Evaluating Web Accessibility: automated, manual and user-based testing techniques

Helen Petrie, University of York, UK
Chris Power, University of York, UK
Gerhard Weber, Technical University of Dresden, Germany
Introduction

Gerhard Weber
Overview of the workshop

Introduction - Gerhard Weber
Using assistive technologies to access the Web - Gerhard Weber
Coffee break
Guidelines for Web accessibility - Chris Power
WCAG2 guidelines - David MacDonald
Hands-on exercises and demonstration of screen reader
Lunch
Research on web accessibility - Helen Petrie
Automated evaluation - Gerhard Weber
Coffee break
Manual evaluation - Chris Power
User evaluation and UWEM - Helen
Conclusions and discussion
Usability, accessibility and the Web

- disability
- guidelines
- assistive technologies
- testing
  - automated, manual and user-based

Course we need his head you idiot!
We’re doing Psychological experiments today!

www.gerardkeegan.co.uk/.../cartoongallery_b.htm
Individualization

Individualization (ISO 9241-10) requires adaptation of presentation media and interaction

- Individualization by selection of modules: *tailoring*
  - application islands, often not compatible to each other
- Individualization is initiated by the user: *adaptation*
  - example: wallpaper, ring tone
- Individualization is initiated by the computer: *adaptivity*
  - based on a user model
  - plan recognition
  - plan completion
Individualization

Individualization by user model
- task specific
- situation specific
- user specific
⇒ users with special needs

Diagram:
- User Expert
- Task Expert
- Situation Expert
- Blackboard
- Plan and Task recognition
- Response Generation
- Input
- Output
Disability

Disability is “a physical or mental impairment which substantially limits one or more major life activities” (Americans with Disabilities Act, 1991)

Access to the Web affects 8.7% of Europeans

European Demographics (Gill, 2004):
- wheel chair bound: 0.4%
- walking aids: 5%
- finger loss: 0.1%
- loss of limbs: 0.1%
- reduced dexterity: 2.8%
- lack of coordination among limbs: 1.4%
- impaired speaking: 0.25%
- language impaired: 0.6%
- mentally impaired: 3%
- dyslexia: 1%
- deaf: 0.1%
- hearing impaired: 6%
- blind: 0.1%
- low vision: 1.5%
Disability by UNESCO

Disability … is a social phenomenon arising from individual capabilities, the situation and the environment

**Problem:** IT supports adaptation for individual, situation specific limitations, possibly created by the environment and the social group

=> IT integrates disabled users
Reversing integration

Any new kind of human-computer interaction may create barriers, if their adaptation is not known

1. graphical user interfaces
   - pixel-barrier
   - mouse and other pointing devices

2. hypertext/WWW
   - lack of overview

3. multimedia and hypermedia
   - multimedia-barrier
   - lack of temporal control
Accessibility and Usability

Are accessibility and usability intersecting sets- what is their relationship?

“Pure accessibility” problems which affect only people with disabilities,
problems which affect all users (both disabled and non-disabled) (“universal usability” problems)
and perhaps “pure usability” problems

Petrie and Kheir (2007)
Using Assistive Technologies to Access the Web

Gerhard Weber
Groups affected by web accessibility

- Not only people with visual disabilities who have difficulty with the Web

- Key groups include:
  - People with physical disabilities, e.g. dexterity problems
  - People with hearing disabilities e.g. Deaf, deafened, hard of hearing
  - People with cognitive disabilities, including those with specific learning difficulties e.g. Dyslexia
  - And people with Visual disabilities, both blind and partially sighted
People with physical disabilities

- require alternative input devices
- sticky keys
- little or no control over hands
  - small keyboards/large keyboard
  - control by speech
  - possibly joystick/touchpad/head mouse if dexterity is sufficient
  - switch access and pointers/ head mounted/mouth operated/

Control the mouse with the keyboard
- Turn on Mouse Keys
  Use the numeric keypad to move the mouse around the screen.
  Set up Mouse Keys

Make it easier to type
- Turn on Sticky Keys
  Press keyboard shortcuts (such as CTRL+ALT+DEL) one key at a time.
  Set up Sticky Keys
- Turn on Toggle Keys
  Hear a tone when you press CAPS LOCK, NUM LOCK, or SCROLL LOCK.
  Turn on Toggle Keys by holding down the NUM LOCK key for 5 seconds
- Turn on Filter Keys
  Ignore or slow down brief or repeated keystrokes and adjust keyboard repeat rates.
Keyboard catastrophes

- irregular link order
- tab stops made invisible with CSS
- too many ads linked
- no keyboard support in menus
- no temporal control in animations

Exercise: browse by scanning keyboard
Speech disabilities

- “A conversation is a dialogue in which the one taking a breath is called the listener”
- People have speech disabilities due to a range of conditions and traumas
- Speech is rapid: 150 words per minute for normal conversation
- Compare: a skilled typist achieves 80 words per minute
- but often limited manual dexterity
Alternative and Augmentative Communication (AAC)

www.peepo.com
BLISS

- 1942-1965 developed by Charles K. Bliss
- Approx. 25 single symbols make up 1400 terms
- EU-Project WAAC developed concept coding for BLISS
Concept Coding: Project WWAAC

© H.Petrie, C.Power, G.Weber

[Judson, 2003]
Concept Coding Example

- The Concept CC-DRINK-1001 has:
- Two sub-concepts (hyponyms) CC-TEA-1001 & CC-COFFEE-1001, whilst also possessing:
  - a representation which is of type bliss image
- The Concept CC-COFFEE-1001 has:
  - a parent concept (hyponym) CC-DRINK-1001 & has:
    - a representation which is a bliss image
- The Concept CC-TEA-1001 has:
  - a parent concept (hyponym) CC-DRINK-1001 & has:
    - a representation which is a bliss image
RDF Graph for “DRINK”
“DRINK” in XML Concept Coding

```xml
<rdfs:Class rdf:ID="Concept"/>
<rdf:Property rdf:ID="hasRepresentation">
<rdfs:domain rdf:resource="#Concept"/>
<rdfs:range rdf:resource="#Literal"/>
</rdf:Property>
<rdf:Property rdf:ID="hasParent">
<rdfs:domain rdf:resource="#Concept"/>
<rdfs:range rdf:resource="#Concept"/>
</rdf:Property>
</rdfs:Class>
<Concept rdf:ID="CC-DRINK-1001" hasRepresentation="Bliss00349">
<hasParent rdf:resource="#CC-LIQUID-1001"/>
</Concept>
<Concept rdf:ID="CC-TEA-1001" hasRepresentation="Bliss00823">
<hasParent rdf:resource="#CC-DRINK-1001"/>
</Concept>
<Concept rdf:ID="CC-COFFEE-1001" hasRepresentation="Bliss00827">
<hasParent rdf:resource="#CC-DRINK-1001"/>
</Concept>
```
Deafness

- sign language in becoming accepted as a mother tongue in many countries
- language competence low (10 year old for congenitally deaf people)
- previously seen as speech impairment and marginally an access problem
  - Multimedia requires redundancy of media: sign language in TV/WWW
  - Telecommunication
- Telecommunication Devices for the Deaf (TDD)
  - Baudot terminal (keyboard, modem)
  - Deaf relay centres
    - video telephony in Germany for professional use
- Common WWW issue: contact forms
Problems Faced by Deaf Users

- Deaf people may have problems with reading
  - Because they are native users of a Sign Language, so using written language is also a second language

- May have a legal right to information in their native language (e.g. British/German Sign Language recently recognised in the relevant countries legally as a language)

- Human interpreters are required
  - Little acceptance of synthetic avatars
  - Strong preference for interpreters to be Deaf themselves
Deaf people’s use of the Web

- Deaf people are currently unlikely to use any assistive technology on the Web
- Greatly appreciate signing and subtitling of film, theatre, opera and television
- People from the Sign language community really want interpretation “from the deaf viewpoint”
- Even some signing is appreciated - in our MultiReader Project we provided signing of the navigation icons, which was greatly appreciated
Hearing Disability

- a “hidden” disability: often difficult to recognize
- Background sounds are disturbing
- Hearing aids allow greater understanding
- Visible Speech: lip reading supports better understanding of speech
People who are deafblind

- a very small number of people, very neglected by society
- Social binding to communities of the deaf or blind depends on education
- Some deafblind people use E-Mail, etc.
- often: “hand writing” letter by letter:
  - lorming (Hieronimous Lorm)
  - UK: alphabet
Web-related Problems

- Unclear and confusing layout of pages
- Confusing and disorienting navigation mechanisms
- Lack of alternative media for audio-based information and complex terms/language
- Inappropriate use of colours and poor contrast between content and background
- Graphics and text size too small
Dyslexia

- Dyslexia is a difficulty in some subset of reading, writing, spelling and computing
- Origins are highly controversial
- Overcome by skilled specialist teaching and the use of compensatory strategies
- Dyslexia can occur at any level of intellectual ability.
- It is not the result of poor motivation, emotional disturbance, or sensory impairment
Typical characteristics of dyslexia

- bright in some areas, with a 'block' in others
- late in learning to talk, or with speaking clearly
- particular difficulty with reading or spelling
- put figures or letters the wrong way e.g. 15 for 51, 6 for 9, b for d, was for saw
- Mirror write
- spell a word several different ways without recognising the correct version
- confuse left and right
- answer questions orally but have difficulty writing the answer
- have trouble with sounds in words, e.g. poor sense of rhyme
People with dyslexia – technologies

- People with dyslexia may use software such as SpeakOut and textHELP!
- But often don’t use assistive technology for the Web, so need to be able to adapt appearance of pages
- Short lines of text, different colour combinations and highlighting of text can help deal with some of these problems
- Text-to-speech synthesis can help people with severe dyslexia
Problems with websites encountered by people with dyslexia

- Unclear and confusing layout of pages
- Confusing and disorienting navigation mechanisms
- Inappropriate use of colours and poor contrast between content and background
- Graphics and text size too small
- Complicated language or terminology

DRC study, top 5 problems of people with dyslexia
People with visual disabilities

- Most blind people may have some vision, light/dark, colours
- Only about 10% read Braille, mostly for professional use
- Most use synthetic speech to read computer output
- Most are experienced keyboard users (but: three finger keystrokes)
People with visual disabilities

People with partial sight

Many ways to be partially sighted
- Restricted field
- Loss of central vision
- Blurred vision
- (moving) holes in vision
- Darkening of vision

Source: National Eye Institute, National Institutes of Health
Paperless Braille

- 8-dot and 6-dot Braille
- grade I and II
- maths and music notations
- piezoelectric
Reading by Braille display

- **Tracking**
  - caret
  - cursor

- **Exploration**
  - line by line to gain an overview
  - decoupled from the application

- **Routing**
  - activate sensors
  - synchronisation between application and Braille display

- **Adaptation**
  - function keys
  - configuration files
  - script language
Interaction by Routing keys

Mouse move

word in

click routing key

© H.Petrie, C.Power, G.Weber
Mouse operations by Braille

Mouse click

\[\text{wordIn}\]

click routing key

\[!\rightarrow S1\]

\[==\]
Direct Manipulation in Braille

- requires some mode
  - select source
  - explore destination
  - synthesized mouse move

<table>
<thead>
<tr>
<th>Hold mouse button</th>
<th>1. Double click</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>word</code> in</td>
<td><img src="image1" alt="Diagram" /></td>
</tr>
<tr>
<td>Drag and release</td>
<td>2. Double click</td>
</tr>
<tr>
<td><code>in</code> word</td>
<td><img src="image2" alt="Diagram" /></td>
</tr>
</tbody>
</table>

© H.Petrie, C.Power, G.Weber
Screen Reader Architecture

- **bottom-up**: OCR

- **top-down**: meta widgets

- **middle-out**: filtering the
  - GUI
  - DOM
Non-visual Design: Pixels

- menu bar
- accelerator key
- selected option
- pulldown menu
- menu separator
- short key
The future: large tactile displays?
Non-visual. Design: spatial
Non-visual. Design: Structure

- an Off-Screen Model:
Dynamic Contents

- Dynamic Contents is controlled by Javascript
  - sometimes called also DHTML, but may be confused with HTML+TIME (Microsoft)
  - Javascript uses events for user input such as mouse move, mouse over or keypress
  - dynamic behaviour may be independent from user input through timers
  - arbitrary changes to the Document Object Model are possible: developers create their own interaction techniques
  - WCAG 1.0: (1.1) Use the NOSCRIPrT element to describe the action or replace the functionality of any scripts you embed in your page
Users have dynamic problems

- Screenreader may fail to track the change of focus if mark-up doesn’t tell
  - WCAG 1.0: (6.2) Ensure that equivalents for dynamic content are updated when the dynamic content changes
- Often keyboard access is not provided
  - A solution must address all users, no personalisation considered (enable/disable sign language videos)
  - If Javascript is disable site must still be operable
  - Screenreader taken care for some Javascript functionality through specific modes for exploration
- Temporal control may be missing (dynamic counters in a quiz, animations)
AJAX

- Asynchronous JavaScript and XML (AJAX, Garrett, 2005) is based on the asynchronous XMLHttpRequest of HTTP
  - XML data may be exchanged between server and browser
  - some GUI frameworks support AJAX: ASP.NET Ajax; Dojo
  - Dojo may support screenreaders:

Fisheye Demo using 1.0
Examples for AJAX

- word processing (http://www.writely.com)
- spreadsheet (http://ajaxxls.com/)
- Flickr, Youtube
- several drawbacks
  - no Bookmarks
  - no History/Undo
  - may be overcome by additional programming effort
Sample AJAX: One Page

```html
<html> <head> <title>AJAX</title>
<script type="text/javascript"> <!--
var http = null;
if (window.XMLHttpRequest) {
    http = new XMLHttpRequest();
} else if (window.ActiveXObject) {
    http = new ActiveXObject("Microsoft.XMLHTTP");
}
if (http != null) {
    http.open("GET", "file.html", true);
    http.onreadystatechange = ausgeben;
    http.send(null);
} --></script>
</head>
<body>
HTML from Server:
<div id="output"></div>
</body>
</html>

function ausgeben() {
    if (http.readyState == 4) {
        document.getElementById("output").innerHTML =
            http.responseText;
    }
} //--></script>

0 – not initialized ...
4 – ready
//onreadystatechange
</head>
<body>
HTML from Server:
<div id="output"></div>
</body>
</html>
```
Modified dynamically

- `file.html`

  `<p> AJAX is fun </p>`
  `<p> AJAX erm&ouml;glicht spannende Effekte </p>`
Screenreaders and AJAX

- Screenreaders keep a copy of the DOM in separate buffer. Full command set for exploration (link list, headings, etc.)
  - Jaws: Virtual PC Cursor
  - Windows Eyes: Virtual buffer
  - Supernova: Virtual focus mode

- Problems
  - onreadystatechange Event wird von Jaws 7.0 in Firefox beachtet jedoch nicht in IE, fixed in Jaws 9.0
  - Focus cannot be set in many tags very well
    - a, area, button, input, object, select, and textarea
    - HTML2 is preapred to fix lack focussing
  - Quick (and dirty) solution: use tabindex=-1 attribute
ARIA

- New W3C guideline Accessible Rich Internet Applications is under preparation
- independent from HTML
- mark-up and Javascript get access to widgets:
  - “Each element or widget is marked with full and corrected semantics that fully describes its behavior (using element names or roles).
  - The relationships between elements and groups are known
  - States, properties, and relationships are valid for each elements behavior and are accessible via the DOM.
  - There is an element having the correct input focus.”
Roles make up a Contract

- *Roles* are described in RDF, e.g.

  ```html
  <div x2:role="wairole:tree" >
  <div x2:role="wairole:treeitem" >Veggies
  <div x2:role="wairole:treegroup">
  <div x2:role="wairole:treeitem">Green
  <div x2:role="wairole:treegroup">
  <div x2:role="wairole:treeitem">Asparagus</div>
  </div>
  </div>
  </div>
  </div>
  </div>
  ```

[http://www.w3.org/WAI/PF/roadmap/DHTMLRoadmap040506.html]
ARIA and dynamic contents

- Consider a AJAX based chat
  - user works with edit field
  - new text added asynchronously by remote user
  - Screenreader cannot show user’s and remote user’s input concurrently
- ARIA proposes 'aria-live'
  - The live attribute may have the values off (default), polite, assertive, or rude. The value of this attribute describes the verbosity assistive technology users can expect from an alert or live region.
ARIA describes dynamics

- **off**: region is not live.
  
  `<p id="lr" aaa:live="off">Initial Content</p>`

- **polite**: normal operation. It is not necessary to respond until user completes their current activity.
  
  `<p id="lr" aaa:live="polite">Initial Content</p>`

- **assertive**: a higher priority than normal but does not necessarily interrupt the user immediately.
  
  `<p id="lr" aaa:live="assertive">Initial Content</p>`

- **rude**: highest priority and will typically result in the user being interrupted.
  
  `<p id="lr" aaa:live="rude">Initial Content</p>`
How partially sighted people use the Web

- tracking
- exploration
  - by mouse
- Re-colouring
  - inverse background
- staircases
- speech synthesis
Colour vision deficiencies (“colour blindness”)

- About 10% of the population have some form of colour vision deficiency: 8 – 12% of men and 0.5% of women
- Most common forms in perception of red and green (see next slide)
- Aging affects our colour perception – the average 80 year old has only 40% of the colour vision of a 20 year old
- If three Caucasian males come and look at your website, there’s a 22% chance that one of them will have a colour vision deficiency
Green weakness (Deuteranomaly)

- Affects 5 in 100 males
- Perception of green as red (missing/weak green-sensitive pigment)
- Poor discrimination in the red/orange/yellow/green region of the spectrum
Scaling of Images

- For browsers available: Scalable Vector Graphics (SVG)
Summary: User Requirements for Interacting with the Web

- **Blind users**
  - Controlled speech instead of text (speed)
  - Description of images
  - Audio descriptions

- **Partially sighted users**
  - Variable fonts and colours
  - Zoom
  - Speech
  - Descriptions of images
  - Audio descriptions
Summary: User Requirements for Interacting with the Web

- **Deaf readers**
  - Textual description of sounds, speech
  - Images, animation, video in addition to text
  - Sign language dictionary
  - Sign language translations of text
  - Short texts
  - Variable fonts, colours, spacing
  - Highlighting

- **Hard of hearing users**
  - All of the above
  - Talking heads
User Requirements for Interacting with the Web

- **Dyslexic users**
  - Variable fonts, colours, spacing
  - Highlighting
  - Short lines
  - Wider inter-line spacing
  - No extraneous movement

- **Users with physical disabilities**
Guidelines for accessibility

Chris Power
Introduction to Guidelines

- Web Content Accessibility Guidelines (WCAG)
- Defined by the W3C Web Accessibility Initiative (WAI)
- The *de facto* international standard for accessibility on the web
- Basis for much of the international legislation on web accessibility (Section 508, many European nations)
WCAG Overview

- WCAG 1.0 accepted in 1999; now almost 9 years old
- WCAG 2.0 is currently a candidate recommendation (August, 2008)
- Other guidelines are available from WAI, such as UAAG (user agents), and ATAG (authoring tools)
WCAG Fast Facts

- 70% of web developers report performing accessibility testing
- Only 54% report using WCAG in their accessibility testing
- The majority of those who use WCAG use it primarily for automated testing techniques.
  - Source Benchmarking Tools for Web Accessibility (BenToWeb)
WCAG 1.0 Structure

65 Checkpoints \{ 14 Guidelines \}
WCAG 1.0 Examples

1. Provide equivalent alternatives to auditory and visual content

1.1. Provide a text equivalent for every non-text element

1.2. Provide redundant text links for each active region of a server side image map
WCAG Conformance

- What does it mean for a website to be conformant to WCAG?
- Each of the 65 checkpoints have a priority level attached to them
- Which checkpoints you pass determines what level of conformance you meet
Level A Conformance

- Meet all **Priority 1** checkpoints
- **Priority 1**: You must satisfy these or certain user groups will find it impossible to use web documents
- Example:
  
  *Checkpoint 1.4*: For any time-based multimedia presentation (e.g. a movie or animation), synchronize equivalent alternative visual track) with the presentation
Double-A (AA) Conformance

- Meet all **Priority 1 & 2** checkpoints
- **Priority 2**: You *should* meet these checkpoints otherwise some user groups will find it difficult to use web documents
- Target level of conformance - yet only 20% meet this level
- Example:
  
  *Checkpoint 3.6*: Mark up lists and list items properly
Triple-A (AAA) Conformance

- Meet all **Priority 1, 2 & 3** checkpoints
- **Priority 3**: You *may* meet these checkpoints. This will greatly improve access to web documents for certain user groups
- Example:
  
  *Checkpoint 5.5*: Provide summaries for tables
Guideline Testing: Automatic

- Some aspects of guidelines can be tested with automatic tools
- Those things that can be programmatically determined
- However, the developer must interpret the test results …
Automatic checks good. But also need manual checking...

- Qualitative aspects of websites cannot be checked by automatic tool
- How useful can an alternative text be if it contains nonsense?
The life of a play often extends beyond its performance into live events and drama workshops for schools.

**Events**

The live events are an opportunity for the people to see a writer, actor or director talking about their work; or possibly a chance to hear from the creative or technical teams about their side of a production.

**Workshops**

Workshops are drama workshops for schools based around the themes and texts from individual productions. An invaluable resource for teachers and learners alike, they offer suggestions for running drama workshops in your school.

**Events**

**His Dark Materials**

**Nicholas Hytner**

**Daemons**

**Workshops**

**Henry V**

**A Workshop Approach**

Nicholas Hytner talks about the experience of staging Philip Pullman’s epic trilogy.
Manual checking of ALT tags (Scripting, video content)
The life of a play often extends beyond its performance into live events and drama workshops for schools.

Events
The live events are an opportunity for the people to see a writer, actor or director talking about their work; or possibly a chance to hear from the creative or technical teams about their side of a production.

Workshops
Workshops are drama workshops for schools based around the themes and texts from individual productions. An invaluable resource for teachers and learners alike, they offer suggestions for running drama workshops in your school.

Events
His Dark Materials
Nicholas Hytner
Daemons

Workshops
Henry V

A Workshop Approach

Nicholas Hytner talks about the experience of staging Philip Pullman’s epic trilogy.
Guidelines Testing Tools

- WebXact (formerly Bobby) - no longer available
- Instead try Imergo (www.imergo.com)
- AIS Toolbar (which we will have an exercise with later)
- Other toolbars …
What does the future hold?

- WCAG 2.0
  - 4 principles
  - 13 guidelines
  - 56 guidelines
- Techniques for meeting each guideline provided much as in WCAG 1.0
In depth research on Web accessibility

Helen Petrie
The Disability Rights Commission
Formal Investigation

I had a chance to conduct an in-depth investigation of Web accessibility when my team were appointed to conduct the research for a Formal Investigation of Web accessibility for the Disability Rights Commission.

Accessibility of web sites potentially comes under the Disability Discrimination Act (1995) – although the original act doesn’t mention web sites, the DRC’s Code of Practice (2002) made it clear that provision of information via the Web is a service.
Overview of the FI research

- User Panel of 50 people – blind, partially sighted, deaf, hard of hearing, dyslexic
- Focus groups with User Panel members
- Automated testing of 1000 home pages
- In-depth automated testing and user testing of 100 websites
- Small controlled study with blind and sighted users
- Surveys of Web developers and owners
Sample of 1000 websites

- Took a representative sample of websites of interest and importance to disabled people in Great Britain

- Five main categories:
  - Government and information
  - Businesses (SMEs to multinationals)
  - E-commerce (banking, travel, retail…)
  - Entertainment and leisure
  - Web services (ISPs, portals, search engines, chat rooms …)
Automated testing

- Automated testing of the home pages of the 1000 sites
- Criteria: WCAG 1 - those items in the Guidelines which can be checked automatically

This covers 12 out of 65 Checkpoints

Used WebXM from Watchfire

© H.Petrie, C.Power, G.Weber
Results of automated testing

- 19% of home pages (192) passed the automatic Priority 1 checks, so less than 19% would be fully Priority 1 compliant (WAI A Conformance)

- 51% of government Website home pages passed automatic Priority 1 checks

- Significantly better than the rest of the sample

- No other differences between the sectors
Results of automated testing

- only 6 (0.6%) of all home pages passed Priority 1 + Priority 2 automatic checks
- But only 2 (0.2%) passed both automatic and manual checks at Priority 1 + Priority 2 (AA Conformance)
- No home pages passed Priority 1 + Priority 2 + Priority 3 (even automatic checks) (AAA Conformance)

- No substantial differences between government/other areas on AA and AAA results

Conclusion: Basic technical accessibility very poor
Measuring accessibility in more detail

- also developed two more detailed measures of website accessibility:
  - Designer metric = number of different Checkpoints violated
    (relates to how many different things the designer needs to address)
  - User metric = number of instances of Checkpoint violations
    (relates to the number of problems a user may experience on a page; of course an individual disabled user isn’t affected by all checkpoints and unlikely to read the entire page, but measures the potential number of problems)
Example of the two new metrics

Different WCAG1 Checkpoints violated on a particular page:
- 1.1 provide alt text on images
- 12.1 title each frame

**Designer metric = 2**

Instances of Checkpoint violations on a particular page:
- no alt text on images 10
- frames not titled 3

**User metric = 13**
Results in our data

Designer metric
- a mean of 7.7 different Checkpoints violated per home page
- plus a mean of 39 different Checkpoints warnings

User metric
- a mean of 108 instances of violations per home page
- plus a mean of 239 instances of warnings

Conclusion: no wonder disabled people have problems with the Web
In-depth automated and user testing of 100 Websites

- Selected 100 websites from the 1000 on the basis of a number of measures:
  - the 5 categories
  - use of different Web technologies
  - accessibility level on automated testing
More automated testing

- Automated testing of whole site or the first 500 pages in the site

- In total we have conducted automated testing on 39,000 web pages
User evaluations

- Fairly standard user testing – concurrent verbal protocols
- Because of the amount of data that we wanted to collect in the time available, couldn’t do all the testing with a researcher in the lab
- One session in the lab evaluating 2 – 3 web sites, and teaching participants how to conduct an evaluation on their own
- Each participant evaluated 7 – 8 web sites alone, making a total of 10 evaluations per participant
User evaluations

- target was 1000 tasks = 50 users x 10 sites x 2 tasks

- Each Web site evaluated by at least 5 users, one each from the different user groups

- 913 tasks actually attempted, logged and analysed

- We compared data collected in the lab to data collected alone, and found no differences in effects, although the quantity of data collected when working alone was smaller

Results: Success at tasks

Overall, panel members were successful on (only) 76% of the tasks

but also significant differences between impairment groups:
  - blind successful on 53%
  - partially sighted 76%
  - dyslexic 83%
  - Deaf/hearing impaired/physically impaired 85%
Ease of task ratings

<table>
<thead>
<tr>
<th></th>
<th>Difficult</th>
<th>....</th>
<th>Easy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blind</td>
<td>61%</td>
<td></td>
<td>32%</td>
</tr>
<tr>
<td>Partially sighted</td>
<td>48</td>
<td></td>
<td>51</td>
</tr>
<tr>
<td>Dyslexic</td>
<td>43</td>
<td></td>
<td>53</td>
</tr>
<tr>
<td>Deaf/Hard of hearing</td>
<td>36</td>
<td></td>
<td>63</td>
</tr>
<tr>
<td>Physically Impaired</td>
<td>29</td>
<td></td>
<td>64</td>
</tr>
</tbody>
</table>

So, partially sighted and dyslexic people also disadvantaged, although not as severely as blind people.
Accessibility problems the users encountered

- 585 instances of problems were encountered
- 55% (319) would probably have been avoided if WCAG had been followed, but 45% not (266)
- Guidelines are NECESSARY but not SUFFICIENT – amazing result to the Web community, shouldn’t really surprise the HCI community
Relationship between WCAG conformance and user testing

- We investigated the relationship between the WCAG conformance (total number of checkpoint violations/warnings, number of Priority 1, 2 and 3 violations/warnings, Designer measure, User measure) and the user testing results (success/fail, ratings of ease of use …)

- No significant correlations at all

- This does need further work - another more recent study showed no relationship between user ratings of individual problems and WCAG1 priority levels

The usability bonus

- We conducted a small, conducted study of web sites
  - High accessibility – performed well on WCAG and disabled user testing
  - Low accessibility – performed poorly on WCAG and disabled user testing

- Asked a group of 12 blind people (using screen readers) and a matched group of 12 sighted people to perform a series of standard tasks on the web sites
task completion times

![Bar chart showing task completion times for Blind users and Control Group. The chart compares Low Accessibility and High Accessibility scenarios.]

© H.Petrie, C.Power, G.Weber
Results of this experiment

- Sighted people were 35% faster on the high accessibility web sites than on the low accessibility web sites.

- Analysis of the problems the disabled users encounter will also reveal many/all of the usability problems of the site.

- So, investigating accessibility will yield a usability bonus as well.
Magnification effect

Another interesting effect that we noticed in this study and are now following up. Assistive technologies/disabilities seem to magnify the effects of usability problems. Let me illustrate this with the 2004 Olympics website. So, looking at accessibility problems will highlight usability problems.
Olympic Games 2004 Web site
Automated evaluation

Gerhard Weber
Types of automatic tools

- **Evaluation tools** analyze
  - single static page HTML/XHTML/CSS only
  - multiple static pages (websites) according to possibly multiple sets of criteria

- **Filter tools** and **transformation tools** work as proxy between server and user agent

- **Repair tools** allow to fix HTML immediately

- **Simulation tools** apply features of automatic tools at design time and simulate user perception or user input
Example: LIFT

Integration in Dreamweaver
Informs authors immediately
Some know how required
Evaluation of Automatic Tools

- Selection of 10 automatic evaluation tools
  - Bobby
  - LIFT
  - ...
- Navigation: Keyboard
- Usability issues
- Error reports too elaborated (too many)
- Error reports incomplete (too few)

[Ivory: Automatic Website Evaluation, 2004]
Limitations in automatic tests

- Currently not automatically tested
  - consistency of navigation
  - simple language
- Difficult for dynamic sites in a general sense
  - CMS generated pages may never again be generated (for example eBay)
  - NP-complete
- No other mark-up but HTML
  - Javascript
  - SMIL, TIME (Microsoft)
  - SVG
  - MathsML
  - Multiple namespaces may increase complexity
Exercise: WebXact checks a Webpage

http://www.cs.york.ac.uk/hci/tenuta/index.html
Result of WebXACT

This page does not comply with all of the automatic and manual checkpoints of the W3C Web Content Accessibility Guidelines, and requires repairs and manual verification.

<table>
<thead>
<tr>
<th>Priority</th>
<th>Automatic Checkpoints</th>
<th>Manual Checkpoints</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Status</td>
<td>Errors</td>
</tr>
<tr>
<td>Priority 1</td>
<td>☒</td>
<td>1</td>
</tr>
<tr>
<td>Priority 2</td>
<td>☑</td>
<td>0</td>
</tr>
<tr>
<td>Priority 3</td>
<td>☑</td>
<td>0</td>
</tr>
</tbody>
</table>

Priority 1 Checkpoints

Errors
1 texts, 1 instances on page

1.1 [Provide alternative text for all images]

Warnings
9 texts, 13 instances on page
Checkpoints

Provide alternative text for all images

All images should contain a short alternative text description that represents the function of the graphic. This includes images used as image maps, spacers, bullets in lists, graphical buttons, links, and images used to present math equations. When creating alternative text, use a functional label based on the context in which the image is used, rather than a visual description.

A good test to determine if alternative text is useful is to imagine reading the document aloud over the telephone. What would you say, upon encountering this image, to make the page comprehensible to the listener?

Most authoring tools provide a space for you to enter this information, or you can add alternative text with the "alt" attribute of the IMG element. For example:

```html
<IMG SRC="sailboats.gif" ALT="Our newest model sailboats"/>
```

If descriptive text is already provided above or below the image, an empty description, called "null alt text" (alt=""), may be used in the IMG tag. Providing null alt text, instead of not alt attribute, signals that the image needs no alt text. In other situations where alternative text is unnecessary or distracting, such as images used as spacers, bullets in lists, and links that also contain text, alt="" should still be included so that non-graphical browsers know that they can ignore the image. It is important, however, not to use this technique if the image is in a link or is important to understanding the page.

Rationale

Computers cannot interpret images and present them in a meaningful alternate format; alternative text gives the computer something to present to the user. This is important for users who have turned off image-loading in their web browsers, those using text-based browsers like Lynx, and people who are visually impaired and require the use of a screen reader to read the contents of the screen.

Guideline references

WAI checkpoint 1.10
Section 508, Part 1194.22, Paragraph a

Website analysis by Imergo

- Imergo crawls through a website
- Validates XML structures and can be scripted if server dynamically generates webpages
- (note: ebay won't work very well here as well)
- Imergo applies multiple rule sets (BITV, WCAG, …)
External Rule sets: Nauticus

Completely driven by mark-up language
Reporting Language: EARL

Evaluation and Report Language

Well defined by XML Schema:
- Who has tested (Assertor)
- Which resource was tested (Test Subject)
- Which criteria (Test criterion)
- What was the result (Test Result)

EARL usage
- Comparison of tools
- Integration of test results
- Benchmarking against test suite
- Monitoring of a website
- Used by repair tools, search engine, …
Test Plan

Test design specification

Test case specification

Test procedure specification

Test Execution

Test Log

Test incident report

Test summary report

Test querying and presentation

EARL
Reporting on multiple sites

[Sullivan & Matson, 2000]
Ratio of possible errors and actual errors (n pages)
- Simple calculation
- All accessibility problem identical
- No user groups distinguished

More recent:
- Zeng, 2004 score each problem
- Nietzio, 2006:

\[ A_3(p, u) = 1 - \prod_b (1 - S_{ub})^{C_{pb}} \]

\[ F_s = \frac{1}{n} \sum_{j=1}^{n} F_{pj} \]
Simulating users automatically

aDesigner for web developers [IBM, alphaworks]
Manual evaluation

Chris Power
Checkpoint 1.1 (Priority 1)

- Provide a text equivalent for every non-text element

  AIS:
  - Images → Image List
  - Images → Toggle Image/Alt
  - Images → Show Images
Checkpoint 2.1 (Priority 1)

- Ensure that all information conveyed with colour is also available without colour, for example from context or markup

  AIS:
  - Colour $\rightarrow$ Greyscale
  - Colour $\rightarrow$ Vischeck Colour Blindness Simulator
  - Tools $\rightarrow$ Simulations
Checkpoint 2.2 (Priority 2/3)

- Using the same tools as 2.1
  - Ensure that foreground and background colour combinations provide sufficient contrast when viewed by someone having colour deficits or when viewed on a black and white screen
Checkpoint 4.1 (Priority 1)

- Clearly identify changes in the natural language of a document's text

  AIS:
  - Doc Info ➔ Show Lang Attributes
Checkpoint 6.1 (Priority 1)

- Organize documents so they may be read without stylesheets

AIS:
- CSS → Disable CSS
- CSS → Show Style Sheet
- CSS → Test Styles
- IE Options → Toggle CSS
Checkpoint 3.3 (Priority 2)

- Related to 6.1
- Use style sheets to control layout and presentation
Checkpoint 6.2 (Priority 1)

- Ensure that equivalents for dynamic content are updated when the dynamic content changes

AIS:
- IE Options → Toggle Javascript
- IE Options → Toggle ActiveX
Checkpoint 14.1 (Priority 1)

- Use the clearest and simplest language appropriate for a site's content

AIS:
  - Tools → Juicy Studio Tools Readability Test
Checkpoint 5.1 (Priority 1)

- For data tables, identify row and column headers
- AIS:
  - Structure → Simple data table
  - Structure → Complex data table
  - Tools → Juicy Studio Tools → Table Inspector
Checkpoint 5.4 (Priority 2)

- Related to 5.1
- If a table is used for layout, do not use any structural markup for the purpose of visual formatting
Checkpoint 3.4 (Priority 2)

- Use relative rather than absolute units in markup language attribute values and style sheet property values

AIS:
- Tools → Juicy Studio Tools CSS Accessibility Analyser
- Resize → 640 X 480
- Resize → 800 X 600
Use header elements to convey document structure and use them according to specification

AIS:
- Structure $\rightarrow$ Headings
- Structure $\rightarrow$ Heading Structure
Checkpoint 3.6 (Priority 2)

- Mark up lists and list items properly
  AIS:
  - Structure → List Items
Checking with assistive technologies

- You can also check your own site with assistive technologies - particularly a screen reader such as JAWS

- You can use the free demonstration version (marginal legality) – runs for 40 minutes, then machine needs to re-boot


- But remember that it’s a very complex program, you may not use it the way an experienced user will

- The example with tables that we had – code was fine, but developer didn’t know how to use JAWS
User-based evaluation

Helen Petrie
Key checkpoints

- In our studies, we have found that 15 checkpoints account for a very large proportion of problems users encounter (e.g. over 80% of problems in the DRC study)

- So, if you want a priority list, this is a reasonably good place to start
Key checkpoints: Priority 1

- 1.1: Provide a text equivalent for every non-text element
- 6.3: Ensure that pages are usable when scripts, applets or other programmatic objects are turned off or not supported
- 14.1: Use the clearest and simplest language appropriate for a site’s content
Key checkpoints: Priority 2

- 3.4: Ensure that text size values are relative rather than absolute
- 7.3: Until user agents allow users to freeze moving content, avoid movement in pages
- 10.1: Unit user agents allow users to turn off spawned windows, do not cause pop-ups or other windows to appear and do not change the current window without informing the users
Key checkpoints: Priority 2

- 12.3: Divide large blocks of information into more manageable groups where natural and appropriate
- 13.1: Clearly identify the target of each link
- 13.4: Use navigational mechanisms in a consistent manner
Key checkpoints: Priority 2/3

2.2 Ensure that foreground and background colour combinations provide sufficient contrast when viewed by someone having colour deficits or when viewed on a black and white screen

(Priority 2 for images, Priority 3 for text)
Model of involving disabled users: double iteration

- Initial designs
- Small scale evaluations
- Integration
- Large scale evaluations
Testing with disabled users

- From the DRC research we found that asking:
  - 3 blind people (if possible with different screen readers,
  - 3 partially sighted people (at least one with a screen magnification program, one who uses browser setting to change the appearance of the site) and
  - 3 severely dyslexic people
to test a site will be very likely to show most accessibility problems

- Ask people to do real tasks that you would expect real users to do - what do you want people to be able to do on your website

- NOT - have a look at it and tell me what you think

- Recruiting people is not easy, need people who are honest about the site, articulate their experience
Testing with disabled users

- Ask people to produce a “think aloud” protocol while they are doing the tasks (you will learn a lot) and rate the severity of the problems they encounter.

- “think aloud” = talk you through what they are doing, what problems they are encountering etc.

- Severity ratings:
  4 = Accessibility catastrophe, I can’t proceed with the task
  3 = major accessibility problem, I’m having great difficulty
  2 = minor accessibility problem, it’s causing me some problems
  1 = irritation only, I could live with this
Testing with disabled users

- Can be very useful to audio/video record the test sessions
- Can then carefully study the problems that arise
- Recording and analysis software such as Morae very useful (but expensive) but not essential
Useful Links

Our website of accessibility resources: www.cs.york.ac.uk/hci/tenuta

Web Accessibility Initiative: www.w3.org/WAI

WCAG Version 1/2:
- www.w3.org/WAI/intro/wcag10docs.php
- www.w3.org/WAI/intro/wcag20.php


WebExact: www.watchfire.com
Useful Tools: Mozilla/Firefox

Firefox Web Developer Toolbar:
- addons.mozilla.org/firefox/60/

Extensions for Mozilla/Firefox:
- firefox.cita.uiuc.edu/index.php
Useful Tools: IE

AIS toolbar:
• www.visionaustralia.org.au/ais/toolbar

IE Web Accessibility Toolbar:
• www.visionaustralia.org.au/ais/toolbar/

IE Developer Toolbar:
Useful Tools: Opera

Opera Web Accessibility Toolbar:
• www.paciellogroup.com/resources/wat-about.html

Opera Developer Tools:
• dev.opera.com/articles/view/opera-developer-tools/