Microsoft Web Accessibility Handbook

Index

Five Views on Web Accessibility and this Handbook

"Accessibility should not be an 'afterthought' for web services. Organizations need to develop and communicate their accessibility strategy with senior stakeholders then provide tools enabling developers to automatically deliver fully accessible services. Only this approach coupled with robust testing - both with real users and from a technical perspective - and a rigorous governance process ensure that an organization not only complies with the legislation but also meets the needs of all its customers".

John Harrison, Director Online Services, Her Majesty'sRevenue and Customs

"The Web Accessibility dialogues and this handbook provide an opportunity for organizations like Lloyds TSB to share best practices and real life solutions so that effective strategies can be understood and adopted more broadly. Evidence suggests a piecemeal approach to accessibility is unlikely to create repeatable, scalable and sustainable processes that ensure adherence to accessibility standards and best practice. Organizations should instead look beyond 'off the shelf' guidelines by adopting a formal and holistic approach which will result in accessibility becoming an ingrained part of their IT processes and practices, not an afterthought, nice to have or an optional extra."

Graeme Whippy, Senior Manager, IT Accessibility, Lloyds TSB

"Whether you are in search of guidance and practical hints to tackle your web accessibility issues or you just need to keep your eInclusion actions and policies well alive, this handbook is indeed a valuable source of knowledge and inspiration."

Steven Sintini, eAccessibility Advisor, CNIPA, Italy

"This handbook is a valuable resource with a strong practical focus to accessibility. It highlights the importance of disabled user testing to complement the automated and manual auditing process - so vital when websites are becoming web applications and user interaction ever more complex."

Robin Christopherson, Head of Accessibility Services, AbilityNet

"Organizations of all sizes and types can take positive steps towards more inclusive and accessible websites. Resources like this handbook, documenting proven strategies and existing solutions, are a great resource as we all strengthen our commitment to web accessibility."

Michael Takemura, Director, Accessibility & Aging Program Office, HP

Chapter 1
Introduction

Microsoft and HiSoftware believe there is now the opportunity and the imperative to make real progress toward a more inclusive society through an accessible Web.

The Riga Ministerial Declaration of June 11, 2006 emphasizes that "only 3% of public Web sites surveyed comply with the minimum Web accessibility standards and guidelines, hindering access to Web content and services for people with disabilities who comprise some 15% of the EU population." The declaration sets an ambitious policy goal: 100% of public Web sites to be accessible by 2010.

The tools and strategies to address this problem exist which means it is possible to make significant and rapid improvement with reasonable effort and expense. Web accessibility can be an area of sincere focus for organizations of all types and sizes *today*. Education and training represent some of the biggest obstacles for public agencies and private sector enterprises to create accessible Web sites – simply knowing what to do and how to do it.

Recognizing that the disappointing statistics show that we are not where we would like to be with Web accessibility and knowing that the tools exist to make progress, we asked ourselves, "What can be done?" We decided to conduct a series of Web accessibility dialogues. Not simply a single day or an ad hoc event, but a process that would lead us from Lisbon to Vienna to the next e-Inclusion Conference and beyond—to real, measurable progress.

Throughout 2008, Microsoft gathered government officials, industry leaders, and non-governmental organization (NGO) representatives across several European countries as part of a series of interactive dialogues on the current state of Web accessibility. The objective of the Dialogues was to explore how critical players in both public organizations and private industry can work together on solutions that create a more accessible World Wide Web. The Dialogues featured robust conversation that both framed the challenges faced by proponents of Web accessibility and illuminated various solutions that organizations are using to realize the shared goal of an accessible Web environment.

This Handbook from Microsoft and HiSoftware is one result of the European Dialogues on Web Accessibility. It combines practical strategies with some of the perspectives, goals, and vision that came out of those Dialogues. Our intent is to provide a tool that will help organizations worldwide foster accessibility in their operations.

The handbook begins with a summary of the important knowledge shared at the series of European Dialogues on Web Accessibility. Those lessons-learned and best practices are followed by an example process for implementing a comprehensive Web accessibility program in any organization. Building on that framework, there is an example of a product and community that helps organizations implement some of the practical strategies from the Dialogues. Finally, the Handbook offers some specific tools and tips to understand and meet Web accessibility standards.

Chapter 2
Summary of European Dialogues on Web Accessibility - Practical Strategies for an Accessible Web

For Microsoft and HiSoftware, the purpose of the European Dialogues on Web Accessibility was very specifically to create a body of knowledge that can be reused to improve the state of Web accessibility across Europe and worldwide. Our approach was to bring together people with in-depth Web accessibility expertise but with different perspectives. We intentionally included representatives from both the public and private sectors and from non-governmental organizations (NGOs). We designed the Dialogues to be very interactive in order to uncover and share the greatest amount of knowledge and experience.

This summary is an attempt to reflect many of the insights, questions, and solutions that were uncovered and explored during the Dialogues. Like the Dialogues, the summary focuses on three important topics: the business imperative for broader Web accessibility, challenges for Web accessibility, and strategies for an accessible Web.

Business Imperatives for Web Accessibility

There is real economic opportunity with accessible Web sites. One of the most important outcomes of the Dialogues was the identification of several compelling reasons for why an accessible Web makes good business sense. Some of those reasons include:

Market Benefits

- The number one reason for making a Web site accessible is to allow people to access your services and your site effectively and easily. It is not just to meet a series of checkpoints.
- The estimated overall economic opportunity associated with Web accessibility, as determined by the Directorate-General for Information Society and Media, is €85 billion over the next five years.
- Across the European Union (EU), improving accessibility for the elderly population means improving access to holders of more than €3 trillion, or approximately one-third of the European economy.
- Half of those elderly Europeans are not only willing to spend their money on a better quality of life, but are called 'hedonistic': they *want* to spend their money on a better quality of life. Often they cannot find products and services that meet their needs and abilities.
- Accessible content opens doors to more consumers. In many cases, Web content built with accessibility in mind is more popular with audiences than content that does not consider accessibility.
- Accessible content helps recruit loyal customers. When customers can work with an organization's online presence, they can better complete purchases and improve conversion rates. Customers will return to those sites that provide them with a positive experience.
- In many European markets there has been an almost explosive growth in internet shopping. It is important that consumers can find the button that lets them buy something. As many consumers as possible need to get there in order to increase sales.
- This is not a minority issue. Microsoft Research shows that more than 60% of internet users would benefit from using an accessible Web page, even if they do not have any disabilities. Accessibility is a majority issue. Everyone gains; it is a win-win proposition.
- Many people are affected by Web accessibility. In addition to people with disabilities, people who are elderly, people who are accessing the Web from mobile phones and people who are accessing it in a second language all benefit from accessible content.

- There are many excellent examples where Web accessibility has been a challenge, been addressed, and led to a direct business benefit. One example is a large, international grocery store chain. They had created a Web site for online shopping that looked fantastic but inaccessible. They used additional resources to create a second, accessible site. It generated significant revenue in the first year, because everyone liked using it. The company has now moved to a single site solution – the accessible one.
- Another example is a large financial services company based in the UK. It had an issue with its portal and was not seeing sales conversions on products, such as life insurance policies. There were complaints about the accessibility and usability of the site. It did a complete revamp of the site and found that, overnight, there was a 90% increase in sales of life insurance policies. The increase was due to improvement of the entire user experience but accessibility was a fundamental component.

Business and Technical Benefits

- Accessibility is "the gift that keeps giving." Benefits from Web accessibility go well beyond legal compliance and risk. For an organization there can be usability and productivity benefits, cost reductions, technical infrastructure improvements, wider markets, increased staff retention and better talent management.
- For many organizations, including governmental bodies, financial services companies, and so on, conducting business online is considerably cheaper than conducting business offline.
- Adopting an accessibility strategy early on enables an organization to implement it on its own schedule. Retrofitting sites after they launch is considerably more expensive that developing accessible content from scratch. Many organizations are already faced with legislation that requires that they provide accessible content, and the threat of expanding legislation is very real for a number of industries. Companies that engage an accessibility strategy soon can get ahead of legislation, as well as develop and implement content on their own terms and budget before it is mandated and before losing customers.
- Adopting an accessibility strategy puts an organization in a leadership position. Fostering accessibility is the right thing to do, and it's not prohibitively expensive. Taking an early stand allows an organization to position itself as a leader in its sector and present excellent opportunities for positive press and free marketing.
- If Web pages are more accessible, they will receive more hits and will move higher up in search engine results. Search engines make ranking decisions based on the same information that assistive technologies use to create an accessible user experience, including document structure and alternative text.
- The technical benefits of an accessible Web site are numerous. An accessible Web site is easier to translate into a different language, easier to internationalize, and easier to render on a mobile phone. Maintenance costs can be reduced and better search engine optimization can be achieved—driving more people to the site. This is in addition to increasing conversion rates and getting a better return on the investment by reducing running costs.

Workplace Benefits

- Organizations want to attract the best talent. In the UK, 6% of honors graduates are students with disabilities but 85% of online recruitment systems are inaccessible to them.
- In the UK, 78% of people with disabilities became disabled during their working lives. Accessible Web sites can help them keep their jobs.
- Employees with disabilities are often the most loyal. Whereas the average employee might change jobs every two to three years, employees with disabilities tend to stay in jobs considerably.
- Much of the power of corporate intranets is contingent on every single employee being able to perform tasks. Using the intranet as a communication tool broadly improves engagement and responsiveness within an organization.

- It makes no sense to move employees to an intranet and then produce a version in Braille. With accessible intranets, printing costs are reduced.
- Many organizations report that staff members with disabilities are highly motivated contributors. If an organization can use Web and intranet accessibility to attract and retain them, it would be a clear benefit.

Web Accessibility Challenges

Participants in the Dialogues identified and explained a complex set of challenges that organizations face when trying to promote Web accessibility. Those challenges include:

Benchmarking

- There is agreement that Web accessibility measurement and benchmarking are important for different reasons: for goal setting, for measuring progress, and for proving a business case for Web accessibility. However, there seem to be numerous metrics for Web accessibility. What is the preferred measurement? Is it sensible to first agree on what Web accessibility is and then to define a common measurement or just measure it according to what you are currently doing without asking how you arrive at a given number?
- After a measurement or benchmarking tool is selected, the challenge is to understand what the analysis means.
- How much compliance is enough? If you have one issue on a Web page, you fail an automated test. What is the threshold? Do you need 100% compliance on one set of pages or tasks and 80% on another? Do you need 90% across the board? Where is that threshold and how do you prioritize against it?
- Because of the dynamic nature of Web sites, they can be accessible one day and two weeks later, after the content has changed, they are not as accessible as they were. It is important to be able to monitor a Web site and its content to ensure that accessibility is maintained. Organizations need to start accessible and remain accessible. It is pointless to have an accessible Web site on day one only.
- In some countries there are regular evaluations of Web accessibility. Often there is little progress and sometimes there is even a decline.
- There is no benchmarking system across European member states to make comparisons across countries and/or to enable sharing of best practices.

Policy Challenges

- Legislation helps promote Web accessibility but it is not sufficient. There are limits to the effectiveness of public policy. Often there is no budget or financial resources to accompany Web accessibility requirements.
- Most countries in Europe have legislation and government guidelines on Web accessibility. While those countries with legislation are doing slightly better, across Europe the rate of Web accessibility is still not that good.
- Multiple policy approaches implemented across Europe to legislate e-Inclusion have led to a mixed array of regulations and, in some cases, further complicate attempts to address the issue.
- Countries with guidelines are pushing the accessibility agenda forward but without support from government, it will not be effective.
- Even with a very strong legal framework there are no ways to control how that legislation is being applied.

Content Challenges

- The Web represents millions of sites, many of which are created by end users and content creators who are not trained software developers. Expecting programmatic accessibility fixes to be made by these end users seems unrealistic.

- Even if you have a Content Management System (CMS) that allows you to build accessible Web site templates and an accessible framework, if your content creators know nothing about accessibility then you will not have an accessible Web site. The platform and the content creators have to work together.
- Even in places where content creators want to do the right thing, it can be so complex that they make mistakes. The result can be implementations that somewhat meet a checklist but they are completely different from each other. All have different accessibility issues and different usability levels.
- An organization can have a governance process that requires a project team to comply with standards, but user-generated content presents additional challenges. Many organizations encourage knowledge sharing through blogs and wikis. While this empowers individual business users, how do you require them to generate accessible content? It is impossible to police it.

Awareness Challenges

- A major challenge in many organizations, especially initially, is to raise the profile of Web accessibility. The challenge is multi-pronged and includes lack of awareness, lack of understanding and lack of investment of key stakeholders who can effect positive change.
- In general, there is a lack of common understanding about accessibility. Without a common understanding how do we measure accessibility across Europe? How do we find the best cases and practices? How do we find a common understanding about strategy?
- At an organizational level, not everyone is aware of Web accessibility and related problems. There is no culture of accessibility among public managers or technology officers. They do not have a technical understanding of Web accessibility.
- There are a growing number of Web developers who have been trained in usability and accessibility and have a good understanding of how to implement and test for accessibility. However, most people either do not know about accessibility or have misconceptions about how or why to implement it.
- There may be a lack of "self-identification" within some organizations. For example, when told that their intranet is not accessible, some Intranet managers say 'but we do not employ any disabled people'. This is a cyclical issue: If an organization does not have an accessible intranet, people with disabilities cannot enter the front door. An organization is usually not aware that this has serious repercussions with regard to employment and equality.
- Organizations looking to implement an accessibility strategy need to be aware of and account for the diverse needs people with disabilities bring to the Web, including the fact that some people have multiple disabilities and require a variety of strategies all at once. People with visual disabilities have a very different set of needs than those with a dexterity disability, who have different needs than someone with dyslexia, and so on. Also, as the online population ages, more individuals will bring combinations of these issues with them to the Web, expanding the need for accessibility considerations in both scope and complexity.
- There is a wealth of information available about Web accessibility, including many different sets of guidelines. This wide range of information can create its own challenges. Which guidelines should an organization use? On which expert should an organization rely? There is almost too much information and it can be difficult to make use of it. It sometimes forces organizations to leap from theory with industry guidelines directly into the practice of implementing those guidelines. There is a gap between theory and practice.

Innovation Challenges

- With technology and the Web there is a rapid development cycle and a push to be more and more innovative. When something new or interesting comes out, accessibility must be infused from the very beginning so that the result is accessible or there must be an accessible way of representing it.
- The number of government agencies providing eGovernment services is growing. Often these are Web 2.0-type activities, such as collecting taxes, paying parking tickets, online education, etc. These

Web 2.0-type applications are highly interactive and highly complex. When you look at WCAG 1.0, written in 1999, you cannot apply it to the kinds of services being delivered today.

- When talking about innovation and accessibility there is the misperception that accessibility is playing to the lowest common denominator and about making things boring. The reality is that accessibility is a challenge that, when relished, is an opportunity to create innovative user interfaces.

Stakeholder Challenges

- Organizations can be complex. Owners of particular products, services or information often own everything about them, including the online presence. Engaging with these stakeholders is critical but difficult.
- Stakeholders often practice 'Grudging accessibility' or just going through a checklist because someone has required it. This does not help usability and inclusivity. People need to be focusing on inclusiveness in order to reap the benefits.
- Often when stakeholders are told their Web sites or intranets are inaccessible the reaction is consistent; it is like the five stages of grieving, or rather the five stages of accessibility acceptance:
  - Denial: 'It cannot be important. Where will we get the funding?'
  - Anger: often stakeholders are very angry to hear they are inadequate in this area.
  - Bargaining: stakeholders work out how to get around accessibility and how to cope with it without really having to start again
  - Depression: often happens when stakeholders start looking at WCAG 1.0.
  - Finally, after stakeholders work out what needs to be done and have carried out some research, they reach acceptance. This can take years for some stakeholders.
- It can be difficult to develop a consistent or sustainable approach to Web accessibility across an organization, so when an employee/evangelist with expertise moves on to a new job or a new role, they take their experience and knowledge with them and it dissipates.
- Accessibility typically is not a budgeted line item for most organizations. It is, unfortunately, usually a competing interest. Given the way that technology works, it is one of many things that organizations have to weigh in terms of whether it will make the list of things that they are going to do.
- When stakeholders are building a new Web site or intranet and there are an extensive number of competing requirements for functions and features, accessible design may be on the list but may be one of many competing interests. When stakeholders do not understand accessibility, it is likely to fall to the bottom of such lists.

Practical Strategies for an Accessible Web

By design, the European Dialogues on Web Accessibility focused on exploring solutions and approaches to address challenges and promote a higher rate of Web accessibility. The experts on the Dialogue panels and in the audience identified a long list of practical strategies, including:

Use Standards Smartly

- Technical specifications and guidelines are a cornerstone of an effective Web accessibility strategy.
- It is important to recognize the emergence of new technologies and to utilize up to date standards. For example WCAG 1.0 is not designed or applicable to building a Web application that is interactive and involves menus, dialogs, scripting, and back and forth conversation between the end user and the server. WCAG 2.0 was designed to address some of these issues and to be applicable to new technologies as they emerge.
- However, even WCAG 2.0 will never be the singular solution to making Web sites accessible. Web sites change continually so the W3C Authoring Tool Accessibility Guidelines (ATAG) can be an important standards tool. Even though WCAG defines what it means to be an accessible Web site, ATAG ensures that, over the long term, the production of Web content will be accessible.

- While standards can give a solid foundation for accessible Web sites, organizations should not expect to use them off the shelf. They will likely need to be interpreted for each organization.
- Make ownership of accessibility standards as broad as possible within an organization, so that they do not become just the opinion or preference of one person.

Raise Awareness and Knowledge

- People do not set out to create inaccessible Web sites – so education is key. Give people the awareness, the desire and the technical skills to deliver accessible IT solutions. If they are not aware of it, if it is outside their frame of reference, if they do not have the skills to deliver it, they will not create accessible Web sites.
- Training software professionals—the people who create development environments and authoring tools— should be a top priority. Often, content creators are not the appropriate people to fix accessibility problems.
- Education and training is critical not only for developers, but also for designers, editors and even management. Knowledge directly affects Web site creation and impacts the creation of goals and strategies that support Web accessibility.
- Raising awareness among business unit owners and sponsors helps them to understand why accessibility is important, and then they may even start demanding it from their IT professionals, rather than having it forced upon them. High-level awareness presentations can talk about what accessibility is, why it is important, and what the benefits are.
- Cross-training and train the trainer approaches can be very effective in organizations of all sizes. Best practices can be extended throughout an organization by identifying individuals and teams that are getting it right and building accessible Web sites. Invite them to showcase their work and to share lessons learned about what works well.
- Training should focus on the rationale behind guidelines –strategies for creating an accessible user experience. The point is not to tick every single compliance box just for the sake of it. Education and training should result in a Web site that is usable by everyone.
- The Web 2.0 world of user-generated content makes accessibility education and training even more important. Many organizations of all types are encouraging people to share their knowledge in blogs and wikis. Strategies to teach people to generate content in an accessible way include articles on an intranet, road shows, and training sessions.

Institutionalize Accessibility

- Within an organization of any size, moving from a guerrilla campaign for an accessible Web to a more institutionalized approach can be difficult.
- Creating an "Accessibility Center of Excellence" or "Web Standards Forum" within an organization can help to define accessibility standards and practices and to drive broad implementation.
- A Web accessibility group should include representatives from across an organization, including influential decision-makers and senior people, so they can help achieve buy in and to raise the profile of accessibility.
- A Web accessibility group can bring draft strategies to decision-makers and seek the real authority to drive a strategic approach to accessibility.
- Horizontal networking – working together in an organization (and externally) with other webmasters and content owners that face similar challenges--can help solve everyday problems.

Test and Benchmark Appropriately

- Many organizations have hundreds of Web sites and millions of Web pages so automatic tools are needed to help identify potential issues.
- The key to automation is repeatability. A repeatable process is important for measuring progress over time.

- Often, a small number of Web sites generate the most traffic. Focusing accessibility testing and remediation on those key sites can enable more rapid progress in the area of accessible Web sites. Automatic scanning tools can allow for quickly selecting any kind of criteria to check against.
- Understanding and interpreting the results of automated tools is important. The reports can sometimes be misleading. For example, if a search box on every page of a Web site is missing a label or not properly tagged, then it could cause 100% failure across all pages. One quick fix could bring several pages into compliance.
- The challenge with any kind of benchmarking, auditing or testing is that Web content is dynamic by nature – it changes all the time. Even if a Web site is accessible one minute, it can change 100 times in the next hour as thousands of content creators add new content.
- However, testing for accessibility is not only an automatic process.   It is important to recognize that there are three additional types of testing that are appropriate in different situations and for different purposes that can lead to even more useful information: functional testing, usability testing and compliance testing. *Note: The last chapter in this guide describes these three types of testing in more detail. It also discusses the importance of sequencing them in the right order to achieve the best results.*
- Compliance testing focuses on adherence to technical specifications. It answers the question, "Does a Web site comply with a standard?"
- Functional testing is about whether the product works as it is designed, and whether the design is a reasonably functional design. Does it actually incorporate accessibility needs? Was it designed to work with a mouse, to read out loud in order, to use large fonts and so on? Functional testing prioritizes by user scenario, addressing task completion and the importance of that particular task.
- Usability testing is about whether actual people can use the product. It involves people using a Web site with their relevant software and assistive technologies. This testing can provide broad and deep insights into additional issues that guidelines will never cover.
- User error is a factor with usability testing. A Web site might expose everything the way it is supposed to and meet relevant standards but users still may not be able to use it. That is a different question than whether it was designed well and does it meet a standard. Usability testing is best done last, after confirming that other key aspects work.
- Rigorously testing all templates used by Web teams and the gateway pages can help ensure that at least the main elements are accessible and can be used by people with assistive technologies to create Web pages.

Make It Easy

- If it is easy to do the right thing, people are much more likely to do it. If you create accessible content by default and someone has to do something to change it, they are less likely to do so. If you create content that is not accessible and if you have to then do something additional, it is that much more of a barrier to Web accessibility.
- Developers can design and develop software in such a way that end users do not have to think about making accessible web content; since it is built into the tools, they get the accessible solution by default.
- A standard software engineering practice is to take functionality that is duplicated in multiple places and to bring it in to a single control. Each of the different places can reference the single control, which avoids replicating the same work over and over again. Web accessibility examples include functions and libraries in coding environments or Comet controls in Windows. Web services such as third-party shopping carts are another example of something where specialized knowledge has been concentrated into a single place.
- Do not become a checkpoint service. Focus on real life issues. It is a myth that accessible Web sites do not look as nice or do not have as rich functionality as inaccessible Web sites. Web developers love the challenge of creating an accessible, engaging, attractive user experience, regardless of whether you can see the screen, understand the language, or use the mouse. It is all achievable; it just requires some innovative thinking. Stick to the guidelines as your bedrock and understand what they are trying to achieve.

- Many Web developers and content creators will never gain a complete knowledge of web accessibility. Automated tools can help fill in the knowledge gaps, but a comprehensive accessibility toolkit that includes manual inspection, tutorials, and other resources as in addition to automated checks will provide the range of support needed by developers.
- Do not expect industry guidelines, technical specifications, or checkpoints to be broadly usable off the shelf. They can be open to interpretation by many different Web developers and content creators. They can be an exceptional starting point to promote an understanding of accessibility and to provide a foundation for further work, but they are more useful when turned into a set of standards specific to an organization and its developers, suppliers, and customers.
- Combine additional best practices and pragmatism with industry guidelines and technical specifications. Once those standards are in place, adhere to them every time a Web site or intranet is developed. Test against them.
- A set of internal standards can be the operational way for developers to implement an organization's strategy. It can control the way that developers code pages, so that anything that produced during the development stage should meet internal standards.
- A lower level, completely organization-specific set of very detailed internal standards can provide an almost "paint by numbers" product for developers. For every element on a page it tells them how it will behave, the size, the branding elements to use, the color, the behavior of the content when it is clicked, etc. It can leave absolutely no room for interpretation. It can guarantee accessibility compliance and brand compliance.
- Content creators may require additional tools. Increasingly they are not technology professionals and have little or no accessibility expertise.

Require Compliance and Governance

- Do not rely on people's best efforts or goodwill to use internal standards, tools, practices, and strategies for Web accessibility. Put a governance process in place. Require compliance and penalize non-compliance.
- Governance provides a set of rails for people to run along. It gives needed guidance to do their jobs properly but it also gives the mechanism to identify non-compliance and a means of dealing with it.
- A governance process does two things at the outset: 1. it makes sure that internal standards are adhered to during the design and development process of Web sites and intranets; 2. it means that procurement documentation includes the right questions to help people purchase accessible solutions.
- A governance process must also effectively address non-compliance – perhaps through a waiver process. If a project does not meet an internal accessibility standard it is best to understand the issues and create a managed remediation plan.
- A waiver for non-compliance may require an assessment of risk and a time bound action plan for making adjustments and improvements as well as steps to minimize the impact on staff and customers. It could be signed off by very senior people within the organization.
- An effective governance process should make it easier for people to comply with internal standards than get through the waiver process.
- A governance process does not have to be hundreds of pages long; it should be a clear, concise policy document.

Approach Web Accessibility as a Process

- A full lifecycle approach to Web accessibility can be effective. The lifecycle begins with an accessible design and continues after the launch of a Web site with good solutions to maintain the compliance of the site.
- Incorporate accessibility as part of the build process, so that designers and developers, and even the tools themselves, enable accessibility to the greatest extent possible.

- Timing can be vital. With any large Web project, there is a window of a few months in which people are planning what will happen over the longer term. Missing that window means probably means missing an accessibility feature in the next release. It is important to really understand an organization and the points at which decisions can be influenced.
- A lifecycle approach can include several stages. A first stage would be an assessment of existing Web sites and benchmarking them against standards and guidelines along with a review of the technologies the site should support such as Flash, JavaScript, Cascading Style Sheets (CSS), HTML, etc. A second stage would be an examination of resources needed – both internal and external. A third stage would include defining a strategy, including a quality assurance and testing plan. Once all these plans are in place, a fourth stage would focus on design, prototyping and testing. A final stage would include feedback, refining code libraries, making sure style guides are up to date, and building in checks for accessibility at regular intervals.
- Within any process, focus on prioritization, both in testing and fixing problems, to ensure the most impact. When evaluating the functional aspects of accessibility and what may need to be fixed, consider the use of scenarios to determine which things are the most important for the user.

Evangelize and Engage

- Many Web accessibility projects fail because nobody has clear ownership of accessibility within the organization. There must be an accessibility project manager for the whole course of a project and beyond. Define who has ownership, who is driving it, and who is managing it.
- In addition to a project manager, choose an accessibility champion. It should be someone senior in the organization that takes responsibility for the accessibility strategy and, when it matters, is prepared to stand up and be counted. Decide how serious the organization is about accessibility. To what lengths is it prepared to go in order to enforce an accessibility standard?
- Determine the key stakeholders within an organization. Who has an interest in making a Web site accessible? This could include Web teams, procurement teams, legal teams, and marketing and branding teams, who quite often are forgotten until towards the end and may come up with a number of missing things.
- Define the key stakeholders and interview them. In large organizations there can be a lot of politics and emotion involved in how people want the Web site to be, so get it all out on the table. Figure it out. What are people's hopes and goals? What do they want to achieve? What are their fears?
- Engage individually with a variety of people and sell them the message in the way that is important to them. Different messages resonate with different people. The most successful seem to be a combination of "this is an interesting problem that you can solve which helps people and also makes money".
  - If it is a business person, then maybe the business opportunities of a growing aging demographic will resonate.
  - If it is a user interface designer, then focusing on interesting user interface challenges may make accessibility a more appealing topic.
  - If it is a developer, then discussions might focus on code quality, standards and implementation.
  - If it is a tester, then how accessibility impacts quality might be of interest.
- Senior level executive buy in to accessibility is absolutely essential. People at the grassroots level, the developers and the designers, are critical to delivering accessibility from the bottom up. However, executive buy in to mandate it from the top down is equally important. Much of the actual guidance and leadership comes from the middle.

Chapter 3
A Process Based Approach to Implementing a Web Accessibility Program

One of the practical strategies consistently identified and discussed at the European Dialogues on Web Accessibility is the importance and usefulness of defining a process for creating and maintaining accessible Web sites. This chapter will describe what an effective process based approach might look like.

Approaching Web content quality management as an afterthought can be a circular and costly strategy. Organizations that include accessible design and quality management as part of their development plans, quality assurance and testing practices will have a tremendous economic advantage. For example, according to a study by Forrester Research, the cost of retrofitting a Web site for accessibility can be as much as ten times the cost of designing an accessible site from the ground up.[1]

Web sites that contain accessibility issues, such as broken links, improperly tagged or missing image alternatives and slow loading pages create a negative experience for users. Any negative experience will adversely impact organizations that rely upon online communication to send their product message as well as businesses built upon online transactions.

Testing, remediation and monitoring solutions that easily integrate into current practices enable organizations to build standards based compliancy testing into their existing planning, development, quality assurance, and defect tracking processes. These solutions can empower content managers and executives to implement, remediate, and monitor compliance to regulatory and corporate Web standards. An integrated approach will also help organizations troubleshoot sources of quality issues and identify inaccessible Web content.

Three important components of a process based approach to accessibility include the following elements:

- **The Right Leadership** and buy in from key stakeholders in the organization;
- **Education** of all stakeholders on the current state of accessibility and creation of a plan of action and;
- **Empowerment** of everyone that is impacted throughout the design, development, quality assurance and monitoring process.

The Right Leadership
As with any project, putting the right leadership team in place is essential. An accessibility project team should include individuals from every group that contributes to the development and maintenance of a Web site. In the average organization the following groups would most likely be represented:

- **Marketing** – For their knowledge of who uses the site and for their knowledge of site usage statistics.
- **Design** – For their knowledge of the usability, interaction, and visual aspects of the site.
- **Development** – For their knowledge of what skills and technologies will be needed to complete the task.
- **Public relations** – To determine whether or not the accessibility effort should be made public.
- **End users** – From internal users , to usability testers, to an outside beta tester group, actual users will help to make any accessibility project successful. Accessibility is part of user interface, and users often have surprising interactions with any user interface.

---

[1]
  Source: December 2001 Forrester Research Report-Design Accessible Sites Now" available at: http://www.forrester.com/ER/Research/Report/Summary/0,1338,11431,00.html

- **Executive management** – Corporate or Executive representation on the team will help to keep the project funded and a top priority for the organization.
- **Operations** – This group is generally responsible for overall Web site performance and testing. This group will be critical in the development of a long term site accessibility strategy.
- **Professional development / training** – This group should be part of the project from the beginning so they can better understand training requirements.

Participation and support from key members of the organization can assist in providing the funding and effort to incorporate accessibility testing into the Web project life cycle. It is extremely important that an organization develop both a top down strategy, with commitment and executive sponsorship of the accessibility plan, as well as bottom up support from the Web architects and Web developers. Through this combined approach, accessibility can be seamlessly integrated into existing company practices for Web planning, development, testing and deployment.

Three Types of Testing

As Web sites become more complex and interactive, development of Web sites becomes more and more like software development. If we look to the processes of software testing, we will see that there other types of testing besides what is typically done for Web sites.

1. **Functional testing** – is the product working as it is designed? Does the design incorporate accessibility requirements?
2. **Usability testing** – Can people (with and without disabilities) complete tasks using the product?
3. **Compliance testing** –Does the Web site conform to guidelines and standards? This can be automated to a certain extent, particularly for sites that contain mostly documents and few interactive application features. Some aspects of compliance will require functional testing, and others will require human inspection of test results generated by automated tools.

Oftentimes, the three types of testing are combined or confused, or they are completed out of order. Completing them in the given order is important for achieving accurate results. For example, while usability testing is valuable early in the process, it can be blocked by functional bugs. So, it is best to engage in usability testing after you have identified and fixed functional issues. Automated compliance testing can also be blocked by functional problems, preventing the tools from reaching all parts of the site.

Involving users throughout the design and development process provides valuable feedback and there are a variety of methods to gather feedback from potential users: focus groups, usability testing on previous versions of the product, and testing with paper prototypes among others.

Automated testing solutions can be used to great effect in compliance testing a site or groups of documents. It is very important to remember that *no* tool alone can validate the absolute accessibility of a Web site. However a good software solution can identify a majority of what needs to be manually inspected. Additionally, a good solution will identify which pages do not need to be manually inspected, based on the absence of elements that require manual inspection. *Remember: A human will still need to assure that all manual checkpoints identified by the solution are accessible, and some aspects of compliance require functional testing.*

Therefore, all three types of testing are important and each has its place in the software development process. This becomes more important as Web sites become Web applications and software testing methodologies are

employed. Testing is no longer just about crawling the code looking for specific attributes, elements, and tree structures, it becomes a matter of ensuring that the application works as designed and that the interface is perceivable, operable, and understandable by end users.

Planning and Implementation

The following are the four basic phases of developing and maintaining an accessible Web site:

A. **Develop a plan** –Set organization goals, design guidelines and standards, and perform an assessment of your Web site or application. Understand that the requirements are different for application features than they are for documents. Many organizations choose to build policies around the World Wide Web Consortium (W3C) Web Accessibility Guidelines (WCAG) 2.0, which are intended to cover the ever-expanding set of Web technologies and to focus on techniques for implementation.

B. **Implement guidelines** – Train designers, developers and testers . Decide which features should be retrofitted and which should be re-created. Design new features to incorporate the principals of accessibility, based on an organization's new quality guidelines and standards, as it is much more expensive to add it on later. Retrofit existing documents, and put in place tools and processes to ensure that new documents are created in an accessible way.

C. **Accessibility testing** – Test that your product has met your standards and guidelines, and that it works as you expected. For Web applications, this will involve three types of testing: Functional, Usability and Compliance. For documents and Web sites made up mostly of documents, with few application features, Compliance testing will sometimes be enough.

D. **Accessibility maintenance** – Evaluate results. Automate verification of accessibility guidelines and standards for documents and static content. Add processes to ensure that new application features are tested for functionality, usability and compliance, before those features are released.

Empowerment

Provide tools to Web site designers and content creators to build accessible content

Giving team members the tools that they need to do the right thing from the beginning is the most cost effective approach to accessibility. Experience tells us that no one builds a Web site intending that it be inaccessible, but education, simplicity, and real empowerment are key factors to a successful strategy.

Implementing Organizational Guidelines

With the accessibility goals and guidelines developed, and a baseline audit completed, the next phase of the project begins. In this phase developers and quality assurance teams receive training and begin the process of retrofitting the company Web site. Organizations must first make an assessment of what is being done right and what is not being developed so that it is accessible.

Automate testing and understand where automation is not enough

An accessibility testing strategy will provide an organization with the ability to view accessibility from a project management perspective. This will enable an organization to:

- Understand what is an application feature and what is content, and have a plan for how to test each;
- Allocate resources appropriately;
- Track site progress;
- Educate employees;
- Identify problem areas;

- Integrate accessibility into product design, quality assurance, and content delivery processes;
- Keep a historical view of your Web site accessibility work.

Reward good behavior and set reasonable consequences if standards are not followed

Most employees strive to be successful in their jobs. Once a Web accessibility strategy has been implemented, recognize teams and team members who work hard to accomplish their goals. Allow them to lead by example. Often evangelists within an organization who embrace the concepts and methodology behind accessible development can serve as mentors and trainers for others.

This phased approach offers the best opportunity for Web site compliance to accessibility guidelines and standards. Companies can avoid becoming a target and look forward to the continued accessibility of their Web content once they develop the above disciplines. There are many good reasons for supporting the creation of accessible Web content. It is important to recognize that these changes will not happen overnight. Once it is clear that "developing accessible" is achievable, affordable, and can provide a competitive edge, it will become common rather than the exception.

Chapter 4
Demonstration of Practical Strategies from the Dialogues

In the previous chapter, we learned about the practical strategies for making Web content accessible that were discussed at the European Dialogues on Web Accessibility. These include using standards wisely, raising awareness within an organization, institutionalizing accessibility, testing and benchmarking web development for accessibility, and making it easy to make sites accessible. In this chapter we'll learn how the Accessibility Kit for SharePoint (AKS), and in particular the community that has grown with the product, is a real-life example of many of the practical strategies that can be used to make Web content accessible.

AKS: A Community-based Approach to Web Accessibility

Microsoft Office SharePoint Server 2007 (MOSS) is a business productivity server based solution that can improve organizational efficiency and collaboration through integrated content management and enterprise search, accelerating shared business processes, and facilitating information-sharing across boundaries for better business insight.

Microsoft Office SharePoint Server 2007 (MOSS) made a number of accessibility improvements and enhancements over its predecessor, SharePoint Portal Server 2003. The W3C Web Content Accessibility Guidelines (WCAG) 2.0 working drafts were considered as part of the design process. However, Microsoft received feedback from customers who had a need to create WCAG 1.0 AA conformant Web sites on the SharePoint platform. Based on analysis of that feedback, and the most urgent customer requirements, the SharePoint team determined that they would create an accessibility add-on for SharePoint, to address those requirements which became AKS 1.0. AKS 2.0 will address additional customer scenarios and will provide additional WCAG 2.0 support.

AKS provides an accessible layer on top of a SharePoint implementation that accelerates the process of accessible Web development. AKS supplements the templates, style sheets and the components that are already utilized in SharePoint to create the most common types of sites, enabling users to deploy accessible SharePoint sites quickly. It also provides a framework and documentation to help developers make more complex and customized sites accessible. The Community site and discussion infrastructure include peer and expert support as well as code-sharing. This provides the community with the ability to extend the tools that Microsoft and HiSoftware have provided.

Microsoft has provided this solution, in collaboration with HiSoftware, as open source and at no additional cost to its customers. The creation of an "accessibility community" for this dominant platform technology will educate, enhance and extend awareness and accessibility initiatives as a high priority for organizations using SharePoint as their business productivity platform.

AKS is intended to significantly reduce the time, knowledge, and effort required to implement a SharePoint-based Web site that conforms to the World Wide Web Consortium's (W3C) Web Content Accessibility Guidelines 1.0 Priority 1 and 2 checkpoints. SharePoint is the fastest growing server product in Microsoft's history with revenues in fiscal year 2007 of more than $800USD million worldwide and greater than 37% annual growth in the UK alone. Microsoft sold 84 million licenses of SharePoint worldwide. These licenses are for internal users of SharePoint, and do not even contemplate the hundreds of millions of additional Web users that

SharePoint may impact when it is used to create public facing Web sites. AKS allows the creation of WCAG 1.0-compliant SharePoint sites. This is extremely important because SharePoint has been deployed extensively throughout enterprise organizations worldwide, to create collaborative work spaces and Intranet sites, as well as Internet Sites. Microsoft's emphasis on accessibility across the SharePoint Platform will have a dramatic impact on improving accessibility of Web sites AND Intranet (employee) portals.

AKS is developed as a set of building blocks rather than an end-to-end solution. SharePoint users can utilize various pieces of the kit and integrate them into their respective product or service offerings while some customers will integrate particular components of the kit into their SharePoint deployment processes.

AKS is available-at no cost- via Microsoft Public License (Ms-PL) which was recently approved by the Open Source Initiative as an open source license. Additionally the kit utilizes an open and well documented interface to enable accessible extensions of SharePoint. Finally, AKS contains tutorials and education-empowering others to "build accessible". This education based approach to accessibility allows business users and content creators within the organization to do the "right thing" in terms of accessible development, more easily. By exposing content creators to more accessible tools and utilities and providing the explanation for "why" these tools are more accessible, users can learn, extend and incorporate these principals into their ongoing Web development efforts, both within and outside of SharePoint.

AKS Approach

There are many, many SharePoint-based Web sites and intranets in the world. AKS allows organizations to continue to use their existing SharePoint infrastructure to build accessible sites, and to incrementally improve their existing sites. AKS significantly reduces the amount of work needed to make these sites conform to WCAG 1.0. This puts a WCAG 1.0 compliant site within reach of many organizations that may not have been able to achieve it in the past, and as such, improves the overall accessibility of the Web.

In developing AKS, Microsoft and HiSoftware wanted to ensure that adopting it would be as painless as possible for organizations using SharePoint. To achieve that it had to be non-invasive, educational, and extensible.

- **Non-invasive** – AKS does not modify any core components of SharePoint, cannot break existing installations, and won't be broken by service releases. Developers can use as much or as little of AKS as they choose, and can apply it incrementally.
- **Educational** – By deploying open tools to assist in the deployment of more usable sites the kit is going beyond just telling the designer to do something. AKS provides tools are being provided to help designers achieve and understand Web accessibility.
- **Extensible** – By openly documenting and issuing guidance around Web parts, the AKS solution provides a simple framework example for all developers to use, allowing for the broader impact of the AKS across all developers who build SharePoint sites and add-ons for reuse.

This non-invasive, extensible, educational approach has made the barrier to entry for WCAG 1.0 conformant Web sites much lower.

AKS allows an organization to meet organizational goals and objectives with regard to compliance with standards and guidelines at a low cost, not only because it is a free add-on, but because it is also designed for a low total cost of ownership. It is non-invasive and won't break existing installations. It does not require training for end user. It allows WCAG 1.0 AA conformance on a site built with SharePoint 2007, and introduced accessible business productivity solutions.

End users reading sites benefit from AKS because it lowers the cost of accessibility improvements on the most common SharePoint sites, which means that more of these sites will be made accessible.

End Users as Authors

AKS and for that matter, SharePoint itself empowers end users to become site authors. The very purpose of SharePoint is to separate design from content. SharePoint uses templates and master pages with CSS to design the look and feel of a site, and then Web parts to feed rich content through those templates. End users choose from templates and add content through them. In this way users are empowered to build their own content through the creation of MySites, TeamSites and through direct publication of content on the Web which can be managed through a SharePoint work flow. This empowering process is extended with accessibility through the implementation of AKS. AKS applied on the server allows users to create content that is more accessible without specific design knowledge.

Since AKS is run on the server, end users do not need to install or configure anything. They don't need to understand the details of Web Accessibility or even Web site development. AKS provides, out of the box, WCAG 1.0 conformant versions of the most common SharePoint scenarios, including site templates and the most popular Web parts.

AKS empowers users to create more accessible output through SharePoint by allowing them to 1) start their site with an accessible template; 2) create accessible rich user experiences with accessible versions of the most common Web parts used in SharePoint 3) accomplish this without any additional burden on the content creator or end user, as AKS is installed entirely on the server side of SharePoint by an administrator. Business users and content creators are then more able to achieve accessible Web site design because they are provided with the tools to help them do the "right thing" without having to fully understand the details of doing so.

The following pieces of the solution empower end users engaged in authoring functions to create WCAG 1.0 AA conformant sites, without having to fully understand the intricacies of that standard or Web development.

The AKS Solution-Enabling Accessibility

The AKS solution represents a set of files, programs, and utilities that facilitate the enhanced accessible and usable output of Microsoft Office Server SharePoint (MOSS) 2007 and Windows SharePoint Services (WSS). While the kit is not an end-all to accessibility it represents both a step forward in technology and education. Throughout this project, HiSoftware's development team worked in collaboration with the Microsoft SharePoint Server development team to ensure that AKS was designed and developed in a way that would have little to no impact on the core system architecture of SharePoint. In this way, there would be little impact to AKS users when Microsoft released a Service Pack or update for SharePoint.

The AKS development effort also used an iterative approach with phased releases of the kit to address top accessibility priorities with gains that could be readily achieved. This was undertaken through the use of updates to Master Pages, Publishing Templates, CSS and Control Adapters that intercept the output of SharePoint Web parts before that output is rendered in a Web browser.

From a technology perspective it provides new Cascading Style Sheets (CSS), Master Pages, multiple control adapters and Reusable Content Examples as needed to be successful. One approach addressing accessibility challenges within SharePoint would have been to modify the Web parts within SharePoint itself. But HiSoftware

in collaboration with the SharePoint development team determined that this approach carried too great a risk of creating other issues that could be detrimental within the SharePoint environment.

The AKS Size Utility

The AKS Size Utility is a C# Application developed to help facilitate the change of units from absolute to relative. While there are many places on the internet where one can find these calculations, the AKS Size Utility provides a practical and automated solution. The utility does this by opening and documenting conversion figures that will be adjustable by individual designers and developers alike.

The AKS Style Sheets

The AKS Style sheets represent a group of style sheets that have been reviewed and tested to allow for the quick implementation of relative sizing for sites that use the styles shipped with SharePoint 2007 and for sites that will be implemented anew with SharePoint styles.

The AKS Master Pages

A master page in SharePoint is used to create a standard "look and feel" for a Web site. The master page controls the primary colors, navigation and "style" of the site. The master pages apply to the entire section of a site and create a consistent experience across the site. These pages can be created through the out of the box templates in SharePoint and can also be created by developers from scratch. The AKS master pages are a group of master pages that will implement the new styles and additional template related corrections that provide better usability overall. These pages will allow for a one click implementation of the AKS Style sheets and other accessibility and usability enhancements as required or wanted by the site designers. The image below is a typical SharePoint Web page. One of the accessibility challenges in SharePoint is its use of Absolute font sizes versus relative font size. Thus when an end user changes the font size in their browser, nothing happens.

AKS Reusable Content

Reusable content can provide SharePoint users with simple code examples that can be cut and pasted into their implementations, The AKS examples will provide a great way to both educate and deliver more usable and in some cases more accessible content. In addition re-usable content provides the ability for more uniform and accessible standardized content to the internet or intranet sites to which you are publishing.

IT Developers

AKS also provides tools for the developers and administrators in the organization who are responsible for more complex or customized SharePoint based sites. All of the features of AKS are available separately, so that developers can pick and choose the ones they need.

This enables developers to either use bits and pieces of the code in AKS that are applicable to their custom development, or to utilize the documented practices in AKS to enhance the accessibility of their own development efforts. In addition, the community of AKS developers offers both peer and expert support, and open source code sharing. This educational aspect of AKS is appealing to these developers, and will help to drive both adoption and quality of implementation.

One of the primary goals of the AKS project was to develop a set of building blocks, components, documentation and educational materials that accelerate the process of accessible development using SharePoint. In order to accomplish this goal, a key priority was to ensure that the kit was open and fully documented and that

developers working in SharePoint would be able to quickly review and understand the components of the kit. Because SharePoint is a highly customizable solution, and every SharePoint site can be totally different, the kit was designed utilizing an "a la carte" methodology, so that developers could pick and choose the pieces that applied to their implementation.

AKS uses a combination of replacement templates, CSS and Control adapters. SharePoint is built on top of ASP.Net 2.0, which includes a feature, called Control Adapters that allows a developer to override the default rendering of a control before it is sent to the browser. AKS makes extensive use of this feature to optimize the output for WCAG 1.0 conformance.

The AKS Control Adapters

AKS control adapters are documented so that they can be implemented in multiple scenarios. The control adapters that ship with AKS improve the accessibility of the out of the box Web parts that are used in the common scenarios for SharePoint deployments for the Internet and Intranet Web publishing scenario. Developers creating custom Web parts can use a similar approach. AKS 1.1 provides control adapters to address Web parts related to WIKIs and Blogs.

The AKS Feature

The AKS Feature is a standard SharePoint implementation that provisions a complete set of master pages, templates and CSS files that replicate the out of the box files included in SharePoint and utilized by the Publishing Portal site definition. Once the AKS Feature is activated on a site collection created using the Publishing Portal site definition, a user can select one of the AKS provided master pages that use the AKS provided CSS files. Developers can then pick through these files to see how things were implemented.

As previously mentioned, these custom ControlAdapter classes allow developers to see how the rendering of Web controls is modified from the default rendering options.

Education and Community

The educational approach is also a very important goal of the AKS solution. AKS does not contemplate an automated process to remediate any accessibility or usability concerns with current published content where the organization has implemented custom approaches that excluded the CORE SharePoint styles or masters. However, this is far from a roadblock for implementing the principals of the kit. The Guidelines and approach outlined in the AKS can be applied to existing sites and in fact should be used by companies who are in the process of hiring an outside firm to develop an internet or intranet site to better define requirements.

Best Practice Integration

The goal of the AKS community site is to not only create a forum through which SharePoint users can share their challenges and solutions, but it also provides an opportunity to bring together key stakeholders with different perspectives in an open and constructive dialogue. The community is open to anyone and the initial 150 members that have joined the community are geographically dispersed, with worldwide representation. Community members include government organizations, Universities, Non-government organizations, Advocacy organizations, technology companies, Systems Integrators, Microsoft SharePoint partners and others. Community members are also encouraged to contribute source code, and best practices to the community repository to further extend the kit. HiSoftware extends the AKS solution through its commercial applications

that also enable "end-to-end" accessibility through the SharePoint Web project life cycle. These solutions empower developers to create accessible custom templates with desktop validation and remediation software. They also enable content monitoring and automated testing through SharePoint work flows. Additionally content creators can use HiSoftware's page tester as part of the Web editor console within SharePoint. While these solutions are available at a cost to customers, they extend the concept of accessibility beyond the tools that Microsoft provides them, and allow the organization to take additional ownership of their longer term accessibility goals.

Conclusions

The AKS initiative is an ongoing project. Phase I of the kit was released in late November 2007. Phase II of the kit was released in March 2008. Phase II will be available in Q3CY2008. In the first 30 days of the release of AKS 1.0 over 1500 organizations downloaded the kit and over 150 organizations joined the AKS community and discussion list. Although this project is an ongoing initiative, several interim conclusions can be drawn:

1. By including a broader community outside of just of the software vendor you can encourage sharing even amongst competitive organizations because they are all working towards a common goal.
2. Involving the target organizations from the beginning ensures that the initiative does meet their needs in a much more relevant way.
3. By delivering small sections of the initiative incrementally value can be demonstrated early on and it ensures that the target audiences' needs are being met. If they are not being met then there is an opportunity to adjust plans and refocus on the highest priorities.
4. Developing an add-on kit with the real-world concerns of site implementers, such as cost and impact on existing installations, in mind will drive adoption and increase the field of accessible Web sites.

References

[1] Accessibility at Microsoft. Retrieved March 20, 2008, Web site: http://www.microsoft.com/enable/

[2] Chisholm, W., Vanderheiden, G., Jacobs, I. (1999, May 5). Web Content Accessibility Guidelines 1.0. Retrieved March 20, 2008, from W3C Web site: http://www.w3.org/TR/WAI-WEBCONTENT/

[3] CSS Friendly Control Adapters. Retrieved March 20, 2008, from CodePlex Web site: http://www.codeplex.com/cssfriendly

[4] Gray, R., Liu, L. Simberkoff, D. The Accessibility Kit for SharePoint. Retrieved March 20, 2008, from epractice.eu Web site: http://www.epractice.eu/cases/2526

[5] HiSoftware Accessibility Compliance Solutions for Microsoft SharePoint® Server. Retrieved March 20, 2008, from HiSoftware Web site: http://www.hisoftware.com/MOSS/MOSSsolutions.htm

[6] Microsoft Public License (Ms-PL). Retrieved March 20, 2008, from Open Source Initiative Web site: http://opensource.org/licenses/ms-pl.html

[7] The Accessibility Kit for SharePoint (AKS) Community Portal. Retrieved March 20, 2008, from The AKS Community Portal Web site: http://aks.hisoftware.com

[8] The Official Blog of the SharePoint Product Group. Retrieved March 20, 2008, from Microsoft SharePoint Products and Technologies Team Blog Web site: http://blogs.msdn.com/sharepoint/

[9] Yonaitis, Robert B. (2002). *Understanding Accessibility*. Nashua, NH: HiSoftware Publishing

Chapter 5
Understanding and Meeting Web Accessibility Standards

Introduction

In this chapter we'll look at specific features within Microsoft and HiSoftware products that help people meet international Web accessibility standards or that help people with disabilities benefit from accessible Web content. We'll also show examples of how to test Web content to determine if it is accessible.

This chapter is organized into 12 sections, one for each of the Guidelines in the W3C's Web Content Accessibility Guidelines 2.0. While this structure reflects that of WCAG 2.0, this chapter does not explain the W3C's work. For more information, please refer to the WCAG 2.0 Overview (http://www.w3.org/WAI/intro/wcag20.php).

Microsoft actively participates in European and global efforts to develop and harmonize ICT accessibility standards, including WCAG 2.0. Microsoft believes that standards harmonization will result in more accessible products that are delivered through a more economically efficient market. Similar to most IT companies, "Microsoft builds products and services for a global marketplace and strives to meet the needs of people with disabilities in all of its markets." (From http://www.access-board.gov/sec508/refresh/report/microsoft.htm)

Principle 1: Making Information Perceivable

This principle lays the foundation for the other three: end users must be able to perceive information before they are expected to interact with it. Text alternatives provide an easy and flexible way to meet this principle since text is easily made available via Braille or speech.

Guideline 1.1 Text Alternatives

> *Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols or simpler language*

Non-text content is anything that isn't text. This includes images, sounds, and videos. It even includes art created using text. For example, "colon hyphen right parentheses" creates a smiley face ":-)" which is not text, it's an emoticon. In this case, the "picture" is not perceivable to someone who cannot see and the meaning needs to be provided via a *text alternative.* Text alternatives are handy because they can be translated to Braille or to speech via assistive technologies. Therefore, the visual information can now be perceived with your ears (audio) or fingers (Braille).

Features in Microsoft products help people who create Web sites provide text alternatives. For example, in Figure 1, when inserting an image into a Web document with Expression Web, the author is prompted
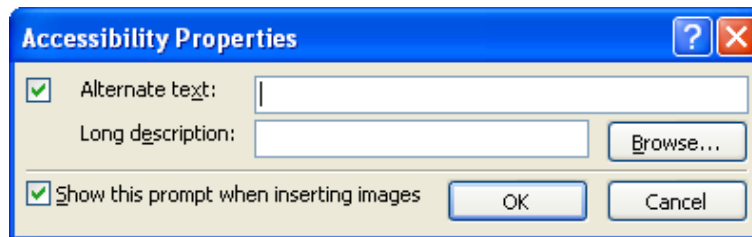

Figure 1: Microsoft Expression Web prompting for alternate text and long description for an image

to add a text alternative.

This is an example of *Accessible by Default*—an approach that Microsoft is integrating into its products. What this means for the end user is that Web sites "just work" because their browsers and assistive technologies have what they need to create a meaningful and usable experience. For the person creating Web content, this means accessibility is integrated into their Web development processes and tools. When accessibility is integrated into development tools, authors can more easily make accessible choices and, in some cases, the tool automatically produces accessible Web content without the author doing anything special. We will see more examples of the *Accessible by Default* approach throughout this chapter.

Guideline 1.2 Time-based Media

Provide alternatives for time-based media

In addition to a short text alternative, *time-based media* such as videos and audio recordings need transcripts or synchronized alternatives. A *transcript* is a text document that conveys all of the dialog and significant sounds from a video. *Synchronized alternatives* include captions and audio descriptions. *Captions* appear on the screen at the same time as words that are spoken and significant sounds are made. Captions are used by people who have difficulty hearing speech and sounds. A variety of tools are available to caption videos, including HiSoftware's Hi-Caption Studio. Microsoft's Expression Encoder can add captions to a media file and export the end-result to Silverlight.

Audio descriptions are provided by a narrator who describes visual events that cannot be understood through sound alone. Audio descriptions are useful to people who have a hard time seeing the visual content of the time-based media. For example, in the following dialog, it isn't clear why Jennifer suddenly changes her mind:

> James: "How is your day so far?"

> Jennifer: "My day is going quite well. . . .Oh! Now it seems to be a little worse."

Figure 2: Turning on the viewing of captions in Windows Media Player

When we add an audio description, it becomes clearer what has happened with Jennifer:

> James: "How is your day so far?"

> Jennifer: "My day is going quite well. . . ."

> Audio Description: A passing waiter spills a pitcher of ice water down Jennifer's back.

> Jennifer: "Oh! Now it seems to be a little worse."

There are two ways to provide captions: *open* and *closed*. Open captions are displayed all the time, for everyone viewing the video. Closed captions are only displayed when the user sets his or her preferences to display them. Figure 2 shows setting the preference Windows Media Player to show captions if available.

Audio descriptions are most commonly provided as "open" meaning that the narration is included in the soundtrack to the video and everyone hears it. Historically, there wasn't enough bandwidth to download multiple audio tracks along with a video track. But this is changing and now we're starting to see the option to turn audio descriptions on and off as well.

Guideline 1.3 Adaptable

> *Create content that can be presented in different ways (for example simpler layout) without losing information or structure*

Good visual design guides the reader's eye through a document or application. When creating Web content, you can use the underlying structure to create an auditory design or tactile design that mimics the same flow of information presented visually.

A Web author creates *structural landmarks* by using specific HTML elements. Landmarks include headings, lists, tables, and paragraphs. Figure 3 shows how to use Live Writer to mark a string of text as a heading

A person looking at the page will be drawn to the large font size and other visually characteristics that distinguish headings from other text. Scanning the headings can help someone understand the overall structure and content of a page. A person listening to a page (or using their fingers to read the page via Braille) can instruct his or her assistive technology to navigate the landmarks to gain an understanding of the overall structure and content of the page. This is in contrast to reading a page line-by-line.
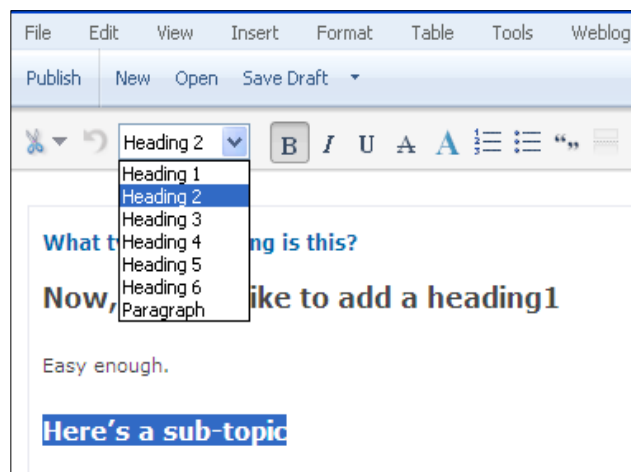
Figure 3: Using Live Writer to indicate that text is a heading

One way to determine if content has appropriate landmarks is to look at the page without the *style sheet*. Figure 4 shows two views of microsoft.com. On the left is how the page looks when the style sheet is used. The image on the right shows how the page looks when the style sheet is not used. While we use HTML to describe the structure of Web pages, we use Cascading Style Sheets (CSS) to describe the style—the colors, fonts, and placement of objects in a Web page.
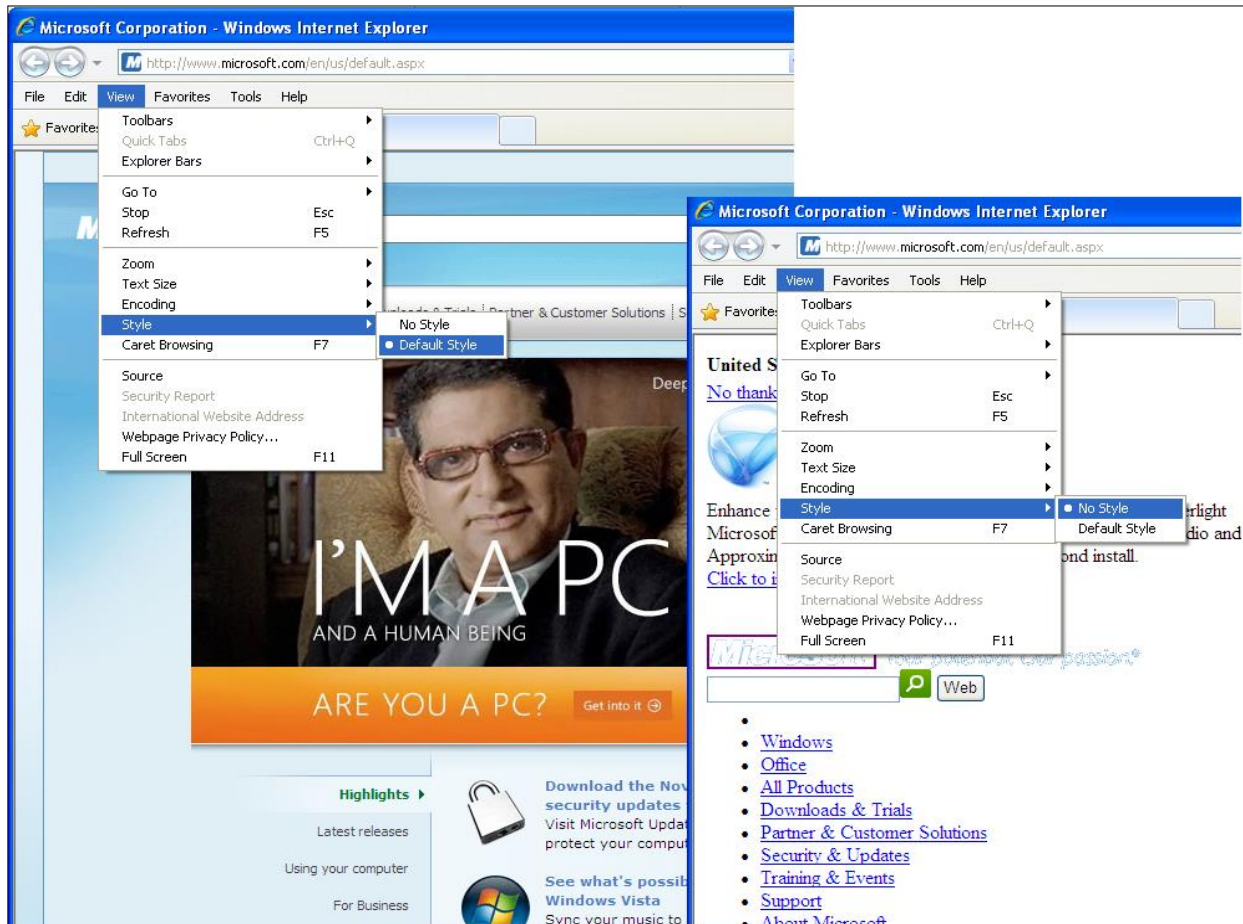
Figure 4: Changing style selection using Internet Explorer 8 to inspect underlying structure of a Web page

When you investigate a Web page without style sheets, what are you looking for to determine if structural landmarks are present? Well, does the page still make sense? Are the paragraphs of text still in chunks? Are links running into each other or are they organized in bulleted lists? Does the order make sense or do you jump all over the page?

This guideline describes the semantics that need to be infused into your HTML to ensure that structure and pathways through the structure are perceivable. We'll learn later, in Guideline 2.4, about additional landmarks that should be provided and how to ensure that users can navigate through the structure in an easy and meaningful way.

Guideline 1.4 Distinguishable

Make it easier for users to see and hear content including separating foreground from background

When color is used to convey information, some people may miss out on that information. For example, a form gathers a person's billing information and indicates, "All fields marked in red are required." Not

everyone sees red as red. For some, red appears more like yellow. People who are blind will not see any color. What should you do? Provide a redundant cue, like an asterisk. However, you will need to tell people what the asterisk means.

A new technology that is quickly gathering support in browsers and assistive technologies is the W3C Web Accessibility Initiative's Accessible Rich Internet Applications Specification (WAI-ARIA). Microsoft has contributed to the development of this specification and Windows Internet Explorer 8 is one of the first browsers to implement the specification. WAI-ARIA provides a new vocabulary to describe the objects in Web applications and make them more accessible. In the case of the required field, there is a WAI-ARIA role "required." When this is used, people no longer need to be told what asterisk means or to guess, their tools will know that it is required and can tell them.

When authors use color combinations that are difficult to read, Internet Explorer allows people to override authors' colors and use combinations that they find easier to read. Figure 5 shows the Internet Explorer dialog box that allows users to choose colors for text, backgrounds, and links.
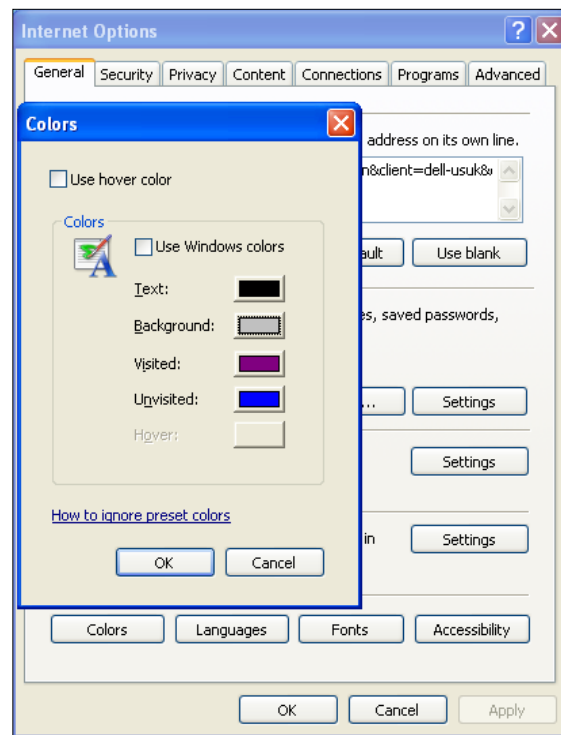
Figure 5: Setting colors in Internet Explorer. "Colors" is an option on the "Internet Options" panel

Similarly, Windows has a setting, *high contrast mode*, which will display every window and button in a Windows' application in a limited palette of colors.

Not only can text be displayed as Braille or spoken aloud by a screen reader, but onscreen it can be resized. Internet Explorer has two options for making text easier to read: zoom and text size. Zoom is useful for enlarging buttons and banners that are *images of text*—someone has used a graphics program like Adobe's Photoshop or Microsoft's Expression Design to use a specific font or create an unusual

effect that is not possible with style sheets. On the other hand, if the text is text, then an end user can make it larger by setting the *text size* to "Larger" or "Largest."

Principle 2: Making User Interfaces Easy to Operate

After someone can perceive all of the objects and information in a Web page, they can begin to interact with it. Whereas the first principle is ensuring that people have access to the information using the sense they prefer (vision, hearing, feeling), the guidelines in this principle are organized around allowing people to use the input device of their choice, such as a mouse, a keyboard, or a microphone (for voice input).

Guideline 2.1 Keyboard Accessible

Make all functionality available from a keyboard

Many Web applications are designed to be operated by a mouse, but there are many people who are not able to use the mouse. For many Web applications that are designed only for mouse input, this means menus do not pop-up to show submenus or that filled in forms cannot be submitted.

Adding keyboard support to an application can be more complicated than adding mouse support. A mouse is a single switch that is usually only modified with a few keys. A mouse can click on, drag with or hover over an object. A keyboard, on the other hand, has 101 keys that can be combined in a variety of ways and these combinations act differently between browsers and across operating systems. An additional layer of complexity is due to the fact that many assistive technologies are operated via key combinations.   This is another great place to apply *Accessible by Default*. Ensuring that the building blocks of Web application have keyboard-support built in is the best way to ensure Web applications are keyboard accessible. To this end, Silverlight 2.0 controls have built-in keyboard support.

Guideline 2.2 Enough Time

Provide users enough time to read and use content

Because you don't know how long someone will need to read, understand or interact with content, ensure that if your application has a time limit the user has control over how often the content updates or how long they have to complete a task. There are a few exceptions to this guideline. For example, if the task is a real-time event, such as an auction, and it isn't possible to provide an alternative, then you do not need to provide user control over the event. In some cases, this control is provided by the browser. For example, Internet Explorer allows the user to pause most animations. Most media players, including the Accessible Media Player and Windows Media Player have pause buttons.

- If you are developing an application that is frequently updated with new content, you might consider one of the following options: Allow the user to turn off the time limit by setting a user preference;

- Provide the user a way to set the time limit themselves, through a user preference or some other mechanism;
- Warn the user with a pop-up that time is about to expire and let them extend it by pressing "ok" to extend the time limit.

Guideline 2.3 Seizures

Do not design content in a way that is known to cause seizures

WCAG 2.0 Success Criteria are divided into three Levels. In most cases, organizations will build policy around the fist two levels: Level A and Level AA. Level AAA is the most difficult to achieve, but in some cases, easier to understand. This is the case with this guideline. We suggest teaching developers to avoid creating content that flashes more than three times per second. If that is too restrictive for someone or they push back for some other reason, refer them to the definition of *general flash and red flash thresholds* (http://www.w3.org/TR/WCAG20/#general-thresholddef).

You can also refer them to the Photosensitive Epilepsy Analysis Tool (PEAT) available at: http://trace.wisc.edu/peat/

Guideline 2.4 Navigable

Provide ways to help users navigate, find content and determine where they are

In Guideline 1.3 we discuss the importance of providing structural landmarks that can help people navigate through Web content to emulate the eye scanning through a document. Why is order important? Consider a recipe for cooking pasta: first you boil some water, add dry pasta, and then let it boil for about 10 minutes. That is a meaningful sequence. If it were mixed up somehow, such that you read, "add dry pasta, boil for about 10 minutes, boil some water" you could end up with burnt pasta.

One way to test that your document or application has a meaningful sequence is to use a free tool called *inspect32*. If you choose "show highlight rectangle" and disable "watch cursor" from the toolbar, as you tab through the links on a page, inspect32 will highlight the links as you tab through them. We'll talk about inspect32 again in Guideline 4.1 and we'll learn more about the rest of the information that inspect32 makes available.

This guideline also suggests providing additional pathways and "secret doors" that will help people more easily find their way to specific pieces of content on a page. People accessing the Web via a screen reader will often get a sense of the structure and contents of a Web page by reading through a list of links on the page. Ensuring that link text makes sense when read out of context is an important strategy to help people gain a quick overview.

This means that a page full of links that say "Click here" or "More" won't be as useful as using a more descriptive link. This usually happens on portal sites where there are many options to choose from: each description of an item is followed by a "More" link. When someone creates a list on one of these

pages, all they hear or see is "More. More. More." In Figure 6, the Microsoft Enable site starts each description with a descriptive link then provides additional information about each story.

Figure 6: Example of link text that makes sense when read out of context

The link list for this part of the Microsoft site is:

- Inclusive Innovation Showroom Opens to Demonstrate Accessibility
- Sales manager and busy mom utilizes magnification and screen reader
- Active retiree makes her computer easier to see and use
- Banker enjoys urban exploration with a Pocket PC and screen reader
- Medical researcher, card shark, and speech recognition fan

Principle 3: Making information understandable

Now that your content is perceivable to a variety of people and they can operate the controls, can they understand the results of those operations?

Guideline 3.1 Readable

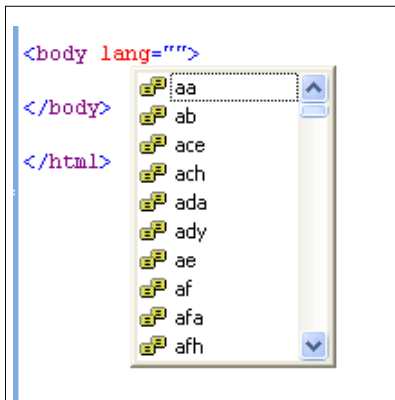### *Make text content readable and understandable*

Figure 7: An example of IntelliSense, a list of language codes in Expression

At Level A, this means indicating the language of a document. In HTML, an author can indicate language by using a 2 to 3 character code such as "en" for English or "fr" for French. Figure 7 shows an example of *IntelliSense*—a technique that prompts the user with a list of what they might like to do next. In this case, an author is presented with a list of language codes.

Higher levels of this guideline provide information about making text easier to understand. This includes using the correct spelling for words, following grammatical rules, and checking the reading level of your text. Many tools, such as Microsoft Word, incorporate spellchecking into the interface—typically red squiggles underline misspelled words or an end user can go through a list of misspelled words one at a time.

Reading level tests (or readability tests) look at the average number of syllables per word and the average number of words per sentence. It is not a perfect measure of readability, but they can provide a good general indication of the reading level of your content.

Guideline 3.2 Predictable

Make Web pages appear and operate in predictable ways

This basically means: don't surprise the user. If you want the user to move the focus from one object to another, do it in a way that they understand what is happening and only move focus when someone is expecting it. When an application is unpredictable or the user loses track of where they are or what they've been doing, at best case they get frustrated and worse case they are disoriented and unable to continue what they were doing. Here are some general design tips for ensuring users find your application predictable and enjoyable to use:

- Don't have the interface change too much as the user moves focus either with the mouse or especially by the keyboard. Some of this is ok, like submenus displaying on mouse hover, but opening a new window or changing focus to something else are too disorienting to users.

- Similarly, as a user begins to interact with an object, don't cause too many things to happen unless the user is well aware of what is about the change. For example, if someone begins to select a country from a drop down menu, let them select the country and move to the next control before changing the interface based on their selection.
- If you have a navigation panel or some other set of objects that appear throughout a Web site, keeping them in the same order and with a similar appearance will give users a "handrail" that they can refer to throughout the site. Some sites allow users to customize the order in which objects appear on a page—that's fine, but ensure that the order that the user chooses is honored on pages throughout the site.

Guideline 3.3 Input Assistance

Help users avoid and correct mistakes

As we mentioned before, A*ccessible by Default* techniques such as *IntelliSense* can help content creators make Web applications accessible. These techniques also help make the Web more usable for everyone. For example, Figure 8 shows that someone has tried to search for "Web content accessbility guidelnes" using Microsoft Live Search. "Accessibility" and "Guidelines" have been misspelled, and Live Search shows the results for the correctly spelled words as well as an option for the user to see results for the misspelled words.
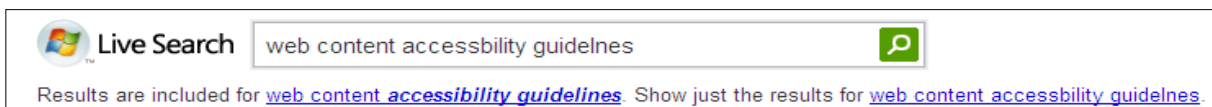

Figure 8: Live Search showing search results for the correct spelling of a search term as well as the misspelled phrase

Other examples include the accelerators and other predictive features in Internet Explorer 8, such as visual search and the smart address bar.

Principle 4: Making Information Work Reliably with User Tools

Of all of the guidelines that are worth the transition to WCAG 2.0 from WCAG 1.0, this Principle is the key. The core of interactive Web application accessibility is covered in Principle 4. We mentioned WAI-ARIA earlier, and it's here that it really shines.

Guideline 4.1 Compatible

Maximize compatibility with current and future user agents, including assistive technologies

Part of making applications work with assistive technologies is how you communicate with them. An accessibility Application Programming Interface (API) is the language that an application uses to communicate with an assistive technology. APIs provide the vocabulary for an application to describe what role an object plays in an interface (a button or a checkbox), what the state of the object is (is the checkbox checked?), and what you can do to an object (push a button, check a checkbox). APIs also

allow an application to tell the assistive technology, "My content has updated. Here's what has changed." The assistive technology can then determine how and when to tell the end user about the changes.

Silverlight applications and Internet Explorer both use APIs to communicate with assistive technologies. Internet Explorer was the first browser to expose HTML elements to the Microsoft Active Accessibility Interface and is one of the first browsers to implement WAI-ARIA to extend that support. Microsoft continues to work with the W3C to standardize the mapping of HTML information to accessibility APIs.

One way to test how well an application communicates with an assistive technology is to use insepct32—a free tool from Microsoft. What sorts of information should you look for? If something looks like a checkbox, is it identified as a checkbox or is identified as something else, like a graphic? If it is a checkbox, can you tell if it is checked or not checked? These are just two examples of the types of information you can learn about a user interface element in an application. With WAI-ARIA, the amount of information available about an HTML widget is much richer. A similar tool is available to inspect accessibility information in Silverlight applications. It's called UIAVerify and is also freely available from Microsoft.

Another important aspect of this guideline is ensuring that the HTML used to create a web page is written in accordance with standards. This is what people mean when they talk about "valid code"—it is HTML that has been run through an automatic checker to ensure it meets the language standards. This is another example where *Accessible by Default* can have a huge impact. Expression Web, Live Writer, and other Microsoft tools make it easy for content creators to generate valid code and also make it easy to test for and correct validation errors; in some cases, the code produced by these tools automatically validates, requiring nothing further from an end user.

Conclusion

In this chapter we looked at the twelve guidelines in the Web Content Accessibility Guidelines 2.0. We learned the basic gist of each guideline and showed strategies for meeting each of them. In some cases, we showed how following the guidance allows end users to access Web content and some tips to help you figure out if content meets the guidelines. We also talked a bit about testing and the different types of testing that an organization should do to ensure that the content meets the design specifications, that it meets accessibility requirements, and that it is usable to people with a variety of needs.

Conclusion

Throughout this guide we've learned about practical strategies for making web content accessible and that there are several pieces that need to fit together to create an organizational approach to accessibility. We've learned how important it is for an organization to set accessibility standards and seen a real-world example, the Accessibility Kit for SharePoint, that helps integrate accessibility testing into the development process—making it easier to meet organizational standards. We've seen how accessibility can increase sales and decrease maintenance costs, building a good business case for integrating accessibility into an organization's web development processes. A key strategy for helping developers implement accessibility is to make it easy for them to make the right decisions and we learned about several examples of *Accessible by Default*, where accessibility strategies are built into development products.

The European Dialogues on Web Accessibility have been a rewarding experience and created the basis for an evolving body of knowledge that can help us reach our shared goal of an accessible Web. Microsoft and HiSoftware will continue to partner with others to improve the Riga Dashboard and increase web accessibility.

List of Resources

Microsoft Products

- Accessible Media Player: http://www.codeplex.com/amp
- inspect32: http://www.microsoft.com/downloads/details.aspx?familyid=3755582A-A707-460A-BF21-1373316E13F0&displaylang=en
- Internet Explorer: http://www.microsoft.com/windows/products/winfamily/ie/
- Microsoft Expression (includes Studio, Design, Web, Media, Blend and Encoder): http://www.microsoft.com/expression/
- Microsoft Office: http://office.microsoft.com/
- Microsoft Office SharePoint Server (MOSS): http://www.microsoft.com/Sharepoint/
- Silverlight: http://silverlight.net/
- Windows Live (including Live Writer): http://windowslivewriter.spaces.live.com/
- Windows Media Player: http://www.microsoft.com/windows/windowsmedia/player/

HiSoftware Products

- Accessibility Kit for SharePoint: http://aks.hisoftware.com/
- HiSoftware solutions for accessibility:
  http://www.hisoftware.com/hmcc/
  http://www.hisoftware.com/MOSS/MOSSsolutions.htm

W3C Standards

The World Wide Web Consortium (W3C) maintains the following Web standards.

- Cascading Style Sheets (CSS): http://www.w3.org/TR/CSS/
- Hypertext Markup Language (HTML): http://www.w3.org/TR/html/
- Web Accessibility Imitative Accessible Rich Internet Applications (WAI-ARIA): http://www.w3.org/WAI/intro/aria.php
- Web Content Accessibility Guidelines (WCAG) 1.0: http://www.w3.org/WAI/intro/wcag10docs.php
- WCAG 2.0: http://www.w3.org/WAI/intro/wcag20.php

About This Guide

This guide was created by Microsoft and HiSoftware to summarize the practical strategies discussed at the European Dialogues on Web Accessibility. Contributors to this guide include James Thurston, Dana Simberkoff, Cynthia Shelly, and Wendy Chisholm.

Microsoft

Microsoft Corporation is a multinational computer technology corporation that develops, manufactures, licenses, and supports a wide range of software products for computing devices.   Founded in 1975, Microsoft is the worldwide leader in software, services and solutions that help people and businesses realize their full potential. More information about Microsoft accessibility is available at: http://www.microsoft.com/enable/

HiSoftware

Founded in 1998, HiSoftware is a leading provider of software, services, and on-demand solutions that test, repair, monitor and enforce audited Web content, quality, and regulatory compliance. The company's solutions empower content developers, Web site architects, and executives to work collaboratively to create and manage corporate Web standards for accessibility, financial banking compliance, privacy, operational security, search engine optimization, site quality and performance, and application transaction testing. With over 4,000 customers across 88 countries, HiSoftware's clients boast many of the world's largest public and private sites including US and international government agencies and Global 2000 companies such as Gateway, HP, California State University, US Department of Transportation and Microsoft. For more information about HiSoftware and its solutions go to www.hisoftware.com

There is now the opportunity and the imperative to make real progress toward a more inclusive society through an accessible Web.

This handbook from Microsoft and HiSoftware is one result of a series of interactive dialogues on the current state of Web accessibility between government officials, industry leaders, and NGO representatives across several European countries. It combines practical strategies with some of the perspectives, goals, and vision that came out of those dialogues. Our intent is to provide a tool that will help organizations worldwide foster accessibility in their operations.

This handbook begins with a summary of the important knowledge shared at the series of European dialogues on Web accessibility. Those lessons-learned and best practices are followed by an example process for implementing a comprehensive Web accessibility program in any organization. Building on that framework, there is an example of a product and community that helps organizations implement some of the practical strategies from the dialogues. Finally, there is a discussion of specific tools and tips to understand and meet Web accessibility standards.