HTML, use the ALT attribute with IMG and AREA SHAPE elements, and so forth. It is important to provide alternate text for graphics because some browsers and screen readers cannot interpret or display graphics properly.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_01, 1.1_HTML_07, 1.1_HTML_08

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

The `ImagePlaceholder` control that is available in the standard version of MCMS 2002 enables content authors to insert images that have an assigned ALT attribute.

Similarly, the `HTMLPlaceholder` control that is available in the standard version of MCMS 2002 enables content authors to insert images that also can have an ALT attribute. It is the responsibility of Web masters and content authors to ensure that their Web pages comply with this checkpoint.

1.2: Is the alternate text for each image relevant to the context in which the image is viewed?

A given image might have several meanings. It is the context of the image that determines the exact meaning. For example, a yellow bullet icon in a newspaper Web site should have the ALT attribute set to ALT="" or ALT="*". However, if the same yellow bullet icon is included in a list of graphics that is available for download, it might make more sense to set the attribute to ALT = "yellow diamond" to distinguish it from other graphics available on the page.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_03

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 1.1 in this section. This criterion is subjective.

1.3: Are graphics that are used only for decorative purposes commented with ALT=""?

Some graphics have no informative value. An example is the use of transparent pixels only for layout purposes. However, all graphical elements must be commented. For this reason, graphics that are used only for decorative or layout purposes should be commented by setting the ALT attribute to null (in HTML, ALT=""). Some browsers do not display the images that use this type of comment.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_02

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 1.1 in this section.

1.4: Is the alternate text for each image no more than 60 characters long?

You should use a maximum of 60 characters in the ALT attribute because alternate text might bother some users. This is often the case for people who use a screen magnification function because of visual disabilities. It is important to limit the number of characters in the image comments so that such users can minimize the effort required to obtain all the information.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1

WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 1.1 in this section.

1.5: Are all comments that are linked to clickable areas of a MAP image relevant?

Definition: A MAP image is an image that presents clickable areas so that the user can click to select an option. When you use a MAP image in HTML (on the client or the server side), it is important to include alternate text for all clickable areas, as well as for the MAP image itself (that is, the image that contains the USEMAP or ISMAP attribute).

For a MAP image on the client side, provide alternate text by using the ALT attribute in each of the clickable areas (AREA SHAPE).

For a MAP image on the server side, provide redundant text links that repeat the content of each of the clickable areas.

Comment 1: The alternate text provided for the image that contains the USEMAP or ISMAP attribute must not be null.

Comment 2: It is recommended that MAP images be used on the client side rather than on the server side.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1, 1.2, 9.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_03, 1.2_HTML_01, 9.1_HTML_01

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

Content authors cannot create map images on the client or server side by using the standard publishing controls in MCMS 2002.

1.6: Is the alternate content for each text image at least the equivalent of the text appearing in the image?

It is not recommended that you present text in the form of an image. However, if there is an unavoidable graphical constraint (for example, non-equivalent CSS rendering), the alternate text for a graphical element containing text should, at minimum, present the exact content of the text itself. The alternate text can provide additional information, but it must at least provide the exact content.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_03

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 1.1 in this section.

1.7: Do all images that require a detailed description provide comment text?

When a graphical element requires a long description in HTML (for example, a complex image or alternate text longer than 60 characters), a text comment must be provided. The comment can be a link to a descriptive file (for example, a D-link or a clickable graphic), a caption that appears directly on the page, or a LONGDESC attribute.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_05

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

The `ImagePlaceholder` and `HTMLPlaceholder` controls that are provided in the standard version of MCMS 2002 do not support the use of the LONGDESC attribute.

The controls can be modified to support this feature.

Another possible approach is to use a WYSIWYG content editor control instead. The r.a.d.editor Lite Edition, developed by Telerik and available without additional cost for all MCMS 2002 clients, makes it possible to create and manage rich, accessible HTML content. The content generated by r.a.d.editor is accessible if the author fills in all the required fields (identified by a special accessibility icon). Any content element that is more complex than simple text provides accessibility fields that support a complete description, including long description support.

It should be noted that this software publisher also offers a full range of controls for MCMS 2002. These controls are available in the r.a.d.controls suite MCMS Edition. This suite offers, in addition to other features, the r.a.d.editor Premium Edition control, which is an extended version of the control mentioned above.

It is the responsibility of Web masters and content authors to ensure that their Web pages comply with this checkpoint.

The WCAG directive specifies that the LONGDESC attribute should be used, but currently most browsers do not support this attribute. Internet Explorer 6.0 exposes this attribute from a programming point of view in its object model. Assistive technologies might thus make use of this feature. Longdesc Linker for Internet Explorer 6 is a free downloadable ActiveX component that makes it possible to add a Long Description element to the context menu used by Internet Explorer for images. This element is enabled if the selected image has a LONGDESC link. If the user selects the link (by holding down the SHIFT key while selecting), Internet Explorer follows it.

1.8: If a detailed description is provided for an image, is the content relevant?

The content should provide additional information to the description defined in the ALT attribute for the image.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_04

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 1.7 in this section. This criterion is subjective.

1.9: Does the text used in the ALT attribute for each image provide the function of the link?

The image description is not always in keeping with the function of the link. For example, if an arrow is used to indicate a link, use ALT="Next" instead of ALT="Right Arrow". Avoid using ALT="", which provides no information about the function of the link.

WCAG 1.0 checkpoint(s) [Priority 1]:
WCAG 1.0 checkpoint(s) [Priority 2]: -13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_02

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 1.1 in this section.

3.1.2 Frames

2.1: Does each frame have a NAME attribute?

In HTML, each frame must be named by using the NAME attribute. Browsing through frame-based Web pages can be complex, especially for visually impaired people who become familiar with site content frame by frame (whereas sighted people have an overview of the entire site). Comment: the content of the NAME attribute should not include any spaces.

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

We do not recommend the use of frames in templates. If, however, frames are used, the Frameset Document Type Definition (DTD) for XHTML 1.0 should be used (see 8.1 under "Mandatory elements").

Content authors cannot manipulate frames in MCMS 2002.

2.2: Are the names assigned to frames relevant?

The names assigned to frames should be relevant. Avoid naming frames with directional data, such as "Left," "Right," or "Middle." It is better to name frames with explicit names that refer to the function provided by the frame, such as "Menu," "Navigation," or "Content."

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: See 2.1 in this section.

2.3: Is there a NOFRAME tag?

In HTML, the NOFRAME tag is used to provide the user with a summary of the main links on the Web site. This tag facilitates browsing for browsers that do not handle frames.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 6.5
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.5_HTML_02

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: See 2.1 in this section.

2.4: Is the content of the NOFRAME tag relevant?

If a NOFRAME tag is present, the information in the tag must be relevant. In many cases, the NOFRAME tag is present but provides information that is either inappropriate or incomplete. This tag is intended to provide the main links existing on the Web site.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 6.5
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.5_HTML_02

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: See 2.1 in this section.

2.5: Does each frame have a TITLE attribute?

In HTML, the TITLE attribute is used to describe, in a few words, the purpose of the frame. This attribute is complementary to the NAME attribute.

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 12.1_HTML_01

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: See 2.1 in this section.

2.6: Is the content of the TITLE attribute relevant?

Some assistive technologies can recognize this attribute. The content of this attribute helps the user navigate from frame to frame independently and quickly determine the function of each frame.

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 12.1_HTML_02

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: See 2.1 in this section.

2.7: Does each page have a maximum of three frames?

It is important to have no more than three frames on any one page. If more than three frames are used, the page becomes difficult to read, especially for users of assistive technologies.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: See 2.1 in this section.

2.8: When frames are used, is scrolling automatic?

In HTML, a frame can no longer be scaled if scrolling is set to "NO." The scrolling function must be enabled. For users of magnification software, which might implement user-specific style sheets with larger fonts, and so forth, in-frame scrolling must be enabled to allow all content to be accessed.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 14.3

UWEM 1.0 Test(s): -

Type of criterion: design / development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: See 2.1 in this section.

3.1.3 Colors

3.1: Is information provided by color still readable when colors are disabled?

Web site designers must be particularly careful about information provided by colors. This information can be lost to color-blind users or users who do not have adequate visual acuity to distinguish between certain colors.

WCAG 1.0 checkpoint(s) [Priority 1]: 2.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 2.1_HTML_01, 2.1_HTML_02, 2.1_HTML_03, 2.1_CSS_01

Type of criterion: design

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: Template developers and content authors are able to satisfy this criterion. It is their responsibility to ensure that the HTML code they produce does so.

Content authors must be aware of this criterion so that it is handled appropriately.

3.2: Is there enough contrast between colors to be distinguishable by users who have impaired color vision?

Some color combinations should be avoided, such as red with blue, yellow with violet, yellow with green, as so forth. When there is not enough contrast between colors on a page, the information provided by contrast might not be accessible to users with poor or impaired color vision.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 2.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 2.2_HTML_01, 2.2_CSS_01

Type of criterion: design

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See comments for 3.1 in this section. A link can be provided to a high contrast version of the Web site. CSS style sheets must be used so that users can automatically apply a different style sheet when they read Web site content. This configuration can be saved for a Web page by using cookies.

Additionally, content authors and graphics designers must be aware of the difficulties in using media where this problem can occur. Graphics designers should consider this criterion when they create images. All images in the resource repository should comply with this criterion.

To assess this criterion, a color contrast evaluation tool, such as the free downloadable software by Vision Australia, can be used to verify foreground and background color combinations to determine whether the color contrast enables proper viewing. The identification of proper color visibility is based on the algorithms provided by the W3C.

3.1.4 Multimedia

4.1: Can the information that is conveyed by multimedia be provided another way?

Whenever multimedia is used to provide information, it is important to provide an alternative for those who do not have access to multimedia. For example, if a multimedia file is made available on a Web site, it is important to provide an accessible text-based alternative in a standard format, such as RTF, TXT, or HTML.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.3_external_01, 1.3_external_02, 1.1_HTML_01, 1.1_HTML_02, 1.1_HTML_03, 1.1_HTML_04, 1.1_HTML_05, 1.1_HTML_06, 1.1_HTML_09, 1.1_HTML_10, 1.1_external_01, 1.1_external_02, 6.3_HTML_01

Type of criterion: design / development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: Content authors cannot add or manage multimedia by using the standard publishing controls in MCMS 2002.

4.2: Is the Multimedia content synchronized with the alternate support?

If a video with sound is used on a site, as in the case of an interview between several people, it is important to synchronize the alternate support with the picture and the sound in the video. The W3C recommends the use of the Synchronized Multimedia Integration Language (SMIL) programming language.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.4, 1.3
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.4_external_01

Type of criterion: design / development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: This criterion requires the use of a tool, such as Hi-Caption from HiSoftware, which is a Synchronized Multimedia Integration Language (SMIL) presentation editor, or Microsoft Synchronized Accessible Media Interchange (SAMI). Please refer to the technical articles Understanding SAMI 1.0 and Adding Closed Captions to Digital Media for additional information about how to add subtitle tracks to the Windows Media format.

You might want to refer to HiSoftware Hi-Caption Studio - Creating Accessible Demos as an illustration.

3.1.5 Tables

5.1: Is the SUMMARY attribute present and relevant?

If the SUMMARY attribute is used correctly, it provides additional information to disabled users, such as the visually impaired, about the tables in a Web page. If a Web page contains menu bars that are created by using an HTML table, setting SUMMARY="Menu bar" provides the user with additional information about the table function.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 5.5

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

For author content, the `HTMLPlaceholder` control that is provided in the standard release of MCMS 2002 does not enable this functionality. A customized control can be developed that inherits from the original `HtmlPlaceholder` control to provide this support.

The best approach is to use a WYSIWYG content editor control instead. The r.a.d.editor Lite Edition, developed by Telerik and available without additional cost for all MCMS 2002 clients, makes it possible to create and manage rich, accessible HTML content. The content generated by r.a.d.editor is accessible if the author fills in all the required fields (identified by a special accessibility icon). Any content element that is more complex than simple text provides accessibility fields that support a complete description.

It should be noted that this software publisher also offers a full range of controls for MCMS 2002. These controls are available in the r.a.d.controls suite MCMS Edition. This suite offers, in addition to other features, the r.a.d.editor Premium Edition control, which is an extended version of the control mentioned above.

5.2: In a data table, does the CAPTION tag provide the title of the table?

Rather than using formatted text to provide the title of a table, it is better to use the CAPTION tag, which is the tag dedicated to displaying table titles.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 5.5

UWEM 1.0 Test(s): 12.3_HTML_07, 12.3_HTML_08, 5.4_HTML_04

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 5.1 in this section.

5.3: In data tables, are the column headers appropriate?

In HTML data tables, the TH tag specifies column headers, enabling users of assistive technologies to use text-to-voice software to determine that they are column headers.

WCAG 1.0 checkpoint(s) [Priority 1]: 5.1
WCAG 1.0 checkpoint(s) [Priority 2]: 5.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 5.1_HTML_01, 5.4_HTML_01, 5.4_HTML_01

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 5.1 in this section.

5.4: In a data table, does a HEADERS attribute link to each of the data cells in the table?

When correctly configured in a data table, the HEADERS and ID attributes make it possible to link each data cell to the appropriate heading cell. This approach enables users that have only a linear view of the table to know which header is linked to the current data cell.

WCAG 1.0 checkpoint(s) [Priority 1]: 5.2
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 5.2_HTML_01, 5.2_HTML_02, 5.4_HTML_03

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 5.1 in this section.

5.5: Is the content in formatted tables in correct sequence?

In formatted tables, the information must be ordered so that it is displayed in an equivalent sequence when the information is read in a linear fashion. Care must be taken when organizing information in a formatted table. HDL development tools that are currently available have a tendency to nest tables within one another. The result is that occasionally differences occur between the visual appearance of information in a graphical browser and the text equivalent, because the latter is based on the structure of the HTML code (as in LYNX-type text browsers, for example).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 5.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 5.3_HTML_01

Type of criterion: design

Can be accomplished by using templates: Yes
Can be accomplished by using author content: No

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

Content authors must be aware of this criterion, even if they are dependent on the features available in their publishing software.

3.1.6 Links

6.1: Are Link titles no more than 80 characters long?

Use a maximum of 80 characters in hypertext link titles (accents and spaces count for one character).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: design / development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

This criterion also applies to content authors when they use the `HtmlPlaceHolder` control. It is the responsibility of Web masters and content authors to ensure that the Web pages produced comply with this criterion.

6.2: Are links explicit enough?

Links must be understandable independent of the page context. Some assistive technologies are able to display a list of the available links. These links must be understandable without the context of the Web page to enable rapid scanning. Avoid links such as "click here."

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_02

Type of criterion: design / development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 6.1 in this section. This criterion is subjective.

6.3: Is the TITLE attribute used, if required, and is it no more than 80 characters long?

In HTML, the TITLE attribute is necessary if additional information is required to understand a link. For example, if a link is used to download a file, the TITLE attribute can provide the file name, file size in bytes, file format, and destination ("open in new window").

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 6.1 in this section.

6.4: Does the TITLE attribute provide more information about the link than the link title itself?

The only information that some users have about a link is the title of the link or the content of the TITLE attribute. Therefore, the content of the TITLE attribute should, at the very least, repeat the information contained in the link title. For example, a TITLE attribute that simply indicates "this link opens in a new window" is not adequate, because the user does not know the link destination. However, "Read article on tourism in China (new window)" provides all the essential information required about the link.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_01, 13.1_HTML_02

Type of criterion: - development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 6.1 in this section.

6.5: Do all identical link titles lead to the same target?

Avoid using link titles such as "Read article" multiple times on a page. It is important to use link titles that are not identical to help people who wish to move from link to link to read the page more quickly. It is also important to use short, explicit links with unambiguous destinations.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_01

Type of criterion: design

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: See 6.1 in this section.

3.1.7 Scripts

7.1: If a script requires alternate text to make it accessible, is the information provided by the alternate text equivalent to the information provided by the script?

The information about a script should remain available even if the script is disabled. For example, if a JavaScript script is used to validate an online form, the alternative might be a server-side data entry control. Scripts can be disruptive to users of assistive technologies that do not support scripts or that support them only in a limited fashion. In addition, some users voluntarily disable script execution in standard browsers. For this reason, an alternative must be available to retain the information generated by the scripts.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.2, 6.3, 8.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.2_HTML_02, 6.2_HTML_03, 6.3_HTML_02

Type of criterion: design / development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

Most .NET ASP controls assume that the Web client supports scripts (the "postback" concept in ASP.NET). For example, when a user clicks on a .NET ASP button that has an associated OnClick event, it causes the browser event handler to execute a script function that, in turn, sends the HTML form.

In such cases, .NET ASP controls that require this function should not be used; only basic HTML controls and elements should be used. For this reason, data must be entered by using query strings, or by submitting the online form without scripts, using HTML Submit buttons.

If scripts are used to present content interactively, a text equivalent should be provided by placing static text within the elements.

<object> and <applet> tags must also provide alternate text for the element content.

It is not possible to use standard MCMS controls in the MCMS 2002 author interface to add this information to this type of object. Customized placeholders must be developed if these objects are required.

7.2: Can actions be performed even if the peripheral for which they were designed is disabled?

Some functions of a Web site, such as those generated by JavaScript scripts, can be activated only by a mouse click. However, some users cannot use a mouse to browse through a Web site. It must be possible to browse through a Web site by using just a keyboard.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 6.4, 9.2, 9.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 9.2_external_01, 6.3_HTML_02, 6.4_HTML_01, 6.4_HTML_02, 6.4_external_01, 8.1_HTML_01, 8.1_external_01, 9.3_HTML_01

Type of criterion: design / development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

Nonspecific event handlers such as `onFocus=""` or `onBlur=""` are preferred to event handlers that are triggered by a specific event such as `onbeforedeactivate=””`.

The standard version of the MCMS 2002 author interface does not support this function.

3.1.8 Mandatory elements

8.1: Is the DOCTYPE tag present at the beginning of the page source code?

International WAI guidelines suggest using an appropriate and valid Document Type Definition (DTD) to indicate the type of electronic document that the browser must load (frame-based, HTML, XML, and so forth). In HTML, the DOCTYPE tag is required to define the document type.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 3.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 3.2_HTML_01

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: This criterion does not depend on the author interface because it is always defined at the template level.

Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

For template developers, we recommend using XHTML 1.0 Transitional.

Please note the following points:

- Visual Studio 2003 modifies the HTML code when you switch from Design view to HTML view. The resulting HTML code that is modified by Visual Studio might violate some rules. For example, HTML tags in lowercase letters might switch to uppercase (uppercase tags do not comply with XHTML 1.0 Transitional). Quotation marks in some HTML attributes might be removed. To prevent some of these modifications from occurring, disable "Apply Automatic Formatting – when switching from design to HTML view" in the Visual Studio options. Visual Studio 2005, which is supported by MCMS 2002 Service Pack 2, no longer behaves in this way.
- In the past, the `HtmlPlaceHolder` control violated several XHTML 1.0 Transitional syntax rules. These problems were corrected in Service Pack 1a, as mentioned previously. For additional information, refer to article 821969 MCMS 2002 ActiveX Control Is Not Compliant with HTML 4.01. The WYSIWYG content editor control, r.a.d.editor Lite Edition from Telerik, presents no problems with syntax validation.
- In the past, the HTML code generated by the console included a <javascript> tag but not the linked attribute, which does not comply with XHTML 1.0. This problem was corrected in Service Pack 1a, as mentioned previously. For additional information, please refer to article *821969 MCMS 2002 ActiveX Control Is Not Compliant with HTML*.
- The RoboMetaTag server control is not XHTML 1.0 compliant. The Microsoft Solution for Internet Business (MSIB) Plus Pack, which can be downloaded from the GotDotNet Web site with the source code, offers an excellent replacement metadata control that is compliant.

Lastly, MCMS 2002 Service Pack 2 provides support for ASP.NET 2.0 Master Pages. When you use Master Pages, you can integrate the appropriate DOCTYPE tags throughout the pages of a Web site. The SiteMaster.master includes a DOCTYPE tag. There is an obvious advantage to specifying the DOCTYPE tag in a Master Page: when the tag needs to be modified in the future, it can be modified in a single location to be applied to all the pages on the Web site. See the MSDN article Building ASP.NET 2.0 Web Sites Using Web Standards for additional information.

8.2: Is the LANG attribute present at the beginning of the page source code to clearly identify the language used?

It is important to specify the language of a document to enable assistive technologies to adapt content to the language. In HTML, the LANG attribute in the HTML tag specifies the language.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 4.3

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: -

8.3: Is there a TITLE tag in the page header?

In HTML, the title of a page is coded by including a TITLE tag in the document header. Voice browsers list the title of the page first so that a user can get positioned on the Web site. The title is a constant source of support to those having an attention deficit disorder. The tag also makes it easier for search engines to reference the Web site.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.2_HTML_01

Type of criterion: development (development rules)

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: -

8.4: Is the content of the TITLE tag explicit?

The TITLE tag provides the title of the page that was just loaded. It is important to provide an explicit title to help users get oriented during browsing.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.2_HTML_01

Type of criterion: design

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: -

8.5: Is the content of the TITLE tag different from one page to the next?

Using an identical title on every page of a Web site can be confusing to users. The title of the page is the first information that appears on the screen. Assigning the same title to multiple documents is misleading.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: design

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: -

8.6: Are language changes on a page indicated?

Whenever a language change occurs on a page, it should be indicated to the user. If text-to-voice software is used, a change in language causes the software detection language to change as specified by the LANG tag.

WCAG 1.0 checkpoint(s) [Priority 1]: 4.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 4.1_HTML_01, 4.1_HTML_02, 4.1_HTML_03, 4.1_HTML_04, 4.1_CSS_01, 4.1_CSS_02

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

The `HtmlPlaceholder` control provided in the standard version of MCMS 2002 does not support the use of LANG attributes in the middle of content in the placeholder. To do this, the author would need to understand the HTML language so that he or she could edit the HTML code for the control to add these attributes. A customized control can be developed to support this feature.

The WYSIWYG content editor control, r.a.d.editor Lite Edition from Telerik, supports the specification of language for all content elements.

3.1.9 Information structuring

9.1: Is information structured consistently for the general context of the site?

The structure of information on a page depends on the type of site (news, gaming, scientific, artistic, and so forth). In addition, the individual elements that provide information on the page (such as images and titles) might not be a problem in and of themselves, but the arrangement of these elements in relation to one another might not be consistent. For example, on a news site, no text should be displayed before the title.

WCAG 1.0 checkpoint(s) [Priority 1]: 14.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 12.3_HTML_10, 12.3_HTML_15, 14.1_HTML_01

Type of criterion: design

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: It is possible to create templates that satisfy this criterion.

Template developers have full control over the (X)HTML used in the template. It is the responsibility of developers to ensure that all (X)HTML code satisfies this criterion.

Similarly, content authors should keep this criterion in mind.

9.2: Is the Web page presented in a consistent fashion?

Use of styles that are available in HTML, for example H1 and H2, can provide structure to a Web page. Such tags are extremely important for structuring a Web page. For example, the H1 tag indicates a title and the H2 tag indicates subheadings. These tags must be used for their intended purpose, that is, to provide structure, and not for their graphic capabilities. Additionally, blocks of information should be separated from each other.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 3.5
WCAG 1.0 checkpoint(s) [Priority 3]: 13.8

UWEM 1.0 Test(s): 3.5_HTML_01, 3.5_HTML_02, 3.5_HTML_03, 3.5_HTML_04, 3.5_HTML_05

Type of criterion: design

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

Content authors must be aware of these issues. They can use the `HtmlPlaceholder` control for content editing and headers.

Nevertheless, the standard `HtmlPlaceholder` control does not support the creation of <hr /> elements unless the HTML code of the control is directly edited. The WYSIWYG content editor control, r.a.d.editor Lite Edition from Telerik, can be used instead.

3.1.10 Information layout

10.1: Is page content separated from content introduction?

If a page is coded in HTML, it should not contain layout information (for example, BGCOLOR, FONT, ALIGN, B, I), because all this information can be included in a style sheet.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 3.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 3.3_HTML_01, 3.3_HTML_02

Type of criterion: development (development rules)

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

Content authors do not have the same degree of control: author content cannot be positioned by using style sheets with the standard `HtmlPlaceholder` control. The WYSIWYG content editor control, r.a.d.editor Lite Edition from Telerik, can be used instead.

10.2: If style sheets are disabled, is the information still accessible?

To separate content from layout on an HTML page, use style sheets (CSS) to specify the layout. However, people who do not wish to read the page by using the author-selected style sheet should obtain the same information when they disable the page style sheet and use their own customized settings instead.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 4.1_CSS_02, 6.1_HTML_01, 6.1_HTML_02

Type of criterion: design / development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: See 10.1 in this section.

10.3: If style sheets are disabled, is the order in which information appears the same as initially defined?

Style sheets can be used to construct HTML pages. If style sheets are disabled by the user, the logical structure of the page should stay the same as initially defined. For example, the order of appearance of the divisions (DIV) should stay the same when viewed by using a graphical browser or by using a text-mode browser.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 9.4

UWEM 1.0 Test(s): 3.6_HTML_01, 6.1_HTML_01, 6.1_HTML_02

Type of criterion: design / development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: See 10.1 in this section.

3.1.11 Forms

11.1: Are the LABEL tag and its corresponding attributes (ID, FOR) present?

In HTML pages, the LABEL tag is used with the ID and FOR attributes to link data entry fields directly to the associated text. This approach enables people who use assistive technologies to unambiguously determine which data entry field corresponds to the linked text.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 10.2, 12.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 10.2_HTML_01, 12.4_HTML_01, 12.4_HTML_02

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

Content authors cannot use MCMS 2002 editing controls (placeholders) to create or manipulate forms or form controls.

11.2: In a form, is the SUBMIT button relevant?

The Submit button is usually the final validation step in filling out an online form. If this button is not clearly identified, it is difficult for the user to interact with the Web site. For example, if the SUBMIT button is an image, comment the image with explicit text, such as "Submit Form."

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_02, 1.1_HTML_03

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: See 11.1 in this section.

11.3: Is the data entry control in online forms accessible?

If an online form is validated by using a script on the client, for example, provide an alternative to the script so that the user, using any type of browser, can submit the information from the form to the server. Additionally, if an HTML data entry control is used, mandatory data entry fields should be specified explicitly (for example, mandatory fields are preceded by an asterisk). Developers should not use colors to indicate mandatory fields or specify the mandatory fields only at the bottom of the page.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.3
WCAG 1.0 checkpoint(s) [Priority 2]: 6.5
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.2_HTML_02, 6.2_HTML_03, 6.3_HTML_01, 6.3_HTML_02, 6.5_HTML_03

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

Most .NET ASP controls assume that the Web client supports scripts (the "postback" concept in ASP.NET). For example, when a user clicks on a .NET ASP button that has an associated OnClick event, it causes the browser event handler to execute a script function that, in turn, sends the HTML form.

In such cases, .NET ASP controls that require this function should not be used; only basic HTML controls and elements should be used. For this reason, data must be entered by using query strings or by submitting the online form without scripts, using HTML Submit buttons.

<object> and <applet> tags must also provide alternate text for the element content.

3.1.12 Browsing aids

12.1: Is the main navigation menu on the Web site located in the same place on all pages?

When users browse through a site, they find it easier to locate a menu or navigation toolbar if it is always in the same place on all pages of the Web site. If browsing mechanisms differ from one page to the next, it creates difficulties for users and might cause them to abandon the Web site.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.4
WCAG 1.0 checkpoint(s) [Priority 3]: 14.3

UWEM 1.0 Test(s): 13.4_HTML_01

Type of criterion: design

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

Using the MCMS 2002 API to create browsing controls makes it possible to provide more consistency throughout a Web site.

Content authors have no control over this feature.

12.2: If keyboard shortcuts are defined for the site, are they active on the page?

If keyboard shortcuts are set up for a page, these shortcuts should be activated and homogeneous throughout the pages of the Web site.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.4_HTML_01

Type of criterion: design / development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

Content authors cannot create or manipulate tab indexes by using standard content editing controls (placeholders). If a content author creates hyperlinks in the Web content, the hyperlinks should be displayed sequentially as a user chooses the Tab button to move from one link to the next.

Accepted practice is to create separate blocks in the template and provide tab index values within each block by assigning a unique tab index range to each, that is, 100-200, 200-300, and so forth. This approach enables consistent and orderly browsing. In this way, the user can browse through all the available content.

3.1.13 Accessible content

13.1: Can the user control screen refresh?

When screen content is automatically refreshed, the cursor in screen readers (such as JAWS) is automatically reset to the top of the page, making it difficult for the user to read the entire page.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 7.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 7.4_HTML_01, 7.4_HTML_02, 7.4_external_01

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so.

Content authors have no control over these features.

13.2: If the user is automatically redirected, is it without using a script?

It is best to use automatic redirection solutions from the server side. In this way, redirection does not depend on the client workstation.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 7.5
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 7.5_HTML_01, 7.5_HTML_02, 7.5_external_01

Type of criterion: Design / development (development rules)

Can be accomplished by using templates: Yes
Can be accomplished by using author content: N/A

Comments: -

13.3: Is a Web site visitor alerted when new windows appear?

The term "new windows" can refer to two types of windows: new browser windows or pop-up windows. New browser windows are opened by using an HTML command, whereas pop-up windows are opened by using a script on the client side. In some cases, people who use browsers that do not support scripting on the client side or who voluntarily disable scripts in their browser do not have access to the information contained in pop-up windows. If a pop-up window opens automatically whenever a page is loaded, it makes the site difficult to read, especially for people who use assistive technologies for browsing. It is not recommended that new windows be opened, especially if the user is not notified.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 10.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 10.1_HTML_01, 10.1_HTML_02, 10.1_HTML_03

Type of criterion: design

Can be accomplished by using templates: Yes
Can be accomplished by using author content: No

Comments: Template developers have full control over the HTML used in the template. It is possible to create templates that satisfy this criterion. It is the responsibility of developers to ensure that the HTML code they produce does so. .

Content authors can create hyperlinks for images by using the `ImagePlaceholder` control. This control can be used to specify whether the link is opened in a new window. If so, the user is not notified.

One possible workaround is to prohibit, at the design rule level, the creation of image hyperlinks that open new windows.

13.4: Is there an alternative to scripts for opening new windows?

If no alternative to pop-ups is provided, some users cannot access pop-ups (for example, users whose browsers do not support scripting or who voluntarily disable scripting in their browser).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 10.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.2_HTML_03, 6.3_HTML_02

Type of criterion: design / development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: No

Comments: See 13.3 in this section.

13.5: Is additional information available to describe files that can be downloaded from the Web site?

When a user downloads files, the following information should be provided: file format, file size, link to a conversion utility if required, link to software needed to read the file format if required, and e-mail address for requesting a file in an alternate format. Files that are available for download on the Web site are not always in an accessible format. Therefore, it is important to provide the necessary means to access the downloadable files. If a file is in PDF format, for example, the user should be provided with the opportunity to download Acrobat Reader software (or a URL for an online conversion utility).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 11.3

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: -

13.6: Does the specific presentation or layout of information interfere with the ability to access its content?

Care must be taken to ensure that specific layout or presentation elements (blinking text, scrolling text, and so forth) do not interfere with the user's ability to access the information: for example, fast blinking text (difficult for most users to read and even more difficult for people with visual disabilities) or fast scrolling information (difficult for people with visual disabilities or for people who have reading difficulties). If this type of presentation or layout is generated by an applet, a FLASH animation, or a GIF file, an alternative should be made available.

WCAG 1.0 checkpoint(s) [Priority 1]: 7.1
WCAG 1.0 checkpoint(s) [Priority 2]: 7.2, 7.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 7.1_HTML_01, 7.1_HTML_02, 7.1_HTML_03, 7.1_CSS_01, 7.1_external_01, 7.1_external_02 , 7.3_HTML_01, 7.3_HTML_02, 7.3_external_01, 7.3_external_02

Type of criterion: design

Can be accomplished by using templates: Yes
Can be accomplished by using author content: Yes

Comments: -

3.1.14 Summary

As can be seen by the preceding discussion, MCMS 2002 supports WCAG 1.0 checkpoints in all three priority levels and can be used to create Web sites that are accessible to people with a wide variety of disabilities.

This said, throughout the lifecycle of a MCMS 2002 project, the design considerations identified in section 2.1 “Design checklist for accessible Web content” as well as the ones described in section 2.2 “Coding checklist for accessible Web content” should be observed.

3.2 Assessment and validation of a MCMS 2002 Web site for WCAG 1.0 checkpoints

The information provided under section 2.3 “Assessment and validation of accessible Web Content” also applies in for MCMS 2002-based Web sites.

In addition to the tools mentioned earlier in this document, third-party solutions that enhance the functionality of MCMS 2002 are available. This section discusses some of these solutions, which are also described in the article Content Management Server: Tool and Technology Partners.

HiSoftware provides a complete Web site quality, accessibility, testing, and remediation suite that works with MCMS 2002 and the Visual Studio 2005 Enterprise (Developer or Architect versions) development environment.

AccVerify 10 provides comprehensive content verification and can identify errors and compliance issues with standards. In addition, the program produces a checklist of criteria for subsequent manual verification.

The AccVerify Integration Pack for Microsoft Content Management Server checks Web files in MCMS 2002 channels for compliance with accessibility standards. A few of the features available through the joint use of MCMS 2002 and AccVerify (with integration pack) are as follows:

- Effective and consistent management of content accessibility throughout all stages of content development and deployment
- Shortens the Web publication cycle by linking Web development processes
- Integrates with the enterprise content management system
- Fully supports WCAG 1.0 priority 1, 2, and 3 checkpoints, as well as section 508 guidelines
- Provides full support for accessibility checking process
- Produces customized reports and supports a wide variety of report formats, including Evaluation And Report Language (*EARL)*

AccRepair 10 provides Web site checking and correction facilities that are based on accessibility policies and standards. This tool fully supports WCAG 1.0 priority 1, 2and 3 checkpoints, as well as section 508 guidelines. It also supports usability testing with the Usability Test Manager.

AccMonitor Compliance Server is designed for Web site accessibility and quality management. AccMonitor Integration Pack II for Microsoft Content Management Server checks the accessibility of Web content by testing compliance with WCAG 1.0 priority 1, 2, and 3 checkpoints and Section 508 guidelines. Accessibility reports can be generated in user-defined formats, such as HTML or XML.

AccMonitor can be configured to automatically supervise MCMS 2002 channels and content submissions at user-defined intervals. This includes, but is not limited to, templates, submissions pending approval, and submissions approved but subject to supervision.

The suite described in this section is included in the Solution for Online Accessibility that was developed jointly by HiSoftware and Microsoft.

4. Accessibility guidelines for using SharePoint 2003 technologies for Web implementations

SharePoint products and technologies facilitate team participation at the enterprise level.

Prior to Microsoft Office system 2007, through the combined features of Windows SharePoint Services (WSS) 2.0 and SharePoint Portal Server (SPS) 2003, users can build their own SharePoint 2003 Web sites and make them accessible throughout the enterprise.

SPS 2003 is a scalable gateway server that enables knowledge sharing between individuals or entire teams. SPS 2003:

- Ensures secure integration of multiple systems by way of federated authentication through single sign-on and enterprise application integration capabilities.
- Provides flexible tools for Web site deployment and management.
- Facilitates teamwork through aggregation, organization, and search capabilities (supporting people, team, and information search).
- Lets users locate the information they require quickly, through content targeting and customization of portal content and layout.

WSS 2.0 (a component of Windows Server 2003 R2) is a Web site creation engine that provides powerful information-sharing and document-based collaborative solutions. SharePoint 2003 Web sites represent a new standard in file management, from uploads to data sharing, enabling users to work collaboratively on documents, tasks, contacts, and events. These sites enable teams and Web site managers to easily manage Web site activity and content.

SPS 2003 makes use of WSS 2.0 by using SharePoint sites to create diverse portal content to effectively disseminate information, targeting individual or enterprise-wide needs. These sites are specific to SPS 2003, but they use technologies such as Web Parts and SharePoint Document Libraries, which are provided by WSS. SPS 2003 also extends the possibilities of WSS 2.0 by providing organizational and management tools for SharePoint Web sites. Use of the Site Registry enables organizations to aggregate, organize, find, and provision SharePoint sites across the enterprise. In addition, SPS 2003 enables users to disseminate information and documents that are stored in their WSS Web sites throughout the enterprise.

For additional information about SharePoint technologies, refer to the links for WSS 2.0 and SPS 2003 respectively at http://www.microsoft.com/windowsserver2003/technologies/sharepoint/default.mspx and http://www.microsoft.com/office/sharepoint/prodinfo/default.mspx.

4.1 Evaluation of SharePoint 2003 Web sites for WCAG 1.0 checkpoints

This white paper evaluates the use of SharePoint 2003 technologies for creating Web sites that comply with the checkpoints defined for the 14 guidelines in the Web Content Accessibility Guidelines (WCAG) 1.0, which were developed by the W3C/WAI and published 5 May 1999 (see section 1.3.1 "WCAG 1.0"). Sometime after WCAG 2.0 is approved and released, a future version of this document will evaluate compliance with the guidelines in the new release.

To assess how SharePoint 2003 technologies support and satisfy WCAG 1.0 guidelines, this document considers whether WSS 2.0 and, consequently, SPS 2003 support the checkpoints, or in some cases, how these products can be adapted to ensure compliance with the checkpoints.

This assessment takes into account the following documents: Windows SharePoint Services 2003 VPAT and SharePoint Portal Server 2003 VPAT.

WSS 2.0 and SPS 2003 are the latest releases available at the date of publication of this white paper. For additional information, refer to the following links:

- Windows SharePoint Services 2.0 with Service Pack 2
- SharePoint Portal Server 2003 Service Pack 2

These two Service Packs provide Microsoft SQL Server 2005 support.

WSS 2.0 with SP2 runs on the new Microsoft .NET Framework 2.0 Common Language Runtime (CLR), which means that it can run using ASP.NET 2.0 technology. However, WSS has not been redesigned to take advantage of any of the new major features of ASP.NET 2.0. At the core level, SP2 does not change the rendering behavior of WSS. In other words, WSS 2.0 cannot use Master Pages or the Web Parts 2.0 Framework. However, these new class libraries are available for customized Web Parts, Web pages, and event handlers.

WSS 2.0 SP2 treats .NET 2.0 Web Parts as standard Web controls and not as Web Parts. One alternative is to write a wrapper Web Part to make the ASP.NET 2.0 Web Part appear to be a WSS 2.0 Web Part.

However, SPS 2003 does not support .NET Framework 2.0, even with SP2, which requires Service Pack 2 of Windows SharePoint Services 2.0. A SPS 2003 SP2 configuration with WSS 2.0 with SP2 is based on .NET Framework 1.1. This does not interfere with a SharePoint site running under ASP.NET 1.1 calling pages and Web services exposed by an ASP.NET 2.0 Web site.

Additional details are provided in the article Service Pack 2 for Windows SharePoint Services and SharePoint Portal Server 2003.

The MOSS 2007 technologies (see chapter 5 Accessibility guidelines for using MOSS 2007 technologies for Web implementations), as a major release of both WSS 2.0 and SPS 2003, are based on, and hence make extensive use of, ASP.NET 2.0 and its multitude of features, including eAccessibility capabilities. See the MSDN article Building ASP.NET 2.0 Web Sites Using Web Standards, which describes these capabilities.

Given the above, the following analysis is based on running WSS 2.0 with .NET Framework 1.1 with Service Pack 1.

Service Pack 1 includes patch 819945 FIX: ASP.NET Controls Do Not Meet Accessibility Guidelines.

4.1.1 Graphics

1.1: Does each graphic have text to display as an alternative to the graphic?

In HTML, use the ALT attribute with IMG and AREA SHAPE elements, and so forth. It is important to provide alternate text for graphics because some browsers and screen readers cannot interpret or display graphics properly.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_01, 1.1_HTML_07, 1.1_HTML_08

Type of criterion: development

Can be accomplished: Yes

Comments: WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion. You can edit the page with Microsoft Office Front Page 2003 or the new Microsoft Office SharePoint Designer 2007 to take advantage of the tight integration between these products and technologies.

Microsoft Office SharePoint Designer 2007 works with the same types of Web sites as Microsoft Office FrontPage 2003 in addition to the specific support of MOSS 2007 technologies (see chapter *5* Accessibility guidelines for using MOSS 2007 technologies for Web implementations). These include Web sites built on Windows SharePoint

Services, disk-based Web sites, and Web sites built on FrontPage Server Extensions. Office SharePoint Designer 2007 can also open and edit Web sites directly with the File Transfer Protocol (FTP) and Web-based Distributed Authoring and Versioning (WebDAV) protocols.

The best approach is to set up a site definition that integrates this criterion as described in the MSDN article Tips for Enhancing the Accessibility of SharePoint Web Sites, in section "Use Alternative Text for Images." This article illustrates how to use these products to create more accessible Web sites while waiting for some already-identified accessibility problems to be resolved in an upcoming release of Office.

HiSoftware offers a free download of new SharePoint templates (in English), which you can use to create Web sites with integrated accessibility.

It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion. However, using site definitions does not enable developers to manage all HTML rendering issues. It is possible to appropriately modify ASPX pages, such as default.aspx, that are integrated in the site definitions. However, this does not influence the standard SharePoint controls that might include the page under consideration. Some controls might need to be rewritten for the rendering to comply with all the relevant criteria. It might be necessary to produce new site definitions that contain new versions of certain SPS 2003 controls (browsing, for example).

1.2: Is the alternate text for each image relevant to the context in which the image is viewed?

A given image might have several meanings. It is the context of the image that determines the exact meaning. For example, a yellow bullet icon in a newspaper Web site should have the ALT attribute set to ALT="" or ALT="*". However, if the same yellow bullet icon is included in a list of graphics that is available for download, it might make more sense to set the attribute to ALT = "yellow diamond" to distinguish it from other graphics available on the page.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_03

Type of criterion: development

Can be accomplished: Yes

Comments: See 1.1 in this section. This criterion is subjective.

1.3: Are graphics that are used only for decorative purposes commented with ALT=""?

Some graphics have no informative value. An example is the use of transparent pixels only for layout purposes. However, all graphical elements must be commented. For this reason, graphics that are used only for decorative or layout purposes should be commented by setting the ALT attribute to null (in HTML, ALT=""). Some browsers do not display the images that use this type of comment.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_02

Type of criterion: development

Can be accomplished: Yes

Comments: See 1.1 in this section.

1.4: Is the alternate text for each image no more than 60 characters long?

You should use a maximum of 60 characters in the ALT attribute because alternate text might bother some users. This is often the case for people who use a screen magnification function because of visual disabilities. It is important to limit the number of characters in the image comments so that such users can minimize the effort required to obtain all the information.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: Yes

Comments: See 1.1 in this section.

1.5: Are all comments that are linked to clickable areas of a MAP image relevant?

Definition: a MAP image is an image that presents clickable areas so that the user can click to select an option. When you use a MAP image in HTML (on the client or the server side), it is important to include alternate text for all clickable areas, as well as for the MAP image itself (that is, the image that contains the USEMAP or ISMAP attribute).

For a MAP image on the client side, provide alternate text by using the ALT attribute in each of the clickable areas (AREA SHAPE).

For a MAP image on the server side, provide redundant text links that repeat the content of each of the clickable areas.

Comment 1: The alternate text provided for the image that contains the USEMAP or ISMAP attribute must not be null.

Comment 2: It is recommended that MAP images be used on the client side rather than on the server side.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1, 1.2, 9.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_03, 1.2_HTML_01, 9.1_HTML_01

Type of criterion: development

Can be accomplished: Yes

Comments: WSS 2.0 and SPS 2003 do not contain and cannot generate standard MAP images.

It remains the responsibility of SharePoint 2003 developers, when creating templates that use this feature, to ensure that their Web pages satisfy this criterion. The solution is to systematically fill in the ALT attribute that is associated with the tag containing the USEMAP attribute (in addition to the images). Each clickable area (AREA SHAPE attribute) must include an appropriate ALT attribute.

1.6: Is the alternate content for each text image at least the equivalent of the text appearing in the image?

It is not recommended that you present text in the form of an image. However, if there is an unavoidable graphical constraint (for example, non-equivalent CSS rendering), the alternate text for a graphical element containing text should, at minimum, present the exact content of the text itself. The alternate text can provide additional information, but it must at least provide the exact content.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_03

Type of criterion: development

Can be accomplished: Yes

Comments: See 1.1 in this section.

1.7: Do all images that require a detailed description provide comment text?

When a graphical element requires a long description in HTML (for example, a complex image or alternate text longer than 60 characters) a text comment must be provided. The comment can be a link to a descriptive file (for example, a D-link or a clickable graphic), a caption that appears directly on the page, or a LONGDESC attribute.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_05

Type of criterion: development

Can be accomplished: Yes

Comments: WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion. The LONGDESC attribute can be added by using FrontPage 2003 to edit the page. The best approach is to create a site definition that integrates this criterion.

The WCAG directive specifies that the LONGDESC attribute should be used, but currently most browsers do not support it. Internet Explorer 6.0 exposes this attribute from a programming point of view in its model object. Therefore, assistive technologies can make use of it. Longdesc Linker for Internet Explorer 6 is a free downloadable ActiveX component that makes it possible to add a Long Description element to the context menu used by Internet Explorer for images. This element is enabled if the selected image has a LONGDESC link. If the user selects the link (by holding down the SHIFT key while selecting), Internet Explorer follows it.

1.8: If a detailed description is provided for an image, is the content relevant?

The content should provide additional information to the description defined in the ALT attribute for the image.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_04

Type of criterion: development

Can be accomplished: Yes

Comments: See 1.7 in this section. This criterion is subjective.

1.9: Does the text used in the ALT attribute for each image provide the function of the link?

The image description is not always in keeping with the function of the link. For example, if an arrow is used to indicate a link, use ALT="Next" instead of ALT="Right Arrow". Avoid using ALT="", which provides no information about the function of the link.

WCAG 1.0 checkpoint(s) [Priority 1]: 13.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_02

Type of criterion: development

Can be accomplished: Yes

Comments: WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion. The solution is to place an ALT attribute between the <IMG> tags embedded in an <A> tag, making sure that the text is relevant.

It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

4.1.2 Frames

2.1: Does each frame have a NAME attribute?

In HTML, each frame must be named by using the NAME attribute. Browsing through frame-based Web pages can be complex, especially for visually impaired people who become familiar with site content frame by frame (whereas sighted people have an overview of the entire site). Comment: The content of the NAME attribute should not include any spaces.

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: WSS 2.0 and SPS 2003 are not frame-based by default. However, it is possible to create pages that have frames. It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

2.2: Are the names assigned to frames relevant?

The names assigned to frames should be relevant. Avoid naming frames with directional data, such as "Left," "Right," or "Middle." It is better to name frames with explicit names that refer to the function provided by the frame, such as "Menu," "Navigation," or "Content."

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.3: Is there a NOFRAME tag?

In HTML, the NOFRAME tag is used to provide the user with a summary of the main links on the Web site. This tag facilitates browsing for browsers that do not handle frames.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 6.5
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.5_HTML_02

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.4: Is the content of the NOFRAME tag relevant?

If a NOFRAME tag is present, the information in the tag must be relevant. In many cases, the NOFRAME tag is present but provides information that is either inappropriate or incomplete. This tag is intended to provide the main links existing on the Web site.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 6.5
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.5_HTML_02

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.5: Does each frame have a TITLE attribute?

In HTML, the TITLE attribute is used to describe, in a few words, the purpose of the frame. This attribute is complementary to the NAME attribute.

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 12.1_HTML_01

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.6: Is the content of the TITLE attribute relevant?

Some assistive technologies can recognize this attribute. The content of this attribute helps the user navigate from frame to frame independently and quickly determine the function of each frame.

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 12.1_HTML_02

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.7: Does each page have a maximum of three frames?

It is important to have no more than three frames on any one page. If more than three frames are used, the page becomes difficult to read, especially for users of assistive technologies.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.8: When frames are used, is scrolling automatic?

In HTML, a frame can no longer be scaled if scrolling is set to "NO." The scrolling function must be enabled. For users of magnification software, which might implement user-specific style sheets with larger fonts, and so forth, in-frame scrolling must be enabled to allow all content to be accessed.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 14.3

UWEM 1.0 Test(s): -

Type of criterion: design / development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

4.1.3 Colors

3.1: Is information provided by color still readable when colors are disabled?

Web site designers must be particularly careful about information provided by colors. This information can be lost to colorblind users or users who do not have adequate visual acuity to distinguish between certain colors.

WCAG 1.0 checkpoint(s) [Priority 1]: 2.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 2.1_HTML_01, 2.1_HTML_02, 2.1_HTML_03, 2.1_CSS_01

Type of criterion: design

Can be accomplished: Yes

Comments: WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion. It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

3.2: Is there enough contrast between colors to be distinguishable by users who have impaired color vision?

Some color combinations should be avoided, such as red with blue, yellow with violet, yellow with green, and so forth. When there is not enough contrast between colors on a page, the information provided by contrast might not be accessible to users with poor or impaired color vision.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 2.2
WCAG 1.0 checkpoint(s) [Priority 3]: 2.2

UWEM 1.0 Test(s): 2.2_HTML_01, 2.2_CSS_01

Type of criterion: design

Can be accomplished: Yes

Comments: See 3.1 in this section. However, the best approach is to set up a site definition that integrates this criterion as described in the MSDN article Tips for Enhancing the Accessibility of SharePoint Web Sites in section "Create and Apply High-Contrast Themes to Web Sites."

It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

To assess this criterion, a color contrast evaluation tool, such as the free downloadable software by Vision Australia, can be used to verify foreground and background color combinations to determine whether the color contrast enables proper viewing. The identification of proper color visibility is based on the algorithms provided by the W3C.

4.1.4 Multimedia

4.1: Can the information that is conveyed by multimedia be provided another way?

Whenever multimedia is used to provide information, it is important to provide an alternative for those who do not have access to multimedia. For example, if a multimedia file is made available on a Web site, it is important to provide an accessible text-based alternative in a standard format, such as RTF, TXT, or HTML.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.3, 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.3_external_01, 1.3_external_02, 1.1_HTML_01, 1.1_HTML_02, 1.1_HTML_03, 1.1_HTML_04, 1.1_HTML_05, 1.1_HTML_06, 1.1_HTML_09, 1.1_HTML_10, 1.1_external_01, 1.1_external_02, 6.3_HTML_01

Type of criterion: design / development

Can be accomplished: Yes

Comments: WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion. It remains the responsibility of SharePoint developers to ensure that their Web pages satisfy this criterion.

4.2: Is the Multimedia content synchronized with the alternate support?

If a video with sound is used on a site, as in the case of an interview between several people, it is important to synchronize the alternate support with the picture and the sound in the video. The W3C recommends the use of the Synchronized Multimedia Integration Language (SMIL) programming language.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.4, 1.3
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.4_external_01

Type of criterion: design / development

Can be accomplished: Yes

Comments: This criterion requires the use a tool such as Hi-Caption from HiSoftware, which is a Synchronized Multimedia Integration Language (SMIL) presentation editor, or Microsoft Synchronized Accessible Media Interchange (SAMI). Please refer to the technical articles Understanding SAMI 1.0 and Adding Closed Captions to Digital Media for addition information about how to add subtitle tracks to the Windows Media format.

You might want to refer to HiSoftware Hi-Caption Studio - Creating Accessible Demos as an illustration.

It remains the responsibility of SharePoint 2003 developers to ensure that their multimedia content satisfies this criterion.

4.1.5 Tables

5.1: Is the SUMMARY attribute present and relevant?

If the SUMMARY attribute is used correctly, it provides additional information to disabled users, such as the visually impaired, about the tables in a Web page. If a Web page contains menu bars that are created by using an HTML table, setting SUMMARY="Menu bar" provides the user with an additional information about the table function.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 5.5

UWEM 1.0 Test(s): -

Type of criterion: development

***Can be accomplished:* No**

Comments: Tables are often used in WSS 2.0 and SPS 2003 especially for laying out information on a page.

5.2: In a data table, does the CAPTION tag provide the title of the table?

Rather than using formatted text to provide the title of a table, it is better to use the CAPTION tag, which is the tag dedicated to displaying table titles.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 5.5

UWEM 1.0 Test(s): 12.3_HTML_07, 12.3_HTML_08, 5.4_HTML_04

Type of criterion: development

***Can be accomplished:* Yes**

Comments: WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion. It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

This criterion might not be met for tables that are generated by WSS 2.0 and SPS 2003 lists and document libraries.

5.3: In data tables, are the column headers appropriate?

In HTML data tables, the TH tag specifies column headers, enabling users of assistive technologies to use text-to-voice software to determine that they are column headers.

WCAG 1.0 checkpoint(s) [Priority 1]: 5.1
WCAG 1.0 checkpoint(s) [Priority 2]: 5.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 5.1_HTML_01, 5.4_HTML_01, 5.4_HTML_01

Type of criterion: development

***Can be accomplished:* Yes**

Comments: In some cases, the TH tag is used for layout purposes rather than for identifying the content of the column.

This criterion might not be met for tables that are generated by WSS 2.0 and SPS 2003 lists and document libraries.

5.4: In a data table, does a HEADERS attribute link to each of the data cells in the table?

When correctly configured in a data table, the HEADERS and ID attributes make it possible to link each data cell to the appropriate column header. This approach enables users that have only a linear view of the table to know which header is linked to the current data cell.

WCAG 1.0 checkpoint(s) [Priority 1]: 5.2
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 5.2_HTML_01, 5.2_HTML_02, 5.4_HTML_03

Type of criterion: development

***Can be accomplished:* Yes**

Comments: WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion. It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

This criterion might not be met for tables that are generated by WSS 2.0 and SPS 2003 lists and document libraries.

5.5: Is the content in formatted tables in correct sequence?

In formatted tables, the information must be ordered so that it is displayed in an equivalent sequence when the information is read in a linear fashion. Care must be taken when organizing information in a formatted table. HDL development tools that are currently available have a tendency to nest tables within one another. The result is that occasionally differences occur between the visual appearance of information in a graphical browser and the text equivalent, because the latter is based on the structure of the HTML code (as in LYNX-type text browsers, for example).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 5.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 5.3_HTML_01

Type of criterion: design

Can be accomplished:* *No

Comments: This criterion might not be met for tables generated by WSS 2.0 and SPS 2003 lists and document libraries.

4.1.6 Links

6.1: Are Link titles no more than 80 characters long?

Use a maximum of 80 characters in hypertext link titles (accents and spaces count for one character as well).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: design / development

Can be accomplished:* *Yes

Comments: The standard Web Part for link handling enables authors to publish links with titles containing a maximum of 256 characters. It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion with a title of no more than 80 characters.

6.2: Are links explicit enough?

Links must be understandable independent of the page context. Some assistive technologies are able to display a list of available links. These links must be understandable without the context of the Web page to enable rapid scanning. Avoid links such as "click here."

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_02

Type of criterion: design / development

Can be accomplished:* *Yes

Comments: This criterion is subjective.

6.3: Is the TITLE attribute used, if required, and is it no more than 80 characters long?

In HTML, the TITLE attribute is necessary if additional information is required to understand a link. For example, if a link is used to download a file, the TITLE attribute can provide the file name, file size in bytes, file format, and destination ("open in new window").

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: Yes

Comments: -

6.4: Does the TITLE attribute provide more information about the link than the link title itself?

The only information that some users have about a link is the title of the link or the content of the TITLE attribute. Therefore, the content of the TITLE attribute should, at the very least, repeat the information contained in the link title. For example, a TITLE attribute that simply indicates "this link opens in a new window" is not adequate, because the user does not know the link destination. However, "Read article on tourism in China (new window)" provides all the essential information required about the link.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_01, 13.1_HTML_02

Type of criterion: - development

Can be accomplished: Yes (see 6.3)

Comments: See 6.3 in this section.

6.5: Do all identical link titles lead to the same target?

Avoid using link titles such as "Read article" multiple times on a page. It is important to use link titles that are not identical to help people who wish to move from link to link to read the page more quickly. It is also important to use short, explicit links with unambiguous destinations.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_01

Type of criterion: design

Can be accomplished: Yes

Comments: -

4.1.7 Scripts

7.1: If a script requires alternate text to make it accessible, is the information provided by the alternate text equivalent to the information provided by the script?

The information about a script should remain available even if the script is disabled. For example, if a JavaScript script is used to validate an online form, the alternative might be a server-side data entry control. Scripts can be disruptive to users of assistive technologies that do not support scripts or that support them only in a limited fashion. In addition, some users voluntarily disable script execution in standard browsers. For this reason, an alternative must be available to retain the information generated by the scripts.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.2, 6.3, 8.1

WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.2_HTML_02, 6.2_HTML_03, 6.3_HTML_02

Type of criterion: design / development

Can be accomplished: No

Comments: WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion. It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

However, note that some operations do not work when scripts are disabled and no alternate page is provided. JavaScript scripts are used in menus, in rendering certain pages, as well as in validating online forms. In the latter case, the .NET code determines where controls are applied, depending upon the browser characteristics.

Because the current versions of WSS and SPS are largely based on the use of JavaScript, this criterion might not be satisfied.

7.2: Can actions be performed even if the peripheral for which they were designed is disabled?

Some functions of a Web site, such as those generated by JavaScript scripts, can be activated only by a mouse click. However, some users cannot use a mouse to browse through a Web site. It must be possible to browse through a Web site by using just a keyboard.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 6.4, 9.2, 9.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 9.2_external_01, 6.3_HTML_02, 6.4_HTML_01, 6.4_HTML_02, 6.4_external_01, 8.1_HTML_01, 8.1_external_01, 9.3_HTML_01

Type of criterion: design / development

Can be accomplished: No

Comments: The use of a mouse is required for some actions in SharePoint 2003 (context menu in list elements and document libraries).

4.1.8 Mandatory elements

8.1: Is the DOCTYPE tag present at the beginning of the page source code?

International WAI guidelines suggest using an appropriate and valid Document Type Definition (DTD) to indicate the type of electronic document that the browser must load (frame-based, HTML, XML, and so forth). In HTML, the DOCTYPE tag is required to define the document type.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 3.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 3.2_HTML_01

Type of criterion: development

Can be accomplished: Yes

Comments: The DOCTYPE tag is not available by default.

WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion. It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

Note that the current versions of WSS and SPS do not strictly comply with the formal syntax requirements specified in HTML 4.01 or XHTML 1.0 as required by checkpoint 3.2.

8.2: Is the LANG attribute present at the beginning of the page source code to clearly identify the language used?

It is important to specify the language of a document to enable assistive technologies to adapt content to the language. In HTML, the LANG attribute in the HTML tag specifies the language.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 4.3

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: Yes

Comments: The LANG attribute is not used by default.

WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion. It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

8.3: Is there a TITLE tag in the page header?

In HTML, the title of a page is coded by including a TITLE tag in the document header. Voice browsers list the title of the page first so that a user can get positioned on the Web site. The title is a constant source of support to those having an attention deficit disorder. The tag also makes it easier for search engines to reference the Web site.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.2_HTML_01

Type of criterion: development (development rules)

Can be accomplished: Yes

Comments: -

8.4: Is the content of the TITLE tag explicit?

The TITLE tag provides the title of the page that was just loaded. It is important to provide an explicit title to help users get oriented during browsing.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.2_HTML_01

Type of criterion: design

Can be accomplished: Yes

Comments: See 8.3 in this section. This criterion is subjective.

8.5: Is the content of the TITLE tag different from one page to the next?

Using an identical title on every page of a Web site can be confusing to users. The title of the page is the first information that appears on the screen. Assigning the same title to multiple documents is misleading.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: design

Can be accomplished:* *Yes

Comments: See 8.3 in this section. This criterion is subjective.

8.6: Are language changes on a page indicated?

Whenever a language change occurs on a page, it should be indicated to the user. If text-to-voice software is used, a change in language causes the software detection language to change as specified by the LANG tags.

WCAG 1.0 checkpoint(s) [Priority 1]: 4.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 4.1_HTML_01, 4.1_HTML_02, 4.1_HTML_03, 4.1_HTML_04, 4.1_CSS_01, 4.1_CSS_02

Type of criterion: development

Can be accomplished:* *No

Comments: WSS 2.0 and SPS 2003 are not multilingual (the language depends upon the installed release). If a language change occurs, it probably originates from a WSS 2.0 or SPS 2003 control.

4.1.9 Information structuring

9.1: Is information structured consistently for the general context of the site?

The structure of information on a page depends on the type of site (news, gaming, scientific, artistic, and so forth). In addition, the individual elements that provide information on the page (such as images and titles) might not be a problem in and of themselves, but the arrangement of these elements in relation to one another might not be consistent. For example, on a news site, no text should be displayed before the title.

WCAG 1.0 checkpoint(s) [Priority 1]: 14.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 12.3_HTML_10, 12.3_HTML_15, 14.1_HTML_01

Type of criterion: design

Can be accomplished:* *Yes

Comments: This criterion is subjective.

9.2: Is the Web page presented in a consistent fashion?

Use of styles that are available in HTML, for example H1 and H2, can provide structure to a Web page. Such tags are extremely important in structuring a Web page. For example, the H1 tag indicates a title and the H2 tag indicates subheadings. These tags must be used for their intended purpose, that is, to provide structure, and not for their graphic capabilities. Additionally, blocks of information should be separated from each other.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 3.5
WCAG 1.0 checkpoint(s) [Priority 3]: 13.8

UWEM 1.0 Test(s): 3.5_HTML_01, 3.5_HTML_02, 3.5_HTML_03, 3.5_HTML_04, 3.5_HTML_05

Type of criterion: design

Can be accomplished:* *No

Comments: SharePoint 2003 technologies do not currently use headers such as H1 and H2 to structure Web pages.

4.1.10 Information layout

10.1: Is page content separated from content introduction?

If a page is coded in HTML, it should not contain formatting information (for example, BGCOLOR, FONT, ALIGN, B, I), because all this information can be included in style sheets.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 3.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 3.3_HTML_01, 3.3_HTML_02

Type of criterion: development (development rules)

Can be accomplished: Yes

***Comments:* -**

10.2: If style sheets are disabled, is the information still accessible?

To separate content from layout on an HTML page, use style sheets (CSS) to specify the layout. However, people who do not wish to read the page by using the author-selected style sheet should obtain the same information when they disable the page style sheet and use their own customized settings instead.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 4.1_CSS_02, 6.1_HTML_01, 6.1_HTML_02

Type of criterion: design / development

Can be accomplished: Yes

Comments: It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

10.3: If style sheets are disabled, is the order in which information appears the same as initially defined?

Style sheets can be used to construct HTML pages. If style sheets are disabled by the user, the logical structure of the page should stay the same as initially defined. For example, the order of appearance of the divisions (DIV) should stay the same when viewed by using a graphical browser or by using a text-mode browser.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 9.4

UWEM 1.0 Test(s): 3.6_HTML_01, 6.1_HTML_01, 6.1_HTML_02

Type of criterion: design / development

Can be accomplished: Yes

Comments: It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

4.1.11 Forms

11.1: Are the LABEL tag and its corresponding attributes (ID, FOR) present?

In HTML pages, the LABEL tag is used with the ID and FOR attributes to link data entry fields directly to the associated text. This approach enables people who use assistive technologies to unambiguously determine which data entry field corresponds to the linked text.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 10.2, 12.4

WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 10.2_HTML_01, 12.4_HTML_01, 12.4_HTML_02

Type of criterion: development

Can be accomplished: Yes

Comments: The LABEL tag is not used by default.

WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion. It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

11.2: In a form, is the SUBMIT button relevant?

The OK button or Submit button is usually the final validation step in filling out an online form. If this button is not clearly identified, it is difficult for the user to interact with the Web site. For example, if the SUBMIT button is an image, comment the image with explicit text, such as "Submit Form."

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_02, 1.1_HTML_03

Type of criterion: development

Can be accomplished: Yes

Comments: WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion.

However, the best approach is to set up a site definition that integrates this criterion as described in the MSDN article Tips for Enhancing the Accessibility of SharePoint Web Sites in section "Add Submit Buttons to the Bottom of Forms."

Note that HiSoftware offers free downloadable SharePoint templates (in English) that can be used to create Web sites with integrated accessibility.

It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

11.3: Is the data entry control in online forms accessible?

If an online form is validated by using a script on the client, for example, provide an alternative to the script so that the user, using any type of browser, can submit the information from the form to the server. Additionally, if an HTML data entry control is used, mandatory data entry fields should be specified explicitly (for example, mandatory fields are preceded by an asterisk). Developers should not use colors to indicate mandatory fields or specify the mandatory fields only at the bottom of the page.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.3
WCAG 1.0 checkpoint(s) [Priority 2]: 6.5
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.2_HTML_02, 6.2_HTML_03, 6.3_HTML_01, 6.3_HTML_02, 6.5_HTML_03

Type of criterion: development

Can be accomplished: Yes

Comments:

4.1.12 Browsing aids

12.1: Is the main navigation menu on the Web site located in the same place on all pages?

When users browse through a site, they find it easier to locate a menu or navigation toolbar if it is always in the same place on all pages of the Web site. If browsing mechanisms differ from one page to the next, it creates difficulties for users and might cause them to abandon the Web site.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.4
WCAG 1.0 checkpoint(s) [Priority 3]: 14.3

UWEM 1.0 Test(s): 13.4_HTML_01

Type of criterion: design

Can be accomplished: Yes

Comments: WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion.

The best approach is to set up a site definition that integrates this criterion as described in the MSDN article Tips for Enhancing the Accessibility of SharePoint Web Sites in section "Add Skip-to-Main Content Links".

Note that HiSoftware offers free downloadable SharePoint templates (in English) that can be used to create Web sites with integrated accessibility.

It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

12.2: If keyboard shortcuts are defined for the site, are they active on the page?

If keyboard shortcuts are set up for a page, these shortcuts should be activated and homogeneous throughout the pages of the Web site.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.4_HTML_01

Type of criterion: design / development

Can be accomplished: Yes

Comments: Shortcuts are not available on screens that do not provide menus.

4.1.13 Accessible content

13.1: Can the user control screen refresh?

When screen content is automatically refreshed, the cursor in screen readers (such as JAWS) is automatically reset to the top of the page, making it difficult for the user to read the entire page.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 7.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 7.4_HTML_01, 7.4_HTML_02, 7.4_external_01

Type of criterion: development

Can be accomplished: Yes

Comments:

13.2: If the user is automatically redirected, is it without using a script?

It is best to use automatic redirection solutions from the server side. In this way, redirection does not depend on the client workstation.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 7.5

WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 7.5_HTML_01, 7.5_HTML_02, 7.5_external_01

Type of criterion: Design / development (development rules)

Can be accomplished:* *N/A

***Comments:* -**

13.3: Is a Web site visitor alerted when new windows appear?

The term "new windows" can refer to two types of windows: new browser windows or pop-up windows. New browser windows are opened by using an HTML command, whereas pop-up windows are opened by using a script on the client side. In some cases, people who use browsers that do not support scripting on the client side or who voluntarily disable scripts in their browser do not have access to the information contained in pop-up windows. If a pop-up window opens automatically whenever a page is loaded, it makes the site difficult to read, especially for people who use assistive technologies for browsing. It is not recommended that new windows be opened, especially if the user is not notified.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 10.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 10.1_HTML_01, 10.1_HTML_02, 10.1_HTML_03

Type of criterion: design

Can be accomplished:* *No

***Comments:* -**

13.4: Is there an alternative to scripts for opening new windows?

If no alternative to pop-ups is provided, some users cannot access pop-ups (for example, users whose browsers do not support scripting or who voluntarily disable scripting in their browser).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 10.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.2_HTML_03, 6.3_HTML_02

Type of criterion: design / development

Can be accomplished:* *No (see 13.3)

Comments: See 13.3 in this section.

13.5: Is additional information available to describe files that can be downloaded from the Web site?

When a user downloads files, the following information should be provided: file format, file size, link to a conversion utility if required, link to software needed to read the file format if required, and e-mail address for requesting a file in an alternate format. Files that are available for download on the Web site are not always in an accessible format. Therefore, it is important to provide the necessary means to access the downloadable files. If a file is in PDF format, for example, the user should be provided with the opportunity to download Acrobat Reader software (or a URL for an online conversion utility).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 11.3

UWEM 1.0 Test(s):

Type of criterion: development

Can be accomplished:* *Yes

Comments: File size is not available.

13.6: Does the specific presentation or layout of information interfere with the ability to access its content?

Care must be taken to ensure that specific layout or presentation elements (blinking text, scrolling text, and so forth) do not interfere with the user's ability to access the information: for example, fast blinking text (difficult for most users to read and even more difficult for people with visual disabilities) or fast scrolling information (difficult for people with visual disabilities or for people who have reading difficulties). If this type of presentation or layout is generated by an applet, a FLASH animation, or a GIF file, an alternative should be made available.

WCAG 1.0 checkpoint(s) [Priority 1]: 7.1
WCAG 1.0 checkpoint(s) [Priority 2]: 7.2, 7.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 7.1_HTML_01, 7.1_HTML_02, 7.1_HTML_03, 7.1_CSS_01, 7.1_external_01, 7.1_external_02, 7.3_HTML_01, 7.3_HTML_02, 7.3_external_01, 7.3_external_02

Type of criterion: design

Can be accomplished: Yes

Comments: WSS 2.0 and SPS 2003 can be used to create Web pages that satisfy this criterion.

It remains the responsibility of SharePoint 2003 developers to ensure that their Web pages satisfy this criterion.

4.1.14 Summary

As can be seen by the preceding discussion, SharePoint 2003 technologies are capable of producing Web sites that comply with WCAG 1.0 checkpoints in all three priority levels. The only issues that arise are related to alternatives for client-side script usage.

This said, throughout the lifecycle of a SharePoint 2003 technologies-based project, the design considerations identified in section 2.1 “Design checklist for accessible Web content” as well as the ones described in section 2.2 “Coding checklist for accessible Web content” should be observed.

4.2 Assessment and validation of a SharePoint 2003 Web site for WCAG 1.0 checkpoints

The information provided under section 2.3 "Assessment and validation of accessible Web Content" also applies in for SharePoint 2003-based Web sites.

In addition to the tools mentioned earlier in this document, this section discusses specific solutions that integrate with SharePoint 2003 technologies.

HiSoftware provides a suite of software tools that work with SharePoint 2003 technologies and the Visual Studio 2005 development environment.

In addition to the solutions described in section 3.2 “Assessment and validation of a MCMS 2002 Web site for WCAG 1.0 checkpoints”, the AccMonitor Integration for Microsoft SharePoint Portal Server for AccMonitor Compliance Server 2005 provides a test management and remediation solution for accessibility and content quality checking for SPS 2003 workspaces.

The HiSoftware white paper Harnessing the power of HiSoftware AccMonitor Compliance Server 2005 for Microsoft SharePoint Portal Server 2003 describes the integration features provided by AccMonitor Compliance Server 2005, as well as how to integrate HiSoftware-SharePoint Tester into a SPS 2003 instance.

For additional information, refer to the article AccMonitor Compliance Server 2005 Advances Accessibility and Content Quality for SharePoint Portal Server v2.0 Sites.

5. Accessibility guidelines for using MOSS 2007 technologies for Web implementations

Microsoft Office SharePoint Server (MOSS) 2007 provides portal and search features that represent the next generation of SharePoint Portal Server (SPS) 2003 (see previous chapter 4 "Accessibility guidelines for using SharePoint 2003 technologies for Web implementations").

However, extra dimensions have been given to MOSS 2007 in Web content management (WCM), document and records management, business process integration and forms, and business intelligence.

As a result, MOSS 2007 constitutes an integrated suite of server capabilities that can help improve organizational effectiveness by providing comprehensive content management and enterprise search, accelerating shared business processes, and facilitating information-sharing across boundaries for better business insight. Office SharePoint Server 2007 supports all intranet, extranet, and Web applications across an enterprise within one integrated platform, instead of relying on separate fragmented systems.

Microsoft Office Forms Server 2007, a subset of MOSS 2007, offers a solution that aims at facilitating the form templates publishing and management from a central local.

MOSS 2007 is built upon Microsoft Windows SharePoint Services (WSS) 3.0[11], which includes into a common architecture many features previously provided by MCMS 2002 (see previous chapter 3 Accessibility guidelines for using Microsoft Content Management Server 2002 for Web implementations), such as web content management, and related workflows. Generally speaking, many tasks such as navigation, summary links, and scheduled deployment that an MCMS 2002 developer would perform can now be performed by users or IT professionals.

Many of the new features in MOSS 2007 are made possible by new WSS 3.0 innovations (and ASP.NET 2.0) such as Master page integration, content types, versioning, and workflows. MOSS 2007 portal sites are created using WSS features and site definitions along with familiar ASP.NET building blocks such as Master Pages and Web Parts.

This means that MOSS 2007 solutions can easily be extended with ASP.NET components such as server-side controls and custom Web Parts as well as custom components that utilize WSS features like custom list definitions, document libraries, event handlers, and workflows.

The white paper Gathering MOSS - New Dev-Centric Features In Office SharePoint Server Keep Your Apps Rolling provided a broad survey of the most significant services and features provided by MOSS 2007 technologies.

For additional information about MOSS 2007 technologies, refer to the links for WSS 3.0 and MOSS 2007 respectively at http://www.microsoft.com/sharepoint/default.mspx.

Please also note the availability of the Microsoft Office SharePoint Server 2007 Software Development Kit (SDK) and the Microsoft Windows SharePoint Services 3.0 SDK that include documentation (overview information, step-by-step, and how-to procedures for programming) for the technologies that MOSS 2007 provides for developers, code several examples within the topics, as well as some tips and best practices for using the development platform of MOSS 2007 technologies.

[11] A comparison of Windows SharePoint Services v3.0 vs. Microsoft Office SharePoint Server 2007 is available on the Marwan Tarek's Blog. It offers a list of things that MOSS 2007 provides that WSS 3.0 does not.

5.1 Evaluation of MOSS 2007 Web sites for WCAG 1.0 checkpoints

This white paper evaluates the use of MOSS 2007 technologies for creating Web sites that comply with the checkpoints defined for the 14 guidelines in the Web Content Accessibility Guidelines (WCAG) 1.0, which were developed by the W3C/WAI and published 5 May 1999 (see section 1.3.1 "WCAG 1.0"). Sometime after WCAG 2.0 is approved and released, a future version of this document will evaluate compliance with the guidelines in the new release.

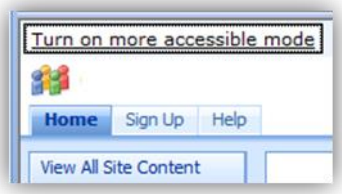
To assess how MOSS 2007 technologies support and satisfy WCAG 1.0 guidelines, this document considers whether WSS 3.0 and, consequently, MOSS 2007 support the checkpoints, or in some cases, how these products can be adapted to ensure compliance with the checkpoints.

This assessment takes into account the following documents: Windows Sharepoint Services 3.0 Beta 2 VPAT and Office System 2007 Beta 2 Shared Features VPAT.

5.1.1 A "More accessible" Mode

On an MOSS 2007 technologies-based site, most user interface (UI) elements, such as links, form controls, and buttons, are designed to use Microsoft Active Accessibility (MSAA). MSAA enables people with disabilities to interact with content by using assistive technologies (see eponym section 1.2.2 Assistive technologies) such as screen readers, which are devices that provide a synthesized speech or Braille description of what a blind or low-vision user is unable to see on a computer screen or Web site. Proper MSAA names are given on all editable controls, links, and buttons.

Because some custom UI elements do not use MSAA properties, the More Accessible mode enable to render custom controls as an equivalent standard HTML controls so that they will be better recognized by AT devices. "More accessible mode" is exposed via the Document Object Model (DOM) so that third parties can create solutions catered to screen reader users.



More Accessible Mode does not enable alternate versions of every UI element. For example, if a custom control has been added to a MOSS 2007 technologies-based site, enabling More Accessible Mode does not automatically create an alternate version of that control.

The More Accessible Mode can be turned on or off by pressing the TAB key immediately after placing focus on the page in a browser. Press the TAB key until you reach the *Turn on more accessible mode* or *Turn off more accessible mode* link.

Even though the link to turn on More Accessible Mode is the first element in the tab order — which means that it is the first option when interacting with on the page — you may need to press the TAB key more than once to activate the feature. This is because the browser menus or toolbars may initially have the active focus. More Accessible Mode changes the way that the page renders for you, not for other users of the site. Because this is a local setting, no one besides you knows that you enabled this setting. More Accessible Mode remains enabled until you turn it off or close your browser. More Accessible Mode enables the following items to be rendered in a way that optimizes them to work better with AT devices such as screen readers:

- *Menus* - Instead of displaying a drop-down menu of options for files in a document library, a new browser window is opened that contains all of the menu items as hyperlinks. This format is easier for AT devices to interpret. At the same time, the user is informed that a new window will open if he clicks on the menu link; the TITLE attribute contains the Open menu terminology (see point 13.3 in section 5.1.14 Accessible content).

- *Optimized fields* - Some fields are difficult for accessibility tools to interpret. When More Accessible Mode is enabled, these fields are replaced with fields that are optimized for accessibility tools. For example, some WSS 3.0 and MOSS 2007 lists support enhanced text fields that enable users to add formatted text, images, tables, and hyperlinks. Because of the way these fields are rendered in a browser, some accessibility tools cannot read them. When More Accessible Mode is enabled, such fields are replaced with standard plain text fields that are compliant with accessibility tools.

Windows Speech Recognition is a speech based assistive technology that is available in Windows Vista. Windows Speech Recognition enables users to perform actions by speaking (in English) instead of using a keyboard or mouse.

There are known issues related to the use of Windows Speech Recognition with some menus on a WSS 3.0/MOSS 2007 page, including the menus for *New*, *Upload*, *Welcome User* and the drop-down menu of options that are available for files in a document library.

To access these menus by using Windows Speech Recognition, you need to activate More Accessible Mode by doing the following:

1. Open the site home page in a browser, and then say, "Press TAB" until the Turn on more accessible mode link appears;
2. Say, "Press ENTER";
3. To use a split button menu such as the New or Upload button on a list toolbar, or to use the menu of options for a document in a document library, bring the active focus to the element first by saying, "Press TAB", and then say, "Press ALT DOWN ARROW";

A new browser window that contains links from the menu will open.

For menus that have more than one command, say, "Press ALT DOWN ARROW".

In addition to enhanced text fields, several other types of fields are replaced with alternate fields in More Accessible Mode:

Field in standard mode	Alternate field in More Accessible Mode
Enhanced text field	*Multiple line text field*
Graphical summary charts (for surveys)	*Table with table headers (see point 5.3 in section 5.1.6 Tables)*
Gantt chart (in a project list)	*Table with table headers (see point 5.3 in section 5.1.6 Tables). A table with table headers also appears below a Gantt chart in standard mode.*
Contact selector	*Text field*

Finally, screen reader friendly Instant Messaging presence status and menu interaction on presence enabled people fields. ALT text that describes the presence status on people fields have been added on the WSS 3.0 and MOSS 2007 sites, and made ENTER the action of opening the menu for these to simplify the keyboard only experience.

5.1.2 Graphics

1.1: Does each graphic have text to display as an alternative to the graphic?

In HTML, use the ALT attribute with IMG and AREA SHAPE elements, and so forth. It is important to provide alternate text for graphics because some browsers and screen readers cannot interpret or display graphics properly.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_01, 1.1_HTML_07, 1.1_HTML_08

Type of criterion: development

Can be accomplished: Yes

Comments: WSS 3.0 and MOSS 2007 sites use by default descriptive ALT text for images that convey meaning or important information to users.

For specific custom pages, templates (see 10.1 in section 5.1.11 Information layout) and Web pages that satisfy this criterion can be created. You can edit the page with Microsoft Office SharePoint Designer 2007 to take advantage of the tight integration between these products and technologies.

It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

1.2: Is the alternate text for each image relevant to the context in which the image is viewed?

A given image might have several meanings. It is the context of the image that determines the exact meaning. For example, a yellow bullet icon in a newspaper Web site should have the ALT attribute set to ALT="" or ALT="*". However, if the same yellow bullet icon is included in a list of graphics that is available for download, it might make more sense to set the attribute to ALT = "yellow diamond" to distinguish it from other graphics available on the page.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_03

Type of criterion: development

Can be accomplished: Yes

Comments: See 1.1 in this section. This criterion is subjective.

1.3: Are graphics that are used only for decorative purposes commented with ALT=""?

Some graphics have no informative value. An example is the use of transparent pixels only for layout purposes. However, all graphical elements must be commented. For this reason, graphics that are used only for decorative or layout purposes should be commented by setting the ALT attribute to null (in HTML, ALT=""). Some browsers do not display the images that use this type of comment.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_02

Type of criterion: development

Can be accomplished: Yes

Comments: By default, images that are considered spacer images or have no informational value contained in them have `ALT=""`, which tells AT devices (see section 1.2.2 Assistive technologies) that the images do not convey content or information. See also 1.1 in this section.

1.4: Is the alternate text for each image no more than 60 characters long?

You should use a maximum of 60 characters in the ALT attribute because alternate text might bother some users. This is often the case for people who use a screen magnification function because of visual disabilities. It is important to limit the number of characters in the image comments so that such users can minimize the effort required to obtain all the information.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

***Can be accomplished:* Yes**

Comments: See 1.1 in this section.

1.5: Are all comments that are linked to clickable areas of a MAP image relevant?

Definition: a MAP image is an image that presents clickable areas so that the user can click to select an option. When you use a MAP image in HTML (on the client or the server side), it is important to include alternate text for all clickable areas, as well as for the MAP image itself (that is, the image that contains the USEMAP or ISMAP attribute).

For a MAP image on the client side, provide alternate text by using the ALT attribute in each of the clickable areas (AREA SHAPE).

For a MAP image on the server side, provide redundant text links that repeat the content of each of the clickable areas.

Comment 1: The alternate text provided for the image that contains the USEMAP or ISMAP attribute must not be null.

Comment 2: It is recommended that MAP images be used on the client side rather than on the server side.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1, 1.2, 9.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_03, 1.2_HTML_01, 9.1_HTML_01

Type of criterion: development

***Can be accomplished:* Yes**

Comments: WSS 3.0 and MOSS 2007 do not contain and cannot generate standard MAP images.

It remains the responsibility of MOSS 2007 developers, when creating templates (see 10.1 in section 5.1.11 Information layout) that use this feature, to ensure that their Web pages satisfy this criterion. The solution is to systematically fill in the ALT attribute that is associated with the tag containing the USEMAP attribute (in addition to the images). Each clickable area (AREA SHAPE attribute) must include an appropriate ALT attribute.

1.6: Is the alternate content for each text image at least the equivalent of the text appearing in the image?

It is not recommended that you present text in the form of an image. However, if there is an unavoidable graphical constraint (for example, non-equivalent CSS rendering), the alternate text for a graphical element containing text should, at minimum, present the exact content of the text itself. The alternate text can provide additional information, but it must at least provide the exact content.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_03

Type of criterion: development

***Can be accomplished:* Yes**

Comments: See 1.1 in this section. In addition, when an image file is uploaded to a site, alternative ALT text to that image can be added. For example, the Site Image Web Part contains a field in which you can type custom ALT text when you add an image. When you add pictures to a picture library, you can define custom ALT text for the pictures.

1.7: Do all images that require a detailed description provide comment text?

When a graphical element requires a long description in HTML (for example, a complex image or alternate text longer than 60 characters) a text comment must be provided. The comment can be a link to a descriptive file (for example, a D-link or a clickable graphic), a caption that appears directly on the page, or a LONGDESC attribute.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_05

Type of criterion: development

Can be accomplished: Yes

Comments: WSS 3.0 and MOSS 2007 can be used to create templates (see 10.1 in section 5.1.11 Information layout) and Web pages that satisfy this criterion. The LONGDESC attribute can be added by using Microsoft Office SharePoint Designer 2007 to edit the page. The best approach is to create a site definition that integrates this criterion.

The WCAG directive specifies that the LONGDESC attribute should be used, but currently most browsers do not support it. Internet Explorer 6.0 exposes this attribute from a programming point of view in its model object. Therefore, assistive technologies can make use of it. Longdesc Linker for Internet Explorer 6 is a free downloadable ActiveX component that makes it possible to add a Long Description element to the context menu used by Internet Explorer for images. This element is enabled if the selected image has a LONGDESC link. If the user selects the link (by holding down the SHIFT key while selecting), Internet Explorer follows it.

1.8: If a detailed description is provided for an image, is the content relevant?

The content should provide additional information to the description defined in the ALT attribute for the image.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_04

Type of criterion: development

Can be accomplished: Yes

Comments: See 1.7 in this section. This criterion is subjective.

1.9: Does the text used in the ALT attribute for each image provide the function of the link?

The image description is not always in keeping with the function of the link. For example, if an arrow is used to indicate a link, use ALT="Next" instead of ALT="Right Arrow". Avoid using ALT="", which provides no information about the function of the link.

WCAG 1.0 checkpoint(s) [Priority 1]: 13.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_02

Type of criterion: development

Can be accomplished: Yes

Comments: In terms of Wording of links, WSS 3.0 and MOSS 2007 try to use wording that makes more sense when read out of the context, i.e. *Add Event* rather than *Add*.

Generally speaking, templates (see 10.1 in section 5.1.11 Information layout) and Web pages satisfy this criterion can be created that. The solution is to place an ALT attribute between the <IMG> tags embedded in an <A> tag, making sure that the text is relevant.

It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

5.1.3 Frames

2.1: Does each frame have a NAME attribute?

In HTML, each frame must be named by using the NAME attribute. Browsing through frame-based Web pages can be complex, especially for visually impaired people who become familiar with site content frame by frame (whereas sighted people have an overview of the entire site). Comment: The content of the NAME attribute should not include any spaces.

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: WSS 3.0 and MOSS 2007 are not frame-based by default. However, it is possible to create templates (see 10.1 in section 5.1.11 Information layout) and pages that have frames. It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

2.2: Are the names assigned to frames relevant?

The names assigned to frames should be relevant. Avoid naming frames with directional data, such as "Left," "Right," or "Middle." It is better to name frames with explicit names that refer to the function provided by the frame, such as "Menu," "Navigation," or "Content."

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.3: Is there a NOFRAME tag?

In HTML, the NOFRAME tag is used to provide the user with a summary of the main links on the Web site. This tag facilitates browsing for browsers that do not handle frames.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 6.5
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.5_HTML_02

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.4: Is the content of the NOFRAME tag relevant?

If a NOFRAME tag is present, the information in the tag must be relevant. In many cases, the NOFRAME tag is present but provides information that is either inappropriate or incomplete. This tag is intended to provide the main links existing on the Web site.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 6.5
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.5_HTML_02

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.5: Does each frame have a TITLE attribute?

In HTML, the TITLE attribute is used to describe, in a few words, the purpose of the frame. This attribute is complementary to the NAME attribute.

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 12.1_HTML_01

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.6: Is the content of the TITLE attribute relevant?

Some assistive technologies can recognize this attribute. The content of this attribute helps the user navigate from frame to frame independently and quickly determine the function of each frame.

WCAG 1.0 checkpoint(s) [Priority 1]: 12.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 12.1_HTML_02

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.7: Does each page have a maximum of three frames?

It is important to have no more than three frames on any one page. If more than three frames are used, the page becomes difficult to read, especially for users of assistive technologies.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

2.8: When frames are used, is scrolling automatic?

In HTML, a frame can no longer be scaled if scrolling is set to "NO." The scrolling function must be enabled. For users of magnification software, which might implement user-specific style sheets with larger fonts, and so forth, in-frame scrolling must be enabled to allow all content to be accessed.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 14.3

UWEM 1.0 Test(s): -

Type of criterion: design / development

Can be accomplished: N/A (default) / Yes

Comments: See 2.1 in this section.

5.1.4 Colors

3.1: Is information provided by color still readable when colors are disabled?

Web site designers must be particularly careful about information provided by colors. This information can be lost to colorblind users or users who do not have adequate visual acuity to distinguish between certain colors.

WCAG 1.0 checkpoint(s) [Priority 1]: 2.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 2.1_HTML_01, 2.1_HTML_02, 2.1_HTML_03, 2.1_CSS_01

Type of criterion: design

Can be accomplished: Yes

Comments: WSS 3.0 and MOSS 2007 sites do not rely on the use of color alone to convey information.

Furthermore, WSS 3.0 and MOSS 2007 can be used to create templates (see 10.1 in section 5.1.11 Information layout) and Web pages that satisfy this criterion. It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

3.2: Is there enough contrast between colors to be distinguishable by users who have impaired color vision?

Some color combinations should be avoided, such as red with blue, yellow with violet, yellow with green, and so forth. When there is not enough contrast between colors on a page, the information provided by contrast might not be accessible to users with poor or impaired color vision.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 2.2
WCAG 1.0 checkpoint(s) [Priority 3]: 2.2

UWEM 1.0 Test(s): 2.2_HTML_01, 2.2_CSS_01

Type of criterion: design

Can be accomplished: Yes

Comments: See 3.1 in this section.

WSS 3.0 and MOSS 2007 natively offer support for High Contrast color schemes: the pages in a WSS 3.0/MOSS 2007 site are designed to display correctly if a High Contrast color scheme is applied.

This is integrated in the site definitions as originally described in the MSDN article Tips for Enhancing the Accessibility of SharePoint Web Sites in section "Create and Apply High-Contrast Themes to Web Sites."

It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

To assess this criterion, a color contrast evaluation tool, such as the free downloadable software by Vision Australia, can be used to verify foreground and background color combinations to determine whether the color contrast enables proper viewing. The identification of proper color visibility is based on the algorithms provided by the W3C.

(At the same time, browsers with Optical Zoom features can easily resize text on WSS 3.0/MOSS 2007 sites.)

5.1.5 Multimedia

4.1: Can the information that is conveyed by multimedia be provided another way?

Whenever multimedia is used to provide information, it is important to provide an alternative for those who do not have access to multimedia. For example, if a multimedia file is made available on a Web site, it is important to provide an accessible text-based alternative in a standard format, such as RTF, TXT, or HTML.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.3, 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.3_external_01, 1.3_external_02, 1.1_HTML_01, 1.1_HTML_02, 1.1_HTML_03, 1.1_HTML_04, 1.1_HTML_05, 1.1_HTML_06, 1.1_HTML_09, 1.1_HTML_10, 1.1_external_01, 1.1_external_02, 6.3_HTML_01

Type of criterion: design / development

Can be accomplished: Yes

Comments: Very few animations in used within WSS 3.0/MOSS 2007 sites and when they are used, an ALT text description of what the animated image is doing is systematically provided. Training videos have an equivalent text based representation available.

This said, WSS 3.0 and MOSS 2007 can be used to create templates (see 10.1 in section 5.1.11 Information layout) and Web pages that satisfy this criterion. It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

4.2: Is the Multimedia content synchronized with the alternate support?

If a video with sound is used on a site, as in the case of an interview between several people, it is important to synchronize the alternate support with the picture and the sound in the video. The W3C recommends the use of the Synchronized Multimedia Integration Language (SMIL) programming language.

WCAG 1.0 checkpoint(s) [Priority 1]: 1.4, 1.3
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.4_external_01

Type of criterion: design / development

Can be accomplished: Yes

Comments: This criterion requires the use a tool such as Hi-Caption from HiSoftware, which is a Synchronized Multimedia Integration Language (SMIL) presentation editor, or Microsoft Synchronized Accessible Media Interchange (SAMI). Please refer to the technical articles Understanding SAMI 1.0 and Adding Closed Captions to Digital Media for addition information about how to add subtitle tracks to the Windows Media format.

You might want to refer to HiSoftware Hi-Caption Studio - Creating Accessible Demos as an illustration.

It remains the responsibility of MOSS 2007 developers to ensure that their multimedia content satisfies this criterion.

5.1.6 Tables

5.1: Is the SUMMARY attribute present and relevant?

If the SUMMARY attribute is used correctly, it provides additional information to disabled users, such as the visually impaired, about the tables in a Web page. If a Web page contains menu bars that are created by using an HTML table, setting SUMMARY="Menu bar" provides the user with an additional information about the table function.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 5.5

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: Yes

Comments: Tables are often used in WSS 3.0 and MOSS 2007 especially for laying out information on a page and systematically have the SUMMARY attribute properly set.

5.2: In a data table, does the CAPTION tag provide the title of the table?

Rather than using formatted text to provide the title of a table, it is better to use the CAPTION tag, which is the tag dedicated to displaying table titles.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 5.5

UWEM 1.0 Test(s): 12.3_HTML_07, 12.3_HTML_08, 5.4_HTML_04

Type of criterion: development

Can be accomplished: Yes

Comments: WSS 3.0 and MOSS 2007 can be used to create templates (see 10.1 in section 5.1.11 Information layout) and Web pages that satisfy this criterion. It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

5.3: In data tables, are the column headers appropriate?

In HTML data tables, the TH tag specifies column headers, enabling users of assistive technologies to use text-to-voice software to determine that they are column headers.

WCAG 1.0 checkpoint(s) [Priority 1]: 5.1
WCAG 1.0 checkpoint(s) [Priority 2]: 5.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 5.1_HTML_01, 5.4_HTML_01, 5.4_HTML_01

Type of criterion: development

Can be accomplished: Yes

Comments: For tabular data tables, table header cells (TH tags) are systematically used. Column heading cells are defined:

```
<TH scope="col">Last Name</TH>
```

For example, on the *Announcements* page, in the default *All Items* view, the column headers *Title* and *Modified* are contained in table header cells.

Similarly, row heading cells are defined where needed:

```
<TH scope="row">Item Code</TH>
```

5.4: In a data table, does a HEADERS attribute link to each of the data cells in the table?

When correctly configured in a data table, the HEADERS and ID attributes make it possible to link each data cell to the appropriate column header. This approach enables users that have only a linear view of the table to know which header is linked to the current data cell.

WCAG 1.0 checkpoint(s) [Priority 1]: 5.2
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 5.2_HTML_01, 5.2_HTML_02, 5.4_HTML_03

Type of criterion: development

Can be accomplished: Yes

***Comments:* -**

5.5: Is the content in formatted tables in correct sequence?

In formatted tables, the information must be ordered so that it is displayed in an equivalent sequence when the information is read in a linear fashion. Care must be taken when organizing information in a formatted table. HDL development tools that are currently available have a tendency to nest tables within one another. The result is that occasionally differences occur between the visual appearance of information in a graphical browser and the text equivalent, because the latter is based on the structure of the HTML code (as in LYNX-type text browsers, for example).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 5.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 5.3_HTML_01

Type of criterion: design

Can be accomplished: Yes

***Comments:* -**

5.1.7 Links

6.1: Are Link titles no more than 80 characters long?

Use a maximum of 80 characters in hypertext link titles (accents and spaces count for one character as well).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: design / development

Can be accomplished: Yes

Comments: The standard Web Part for link handling enables authors to publish links with titles containing a maximum of 256 characters. It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion with a title of no more than 80 characters.

6.2: Are links explicit enough?

Links must be understandable independent of the page context. Some assistive technologies are able to display a list of available links. These links must be understandable without the context of the Web page to enable rapid scanning. Avoid links such as "click here."

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_02

Type of criterion: design / development

Can be accomplished: Yes

Comments: This criterion is subjective.

6.3: Is the TITLE attribute used, if required, and is it no more than 80 characters long?

In HTML, the TITLE attribute is necessary if additional information is required to understand a link. For example, if a link is used to download a file, the TITLE attribute can provide the file name, file size in bytes, file format, and destination ("open in new window").

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: Yes

Comments: -

6.4: Does the TITLE attribute provide more information about the link than the link title itself?

The only information that some users have about a link is the title of the link or the content of the TITLE attribute. Therefore, the content of the TITLE attribute should, at the very least, repeat the information contained in the link title. For example, a TITLE attribute that simply indicates "this link opens in a new window" is not adequate, because the user does not know the link destination. However, "Read article on tourism in China (new window)" provides all the essential information required about the link.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_01, 13.1_HTML_02

Type of criterion: - development

Can be accomplished: Yes (see 6.3)

Comments: See 6.3 in this section.

6.5: Do all identical link titles lead to the same target?

Avoid using link titles such as "Read article" multiple times on a page. It is important to use link titles that are not identical to help people who wish to move from link to link to read the page more quickly. It is also important to use short, explicit links with unambiguous destinations.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.1_HTML_01

Type of criterion: design

Can be accomplished: Yes

Comments: -

5.1.8 Scripts

7.1: If a script requires alternate text to make it accessible, is the information provided by the alternate text equivalent to the information provided by the script?

The information about a script should remain available even if the script is disabled. For example, if a JavaScript script is used to validate an online form, the alternative might be a server-side data entry control. Scripts can be disruptive to users of assistive technologies that do not support scripts or that support them only in a limited fashion. In addition, some users voluntarily disable script execution in standard browsers. For this reason, an alternative must be available to retain the information generated by the scripts.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.2, 6.3, 8.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.2_HTML_02, 6.2_HTML_03, 6.3_HTML_02

Type of criterion: design / development

Can be accomplished: Yes

Comments: As previously described (see section 5.1.1 A "More accessible" Mode), a "more accessible" mode allows users with special needs to identify themselves so that the site can change the way some of the dynamic content is rendered. With this mode, the JavaScript menus are rendered as plain HTML links in a new window making it easier for users to interact with the menu. At the same time, the user is informed that a new window will open if he clicks on the menu link; the TITLE attribute contains the Open menu terminology (see point 13.3 in section 5.1.14 Accessible content).

Furthermore, WSS 3.0 and MOSS 2007 can be used to create templates (see 10.1 in section 5.1.11 Information layout) and Web pages that satisfy this criterion. It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

7.2: Can actions be performed even if the peripheral for which they were designed is disabled?

Some functions of a Web site, such as those generated by JavaScript scripts, can be activated only by a mouse click. However, some users cannot use a mouse to browse through a Web site. It must be possible to browse through a Web site by using just a keyboard.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 6.4, 9.2, 9.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 9.2_external_01, 6.3_HTML_02, 6.4_HTML_01, 6.4_HTML_02, 6.4_external_01, 8.1_HTML_01, 8.1_external_01, 9.3_HTML_01

Type of criterion: design / development

Can be accomplished: Yes

Comments: Compared to the SharePoint 2003 technologies, MOSS 2007 technologies provide smarter and more intuitive use of access keys on common commands. The UI controls, features and commands are available directly by using the keyboard. Every action a mouse user can perform is available to keyboard user. For example, the MOSS 2007 technologies offer a keyboard accessible date pickers.

You can press the TAB key and SHIFT+TAB to move back and forth between elements on any page, ALT+<access key> to bring focus to commonly used controls making it easier for users to activate these common controls. The access keys that make sense are privileged while attempting to avoid duplicate access keys on a page and the use of Internet Explorer reserved access keys. Thus, an OK button uses ALT+O on all forms pages.

You can also find keyboard shortcuts for many commands at the following Internet address: http://office.microsoft.com/en-us/help/HA101733621033.aspx.

It is recommended that users with AT devices activate the More Accessible mode (see eponym section 5.1.1 A “More accessible” Mode). As previously explained, this mode enables users to interact with the elements of a WSS 3.0/MOSS 2007 site by using assistive technologies (see eponym section 1.2.2 Assistive technologies) such as screen readers.

Templates (see 10.1 in section 5.1.11 Information layout) and Web pages that satisfy this criterion can be created. It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

5.1.9 Mandatory elements

8.1: Is the DOCTYPE tag present at the beginning of the page source code?

International WAI guidelines suggest using an appropriate and valid Document Type Definition (DTD) to indicate the type of electronic document that the browser must load (frame-based, HTML, XML, and so forth). In HTML, the DOCTYPE tag is required to define the document type.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 3.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 3.2_HTML_01

Type of criterion: development

Can be accomplished: Yes

Comments: The DOCTYPE tag is not available by default.

WSS 3.0 and MOSS 2007 can be used to create templates (see 10.1 in section 5.1.11 Information layout) and Web pages that satisfy this criterion. It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

Note that the current versions of WSS and MOSS do not strictly comply with the formal syntax requirements specified in HTML 4.01 or XHTML 1.0 as required by checkpoint 3.2.

8.2: Is the LANG attribute present at the beginning of the page source code to clearly identify the language used?

It is important to specify the language of a document to enable assistive technologies to adapt content to the language. In HTML, the LANG attribute in the HTML tag specifies the language.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 4.3

UWEM 1.0 Test(s): -

Type of criterion: development

Can be accomplished: Yes

Comments: The LANG attribute is not used by default.

WSS 3.0 and MOSS 2007 can be used to create templates (see 10.1 in section 5.1.11 Information layout) and Web pages that satisfy this criterion. It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

8.3: Is there a TITLE tag in the page header?

In HTML, the title of a page is coded by including a TITLE tag in the document header. Voice browsers list the title of the page first so that a user can get positioned on the Web site. The title is a constant source of support to those having an attention deficit disorder. The tag also makes it easier for search engines to reference the Web site.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.2_HTML_01

Type of criterion: development (development rules)

Can be accomplished: Yes

Comments: -

8.4: Is the content of the TITLE tag explicit?

The TITLE tag provides the title of the page that was just loaded. It is important to provide an explicit title to help users get oriented during browsing.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.2_HTML_01

Type of criterion: design

Can be accomplished: Yes

Comments: See 8.3 in this section. This criterion is subjective.

8.5: Is the content of the TITLE tag different from one page to the next?

Using an identical title on every page of a Web site can be confusing to users. The title of the page is the first information that appears on the screen. Assigning the same title to multiple documents is misleading.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.2
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): -

Type of criterion: design

Can be accomplished: Yes

Comments: See 8.3 in this section. This criterion is subjective.

8.6: Are language changes on a page indicated?

Whenever a language change occurs on a page, it should be indicated to the user. If text-to-voice software is used, a change in language causes the software detection language to change as specified by the LANG tags.

WCAG 1.0 checkpoint(s) [Priority 1]: 4.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 4.1_HTML_01, 4.1_HTML_02, 4.1_HTML_03, 4.1_HTML_04, 4.1_CSS_01, 4.1_CSS_02

Type of criterion: development

Can be accomplished: N/A

Comments: Compared to SharePoint 2003 technologies, MOSS 2007 technologies now support multilingual websites. The implementation of a multilingual environment is based on a new concept called "variations".

This is based on a page or site that is marked as a "source label". Other languages can inherit from that source label and use the defined settings for its specific language. For additional information, you can refer to the article SharePoint 2007 Beta 2: Variations (multi language) on sub sites on Steven Van de Craen's Blog.

Taking into account the adopted approach, a language change should normally not occur on a page. If it does, it probably originates from a WSS 3.0 or MOSS 2007 control.

5.1.10 Information structuring

9.1: Is information structured consistently for the general context of the site?

The structure of information on a page depends on the type of site (news, gaming, scientific, artistic, and so forth). In addition, the individual elements that provide information on the page (such as images and titles) might not be a problem in and of themselves, but the arrangement of these elements in relation to one another might not be consistent. For example, on a news site, no text should be displayed before the title.

WCAG 1.0 checkpoint(s) [Priority 1]: 14.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 12.3_HTML_10, 12.3_HTML_15, 14.1_HTML_01

Type of criterion: design

Can be accomplished: Yes

Comments: This criterion is subjective.

9.2: Is the Web page presented in a consistent fashion?

Use of styles that are available in HTML, for example H1 and H2, can provide structure to a Web page. Such tags are extremely important in structuring a Web page. For example, the H1 tag indicates a title and the H2 tag indicates subheadings. These tags must be used for their intended purpose, that is, to provide structure, and not for their graphic capabilities. Additionally, blocks of information should be separated from each other.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 3.5
WCAG 1.0 checkpoint(s) [Priority 3]: 13.8

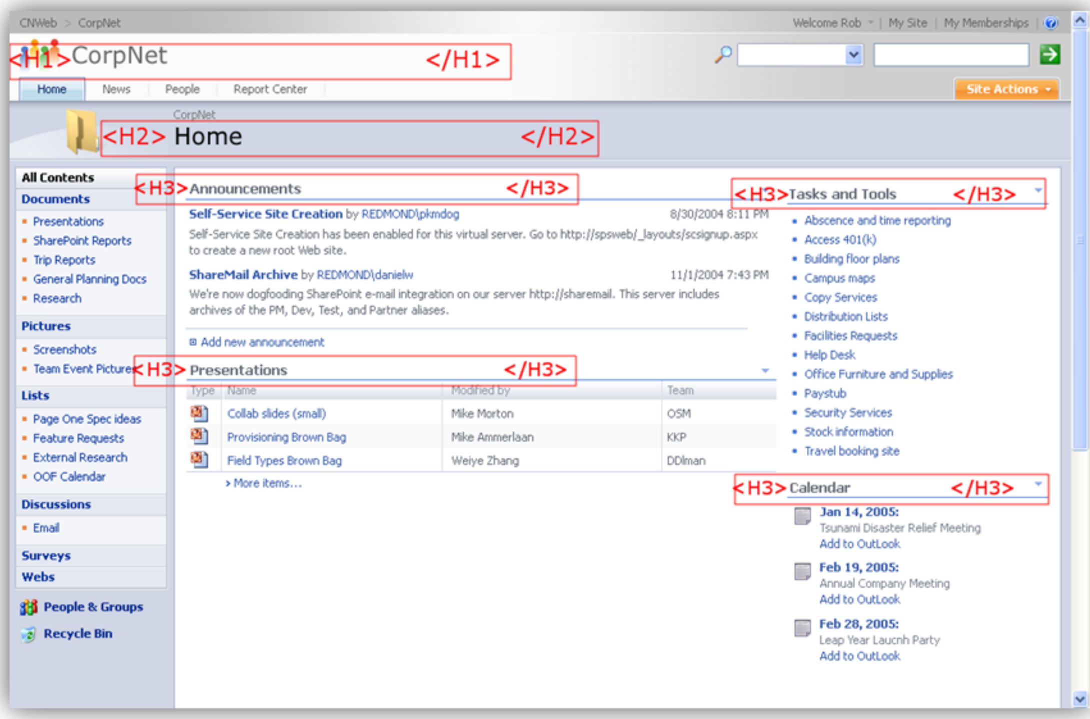
UWEM 1.0 Test(s): 3.5_HTML_01, 3.5_HTML_02, 3.5_HTML_03, 3.5_HTML_04, 3.5_HTML_05

Type of criterion: design

Can be accomplished: Yes

Comments: Tags are systematically used on WSS 3.0/MOSS 2007 sites to help convey the structure of the page to users of assistive technologies (see eponym section 1.2.2 Assistive technologies). This helps AT device users better understand the layout and structure of the pages.

The heading tags (H1, H2, H3, and H4) build a logical outline of the content on a page. If you use an assistive technology such as a screen reader, you can configure it to read only the heading tags on a page. This virtual outline of heading tags helps to improve the clarity of the page content. For example, the home page for a site contains one H1 tag that contains the site title, one H2 tag that contains the page title, an H3 tag around the title of each Web Part.



Self-Service Site Creation by REDMOND\pkmdog 8/30/2004 8:11 PM

Self-Service Site Creation has been enabled for this virtual server. Go to http://spsweb/_layouts/scsignup.aspx to create a new root Web site.

ShareMail Archive by REDMOND\danielw 11/1/2004 7:43 PM

We're now dogfooding SharePoint e-mail integration on our server http://sharemail. This server includes archives of the PM, Dev, Test, and Partner aliases.

Type	Name	Modified by	Team
	Collab slides (small)	Mike Morton	OSM
	Provisioning Brown Bag	Mike Ammerlaan	KKP
	Field Types Brown Bag	Weiye Zhang	DDlman

- Absence and time reporting
- Access 401(k)
- Building floor plans
- Campus maps
- Copy Services
- Distribution Lists
- Facilities Requests
- Help Desk
- Office Furniture and Supplies
- Paystub
- Security Services
- Stock information
- Travel booking site

Jan 14, 2005: Tsunami Disaster Relief Meeting

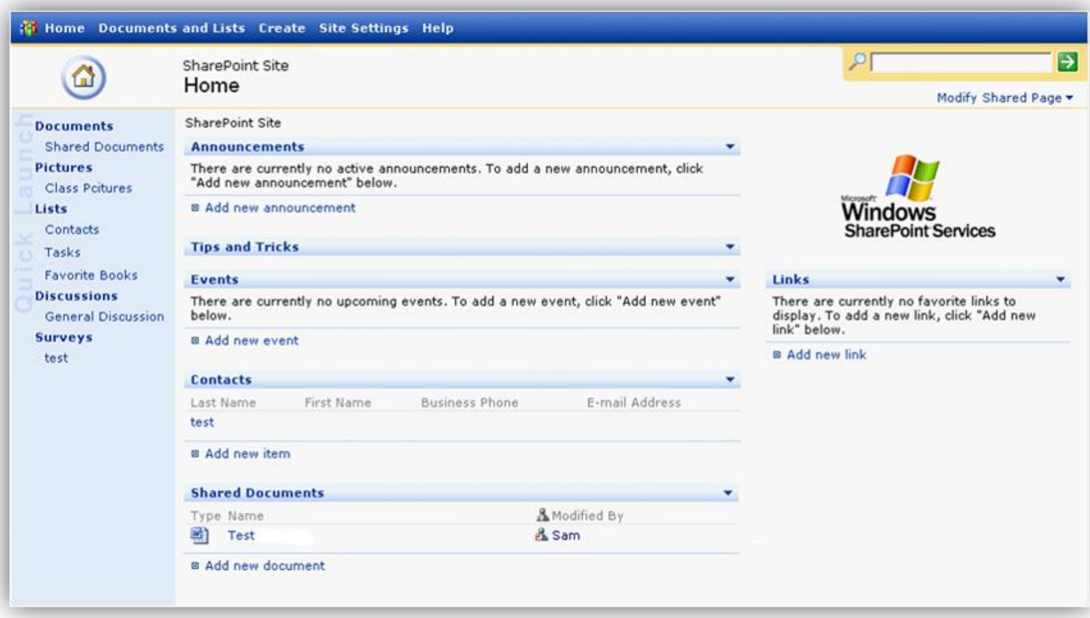
Feb 19, 2005: Annual Company Meeting

Feb 28, 2005: Leap Year Laucnh Party

Users of certain AT devices can retrieve an outline based on these tags and use it to easily understand the layout of the page and skip to the heading that they want.

As far as screen readers are concerned, this comes with further improvements in terms of navigation compared to SharePoint 2003 technologies.

For instance, each time a user visits a WSS 2.0 home page he will have to listen to the top navigation bar links, left side navigation links and finally the content of the page. The tab order is structured in the same way. Users will have to tab multiple times to get to the page content. In MOSS 2007 technologies, a *Skip to main content* link helps users to quickly jump to the most important content without having to tab through our navigation links (see 12.1 in section 5.1.13 Browsing aids).



There are currently no active announcements. To add a new announcement, click "Add new announcement" below.

There are currently no upcoming events. To add a new event, click "Add new event" below.

Last Name	First Name	Business Phone	E-mail Address
test			

Type	Name	Modified By
	Test	Sam

There are currently no favorite links to display. To add a new link, click "Add new link" below.

Moreover, the reading of the page has also evolved. As an illustration, here is how a screen reader reads the above WSS 2.0 homepage. Items in [ ] are ALT text descriptions, underlined items are links.

[Logo.]
*Home*Documents and ListsCreateSite SettingsHelp
[Icon.]
SharePoint Site
Home
(Start of form1.)[Text.]
(End of form1.)
[images/gosearch.gif.]
Modify Shared Page[Change Web Parts in shared view. These changes will apply to all users..]
Documents
[Icon.]
Shared Documents
Pictures
[Icon.]
Class Pcitures
Lists
[Icon.]
Contacts
[Icon.]
Tasks
[Icon.]
Favorite Books
Discussions
[Icon.]
General Discussion
Surveys
[Icon.]
test
[Icon.]
(Start of form2.)
SharePoint Site
Announcements
[Web Part Menu.]
There are currently no active announcements. To add a new announcement, click "Add new announcement" below.
Add new announcement
Tips and Tricks
[Web Part Menu.]
Events
[Web Part Menu.]
There are currently no upcoming events. To add a new event, click "Add new event" below.
Add new event
Contacts
[Web Part Menu.]
Last Name
First Name
Business Phone
E-mail Address
test
Add new item
Shared Documents
[Web Part Menu.]
Type
Name
[Presence enabled for this column.]
Modified By
[Icon.]
Test
Sam
Add new document
[/_layouts/images/homepage.gif.]
Links
[Web Part Menu.]
There are currently no favorite links to display. To add a new link, click "Add new link" below.
Add new link
(End of form 2.)

In WSS 3.0, here is how the homepage now looks to a screen reader. The red text existed in WSS 2.0 but has been removed (also identified by <removed>), and yellow text has been added in WSS 3.0 (also identified by <added>).

```
Skip to Main Content  <added>
[Logo.]  <removed>
HomeDocuments and ListsCreateSite SettingsHelp
[Icon.]  <removed>
SharePoint Site
Home
[Search] <added>
(Start of form1.)[Text.]
(End of form1.)
[images/gosearch.gif.]  <removed>
Go <added>
Modify Shared Page[Change Web Parts in shared view. These changes will apply to all users..]
Documents
[Icon.]  <removed>
Shared Documents
Pictures
[Icon.]  <removed>
Class Pcitures
Lists
[Icon.]  <removed>
Contacts
[Icon.]  <removed>
Tasks
[Icon.]  <removed>
Favorite Books
Discussions
[Icon.]  <removed>
General Discussion
Surveys
[Icon.]  <removed>
test
[Icon.]  <removed>
//note: the Skip to Main Content link takes users here <added>
(Start of form2.)
SharePoint Site
Announcements <removed>
Announcements <add H1 tag to this>
[Web Part Menu.]
There are currently no active announcements. To add a new announcement, click "Add new announcement"
below.
Add new announcement
Tips and Tricks <removed>
Tips and Tricks <add H1 tag to this>
[Web Part Menu.]
Events  <removed>
Events  <add H1 tag to this>
[Web Part Menu.]
There are currently no upcoming events. To add a new event, click "Add new event" below.
Add new event
Contacts <removed>
Contacts <add H1 tag to this>
[Web Part Menu.]
Last Name
First Name
Business Phone
E-mail Address
test
Add new item
Shared Documents <removed>
Shared Documents <add H1 tag to this>
[Web Part Menu.]
Type
Name
[Presence enabled for this column.]
Modified By
[Icon.]  <removed>
[Test.doc] <added>
Test
Sam
Add new document
[/_layouts/images/homepage.gif.]  <removed>
```

Custom ALT text for the Image w eb part. By default ALT="Window s SharePoint Services site" for out of the box WSS image. <added>
Links <removed>
Links <add H1 tag to this>
[Web Part Menu.]
There are currently no favorite links to display. To add a new link, click "Add new link" below .
Add new link
(End of form 2.)

As illustrated above, understanding the layout and the structure of the pages is greatly facilitated.

For specific custom pages, templates (see 10.1 in section 5.1.11 Information layout) and Web pages that satisfy this criterion can be created. It therefore remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

5.1.11 Information layout

10.1: Is page content separated from content introduction?

If a page is coded in HTML, it should not contain formatting information (for example, BGCOLOR, FONT, ALIGN, B, I), because all this information can be included in style sheets.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 3.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 3.3_HTML_01, 3.3_HTML_02

Type of criterion: development (development rules)

Can be accomplished: Yes

Comments: Publishing in WSS 3.0 and MOSS 2007 technologies is fundamentally a means of authoring and displaying information by using a set of templates. There are two fundamental templates to understand: master pages and page layouts.

Master pages (in both WSS 3.0 and MOSS 2007) and page layouts (only in MOSS 2007) dictate the overall look and feel of your MOSS 2007 technologies-based site.

Master pages contain controls that are shared across multiple page layouts, such as navigation, search, or language-preference for multilingual sites. Master pages are intended to easily provide a consistent look and feel for the site. They can be used to position items that must be shared by all pages, such as navigational controls, company logos, and copyright notices. ASP.NET 2.0 user controls, Web server controls, and Web Parts can be used within master pages

Page layouts contain field controls and Web Parts. A page layout provides a structured approach to collecting content from content authors and displaying it on a page within a site. Each page layout is based on a WSS 3.0 content type and an associated ASP.NET 2.0 page template. In addition to field controls, an ASP.NET 2.0 page template for a page layout can also contain ASP.NET server controls and Web Part zones.

It remains the responsibility of MOSS 2007 developers to ensure that their templates satisfy this criterion.

The top-level site for a site collection hosted on WSS 3.0 or MOSS 2007 has a special document library called the Master Page and Page Layout Gallery. All page layouts and master pages are stored in this document library. The Master Page and Page Layout Galleries support versioning and workflow.

When a content migration from MCMS 2002 is performed, MOSS 2007 creates a set of folders for keeping the Template Gallery structure. In each folder, it generates a basic, default page layout for each template gallery item that was present in MCMS 2002. For additional information, you can refer to the white paper Planning MCMS 2002 Application Migration to SharePoint Server 2007.

10.2: If style sheets are disabled, is the information still accessible?

To separate content from layout on an HTML page, use style sheets (CSS) to specify the layout. However, people who do not wish to read the page by using the author-selected style sheet should obtain the same information when they disable the page style sheet and use their own customized settings instead.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 4.1_CSS_02, 6.1_HTML_01, 6.1_HTML_02

Type of criterion: design / development

Can be accomplished: Yes

Comments: It remains the responsibility of MOSS 2007 developers to ensure that their templates (see 10.1 in this section) and Web pages satisfy this criterion.

10.3: If style sheets are disabled, is the order in which information appears the same as initially defined?

Style sheets can be used to construct HTML pages. If style sheets are disabled by the user, the logical structure of the page should stay the same as initially defined. For example, the order of appearance of the divisions (DIV) should stay the same when viewed by using a graphical browser or by using a text-mode browser.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: 9.4

UWEM 1.0 Test(s): 3.6_HTML_01, 6.1_HTML_01, 6.1_HTML_02

Type of criterion: design / development

Can be accomplished: Yes

Comments: Templates (see 10.1 in this section) and Web pages that satisfy this criterion can be created. It therefore remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

This said, compared to the SharePoint 2003 technologies, please note that MOSS 2007 technologies provide an improved and more intuitive keyboard tab order (generally from near to far, top to bottom).

The page and navigation elements on a site follow a logical and intuitive tab order. The tab order is the order in which you move the focus from one UI element to another by pressing the TAB key.

5.1.12 Forms

11.1: Are the LABEL tag and its corresponding attributes (ID, FOR) present?

In HTML pages, the LABEL tag is used with the ID and FOR attributes to link data entry fields directly to the associated text. This approach enables people who use assistive technologies to unambiguously determine which data entry field corresponds to the linked text.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 10.2, 12.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 10.2_HTML_01, 12.4_HTML_01, 12.4_HTML_02

Type of criterion: development

Can be accomplished: Yes

Comments: The LABEL tag is not used by default.

WSS 3.0 and MOSS 2007 can be used to create templates (see 10.1 in this section) and Web pages that satisfy this criterion. It remains the responsibility of MOSS 2007 developers to ensure that their templates and Web pages satisfy this criterion.

11.2: In a form, is the SUBMIT button relevant?

The OK button or Submit button is usually the final validation step in filling out an online form. If this button is not clearly identified, it is difficult for the user to interact with the Web site. For example, if the SUBMIT button is an image, comment the image with explicit text, such as "Submit Form."

WCAG 1.0 checkpoint(s) [Priority 1]: 1.1
WCAG 1.0 checkpoint(s) [Priority 2]: -
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 1.1_HTML_02, 1.1_HTML_03

Type of criterion: development

Can be accomplished: Yes

Comments: -

11.3: Is the data entry control in online forms accessible?

If an online form is validated by using a script on the client, for example, provide an alternative to the script so that the user, using any type of browser, can submit the information from the form to the server. Additionally, if an HTML data entry control is used, mandatory data entry fields should be specified explicitly (for example, mandatory fields are preceded by an asterisk). Developers should not use colors to indicate mandatory fields or specify the mandatory fields only at the bottom of the page.

WCAG 1.0 checkpoint(s) [Priority 1]: 6.3
WCAG 1.0 checkpoint(s) [Priority 2]: 6.5
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.2_HTML_02, 6.2_HTML_03, 6.3_HTML_01, 6.3_HTML_02, 6.5_HTML_03

Type of criterion: development

Can be accomplished: Yes

Comments: -

5.1.13 Browsing aids

12.1: Is the main navigation menu on the Web site located in the same place on all pages?

When users browse through a site, they find it easier to locate a menu or navigation toolbar if it is always in the same place on all pages of the Web site. If browsing mechanisms differ from one page to the next, it creates difficulties for users and might cause them to abandon the Web site.

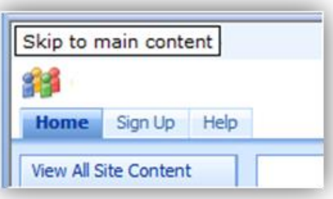
WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.4
WCAG 1.0 checkpoint(s) [Priority 3]: 14.3

UWEM 1.0 Test(s): 13.4_HTML_01

Type of criterion: design

Can be accomplished: Yes

Comments: Compared to the SharePoint 2003 technologies, MOSS 2007 technologies provide two important options, i.e. the *Turn on more accessible mode* link and the *Skip to main content* link. These two hidden links are the first two options in the tab order.



As previously mentioned in section 5.1.1 A “More accessible” Mode, if you want to turn on More Accessible mode, you can do so by pressing the TAB key after the page in the browser has active focus. In addition, if you want to skip repetitive navigation links and go directly to the main content area of the page, you can do so by pressing the TAB key until you reach the Skip to main content link.

Skip to main content links to help visually impaired users quickly jump to the most important content without having to tab through our navigation links.

In addition, standard navigation bread crumbs to assist with site navigation and orienting users on where they are in the site. The pages in a WSS 3.0/MOSS 2007 site use breadcrumb navigation links that identify where the current page or site fits into the overall hierarchy of the site or site collection. These navigation links appear near the top of most pages, under the page name. For example, if you browse to a folder named *Reports* in the *Shared Documents* library for the *Contoso* site, the navigation links may display *Contoso* > *Shared Documents* > *Reports*.

12.2: If keyboard shortcuts are defined for the site, are they active on the page?

If keyboard shortcuts are set up for a page, these shortcuts should be activated and homogeneous throughout the pages of the Web site.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 13.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 13.4_HTML_01

Type of criterion: design / development

Can be accomplished: Yes

Comments: Compared to the SharePoint 2003 technologies, MOSS 2007 technologies provide smarter and more intuitive use of access keys on common commands. The UI controls, features and commands are available directly by using the keyboard.

You can press the TAB key and SHIFT+TAB to move back and forth between elements on any page. You can also find keyboard shortcuts for many commands at the following Internet address: http://office.microsoft.com/en-us/help/HA101733621033.aspx.

5.1.14 Accessible content

13.1: Can the user control screen refresh?

When screen content is automatically refreshed, the cursor in screen readers (such as JAWS) is automatically reset to the top of the page, making it difficult for the user to read the entire page.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 7.4
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 7.4_HTML_01, 7.4_HTML_02, 7.4_external_01

Type of criterion: development

Can be accomplished: Yes

Comments:

13.2: If the user is automatically redirected, is it without using a script?

It is best to use automatic redirection solutions from the server side. In this way, redirection does not depend on the client workstation.

WCAG 1.0 checkpoint(s) [Priority 1]: -

WCAG 1.0 checkpoint(s) [Priority 2]: 7.5
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 7.5_HTML_01, 7.5_HTML_02, 7.5_external_01

Type of criterion: Design / development (development rules)

Can be accomplished: N/A

Comments: -

13.3: Is a Web site visitor alerted when new windows appear?

The term "new windows" can refer to two types of windows: new browser windows or pop-up windows. New browser windows are opened by using an HTML command, whereas pop-up windows are opened by using a script on the client side. In some cases, people who use browsers that do not support scripting on the client side or who voluntarily disable scripts in their browser do not have access to the information contained in pop-up windows. If a pop-up window opens automatically whenever a page is loaded, it makes the site difficult to read, especially for people who use assistive technologies for browsing. It is not recommended that new windows be opened, especially if the user is not notified.

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 10.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 10.1_HTML_01, 10.1_HTML_02, 10.1_HTML_03

Type of criterion: design

Can be accomplished: Yes

Comments: Occasionally within a WSS 3.0/MOSS 2007 site, clicking an element opens a new browser window. In such a case, additional information or warnings (through ALT text on image links and TITLE attributes on text links) are systematically provided for hyperlinks that will open a new window to help reduce confusion and improve the user experience. For example, the title attribute of a hyperlink that opens a new window contains (*New window*).

13.4: Is there an alternative to scripts for opening new windows?

If no alternative to pop-ups is provided, some users cannot access pop-ups (for example, users whose browsers do not support scripting or who voluntarily disable scripting in their browser).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: 10.1
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 6.2_HTML_03, 6.3_HTML_02

Type of criterion: design / development

Can be accomplished: Yes

Comments: See section 5.1.1 A “More accessible” Mode.

13.5: Is additional information available to describe files that can be downloaded from the Web site?

When a user downloads files, the following information should be provided: file format, file size, link to a conversion utility if required, link to software needed to read the file format if required, and e-mail address for requesting a file in an alternate format. Files that are available for download on the Web site are not always in an accessible format. Therefore, it is important to provide the necessary means to access the downloadable files. If a file is in PDF format, for example, the user should be provided with the opportunity to download Acrobat Reader software (or a URL for an online conversion utility).

WCAG 1.0 checkpoint(s) [Priority 1]: -
WCAG 1.0 checkpoint(s) [Priority 2]: -

WCAG 1.0 checkpoint(s) [Priority 3]: 11.3

UWEM 1.0 Test(s):

Type of criterion: development

Can be accomplished:* *Yes

Comments: File size is not available. See also 1.6 in section 5.1.2 Graphics.

13.6: Does the specific presentation or layout of information interfere with the ability to access its content?

Care must be taken to ensure that specific layout or presentation elements (blinking text, scrolling text, and so forth) do not interfere with the user's ability to access the information: for example, fast blinking text (difficult for most users to read and even more difficult for people with visual disabilities) or fast scrolling information (difficult for people with visual disabilities or for people who have reading difficulties). If this type of presentation or layout is generated by an applet, a FLASH animation, or a GIF file, an alternative should be made available.

WCAG 1.0 checkpoint(s) [Priority 1]: 7.1
WCAG 1.0 checkpoint(s) [Priority 2]: 7.2, 7.3
WCAG 1.0 checkpoint(s) [Priority 3]: -

UWEM 1.0 Test(s): 7.1_HTML_01, 7.1_HTML_02, 7.1_HTML_03, 7.1_CSS_01, 7.1_external_01, 7.1_external_02, 7.3_HTML_01, 7.3_HTML_02, 7.3_external_01, 7.3_external_02

Type of criterion: design

Can be accomplished:* *Yes

Comments: WSS 3.0 and MOSS 2007 can be used to create templates (see 10.1 in section 5.1.11 Information layout) and Web pages that satisfy this criterion.

It remains the responsibility of MOSS 2007 developers to ensure that their Web pages satisfy this criterion.

Furthermore, as described in section 5.1.1 A “More accessible” Mode, the More Accessible Mode does not limit functionality, but instead enables alternate rendering methods for page elements so that they are compatible with assistive technologies (see eponym section 1.2.2 Assistive technologies).

5.1.15 Summary

As can be seen by the preceding discussion, MOSS 2007 technologies are capable of producing Web sites that comply with WCAG 1.0 checkpoints in all three priority levels.

This said, throughout the lifecycle of a MOSS 2007 technologies-based project, the design considerations identified in section 2.1 “Design checklist for accessible Web content” as well as the ones described in section 2.2 “Coding checklist for accessible Web content” should be observed.

5.2 Assessment and validation of a MOSS 2007 Web site for WCAG 1.0 checkpoints

The assessment and evaluation of a Web site content based on the MOSS 2007 technologies follow the overall approach and related methods developed under section 2.3 "Assessment and validation of accessible Web Content".

As indicated, such a validation should begin very early in the project, right from the design phase.

6. Conclusion

This document introduces the concept of eAccessibility and describes the issues involved, as well as the nature of the problems that require solutions. It provides an overview of initiatives, norms, and standards that apply to Web applications if we are to provide better accessibility to a broad public.

This document examines Microsoft products and technologies with respect to the design and production of accessible Web sites, analyzing the capabilities of these technologies to comply with WCAG 1.0 checkpoints of all three priority levels. It is intended to provide reference guidelines for producing accessible Web sites that are based on the new Microsoft Office SharePoint Server (MOSS) 2007 or previous solutions such as Content Management Server 2002 and SharePoint 2003 technologies. We encourage a systematic approach to Web site accessibility throughout the entire project lifecycle, supported by an appropriate testability plan.

The major guiding principles described here for the design and production phases in Web site creation, as well as the technical issues involved, are expected to remain stable when the new international WCAG 2.0 guidelines are released, which is anticipated sometime in 2007. The Internet accessibility guidelines practiced by European administrations will also evolve at that time, as indicated in the introduction, to converge towards the new international standards and criteria for success.

Although the Microsoft products and technologies addressed in this document are able to meet the requirements for providing accessible Web sites today, these products and technologies continuously evolve providing new features that support eAccessibility, in accordance with Microsoft's commitment to eInclusion.

Appendix A: WCAG 1.0 checkpoints

This appendix reproduces the Checklist of Checkpoints for Web Content Accessibility Guidelines 1.0. If includes all checkpoints for Priority 1, 2, and 3.

Priority 1 checkpoints

In General (Priority 1)	Yes	No	N/A
1.1 Provide a text equivalent for every non-text element (e.g., via "alt", "Longdesc", or in element content). This includes: images, graphical representations of text (including symbols), image map regions, animations (e.g., animated GIFs), applets and programmatic objects, ascii art, frames, scripts, images used as list bullets, spacers, graphical buttons, sounds (played with or without user interaction), stand-alone audio files, audio tracks of video, and video.			
2.1 Ensure that all information conveyed with color is also available without color, for example from context or markup.			
4.1 Clearly identify changes in the natural language of a document's text and any text equivalents (e.g., captions).			
6.1 Organize documents so they may be read without style sheets. For example, when an HTML document is rendered without associated style sheets, it must still be possible to read the document.			
6.2 Ensure that equivalents for dynamic content are updated when the dynamic content changes.			
7.1 Until user agents allow users to control flickering, avoid causing the screen to flicker.			
14.1 Use the clearest and simplest language appropriate for a site's content.			

And if you use images and image maps (Priority 1)	Yes	No	N/A
1.2 Provide redundant text links for each active region of a server-side image map.			
9.1 Provide client-side image maps instead of server-side image maps except where the regions cannot be defined with an available geometric shape.			

And if you use tables (Priority 1)	Yes	No	N/A
5.1 For data tables, identify row and column headers.			
5.2 For data tables that have two or more logical levels of row or column headers, use markup to associate data cells and header cells.			

And if you use frames (Priority 1)	Yes	No	N/A
12.1 Title each frame to facilitate frame identification and navigation.			

And if you use applets and scripts (Priority 1)	Yes	No	N/A
6.3 Ensure that pages are usable when scripts, applets, or other programmatic objects are turned off or not supported. If this is not possible, provide equivalent information on an alternative accessible page.			

And if you use multimedia (Priority 1)	Yes	No	N/A

1.3 Until user agents can automatically read aloud the text equivalent of a visual track, provide an auditory description of the important information of the visual track of a multimedia presentation.			
1.4 For any time-based multimedia presentation (e.g., a movie or animation), synchronize equivalent alternatives (e.g., captions or auditory descriptions of the visual track) with the presentation.			

And if all else fails (Priority 1)	Yes	No	N/A
11.4 If, after best efforts, you cannot create an accessible page, provide a link to an alternative page that uses W3C technologies, is accessible, has equivalent information (or functionality), and is updated as often as the inaccessible (original) page.Priority			

Priority 2 checkpoints

In General (Priority 2)	Yes	No	N/A
2.2 Ensure that foreground and background color combinations provide sufficient contrast when viewed by someone having color deficits or when viewed on a black and white screen. [Priority 2 for images, Priority 3 for text].			
3.1 When an appropriate markup language exists, use markup rather than images to convey information.			
3.2 Create documents that validate to published formal grammars.			
3.3 Use style sheets to control layout and presentation.			
3.4 Use relative rather than absolute units in markup language attribute values and style sheet property values.			
3.5 Use header elements to convey document structure and use them according to specification.			
3.6 Mark up lists and list items properly.			
3.7 Mark up quotations. Do not use quotation markup for formatting effects such as indentation.			
6.5 Ensure that dynamic content is accessible or provide an alternative presentation or page.			
7.2 Until user agents allow users to control blinking, avoid causing content to blink (i.e., change presentation at a regular rate, such as turning on and off).			
7.4 Until user agents provide the ability to stop the refresh, do not create periodically auto-refreshing pages.			
7.5 Until user agents provide the ability to stop auto-redirect, do not use markup to redirect pages automatically. Instead, configure the server to perform redirects.			
10.1 Until user agents allow users to turn off spawned windows, do not cause pop-ups or other windows to appear and do not change the current window without informing the user.			
11.1 Use W3C technologies when they are available and appropriate for a task and use the latest versions when supported.			
11.2 Avoid deprecated features of W3C technologies.			
12.3 Divide large blocks of information into more manageable groups where natural and appropriate.			
13.1 Clearly identify the target of each link.			
13.2 Provide metadata to add semantic information to pages and sites.			
13.3 Provide information about the general layout of a site (e.g., a site map or table of			

contents).

13.4 Use navigation mechanisms in a consistent manner.

And if you use tables (Priority 2)	Yes	No	N/A
5.3 Do not use tables for layout unless the table makes sense when linearized. Otherwise, if the table does not make sense, provide an alternative equivalent (which may be a linearized version).			
5.4 If a table is used for layout, do not use any structural markup for the purpose of visual formatting.			

And if you use frames (Priority 2)	Yes	No	N/A
12.2 Describe the purpose of frames and how frames relate to each other if it is not obvious by frame titles alone.			

And if you use forms (Priority 2)	Yes	No	N/A
10.2 Until user agents support explicit associations between labels and form controls, for all form controls with implicitly associated labels, ensure that the label is properly positioned.			
12.4 Associate labels explicitly with their controls.			

And if you use applets and scripts (Priority 2)	Yes	No	N/A
6.4 For scripts and applets, ensure that event handlers are input device-independent.			
7.3 Until user agents allow users to freeze moving content, avoid movement in pages.			
8.1 Make programmatic elements such as scripts and applets directly accessible or compatible with assistive technologies [Priority 1 if functionality is important and not presented elsewhere, otherwise Priority 2.]			
9.2 Ensure that any element that has its own interface can be operated in a device-independent manner.			
9.3 For scripts, specify logical event handlers rather than device-dependent event handlers.			

Priority 3 checkpoints

In General (Priority 3)	Yes	No	N/A

4.2 Specify the expansion of each abbreviation or acronym in a document where it first occurs.			
4.3 Identify the primary natural language of a document.			
9.4 Create a logical tab order through links, form controls, and objects.			
9.5 Provide keyboard shortcuts to important links (including those in client-side image maps), form controls, and groups of form controls.			
10.5 Until user agents (including assistive technologies) render adjacent links distinctly, include non-link, printable characters (surrounded by spaces) between adjacent links.			
11.3 Provide information so that users may receive documents according to their preferences (e.g., language, content type, etc.)			
13.5 Provide navigation bars to highlight and give access to the navigation mechanism.			
13.6 Group related links, identify the group (for user agents), and, until user agents do so, provide a way to bypass the group.			
13.7 If search functions are provided, enable different types of searches for different skill levels and preferences.			
13.8 Place distinguishing information at the beginning of headings, paragraphs, lists, etc.			
13.9 Provide information about document collections (i.e., documents comprising multiple pages.).			
13.10 Provide a means to skip over multi-line ASCII art.			
14.2 Supplement text with graphic or auditory presentations where they will facilitate comprehension of the page.			
14.3 Create a style of presentation that is consistent across pages.			
And if you use images and image maps (Priority 3)	**Yes**	**No**	**N/A**
1.5 Until user agents render text equivalents for client-side image map links, provide redundant text links for each active region of a client-side image map.			
And if you use tables (Priority 3)	**Yes**	**No**	**N/A**
5.5 Provide summaries for tables.			
5.6 Provide abbreviations for header labels.			
10.3 Until user agents (including assistive technologies) render side-by-side text correctly, provide a linear text alternative (on the current page or some other) for all tables that lay out text in parallel, word-wrapped columns.			
And if you use forms (Priority 3)	**Yes**	**No**	**N/A**
10.4 Until user agents handle empty controls correctly, include default, place-holding characters in edit boxes and text areas.			