An abstract graphic consisting of several overlapping, concentric-like circular bands in various shades of blue, ranging from a deep navy blue to a very light, almost white cyan. The bands are slightly offset from each other, creating a sense of depth and movement. The overall shape is roughly circular, filling most of the frame.

# How to meet the **Web Content Accessibility Guidelines 2.0**

Luke McGrath



An abstract graphic consisting of several overlapping, concentric-like circular bands in various shades of blue, ranging from a deep navy blue to a light sky blue. These bands are set against a plain white background, creating a sense of depth and movement. The overall shape is roughly circular, filling most of the frame.

# How to meet the **Web Content Accessibility Guidelines 2.0**

Luke McGrath

Copyright © Luke McGrath 2012.

All rights reserved, contact for permissions regarding extracts and reprints.

**Web:** [www.lukemcg.com](http://www.lukemcg.com)

**Twitter:** [@lukejmcgrath](https://twitter.com/lukejmcgrath)

**Profile:** [Google+](#)

**Cover designer** Jessica Horton ([www.jessicahorton.co.uk](http://www.jessicahorton.co.uk))

**Proofreader** Gareth Watkins ([watkinseditorial@gmail.com](mailto:watkinseditorial@gmail.com))

### **Notice of Liability**

The information presented in this book is given without warranty. While every precaution has been taken in the publication of this book, the author shall not accept liability to any person or entity with respect to any loss or damage caused directly or indirectly by the advice given in this book.

### **Associations**

Within this book, the author has provided links to useful websites and online tools.

The author has no relationship with any of the companies named and cannot be held responsible for the content of third party websites. He recommends you keep your anti-virus software up-to-date.

### **A Note about Bass**

Throughout this book references are made to bass fishing websites. The names and URLs of these websites were chosen at random for use as examples where useful. At the time of writing none of the examples returns a live website. As with other companies, the author has no association with the bass websites and cannot be held responsible for the content of third party websites.

First published 2012

This edition 2012

# How to meet the **Web Content Accessibility Guidelines 2.0**

Luke McGrath



# Contents

---

<b>Disability and the Internet .....</b>	<b>10</b>
<b>How to use this guide .....</b>	<b>14</b>
<b>Principle One – Perceivable .....</b>	<b>18</b>
<b>Guideline 1.1 – Text Alternatives .....</b>	<b>19</b>
1.1.1 – Text Alternatives .....	19
1.2.1 – Audio-only and Video-only (Pre-recorded) .....	21
1.2.2 – Captions (Pre-recorded) .....	22
1.2.3 – Audio Description or Media Alternative (Pre-recorded) .....	23
1.2.4 – Captions (Live) .....	24
1.2.5 – Audio Description (Pre-recorded) .....	25
1.2.6 – Sign Language (Pre-recorded) .....	26
1.2.7 – Extended Audio Description (Pre-recorded) .....	27
1.2.8 – Media Alternative (Pre-recorded) .....	28
1.2.9 – Audio-only (Live) .....	29

**Guideline 1.3 – Adaptable**..... **30**

1.3.1 – Info and relationships..... 30

1.3.2 – Meaningful Sequence..... 32

1.3.3 – Sensory Characteristics..... 33

**Guideline 1.4 – Distinguishable**..... **34**

1.4.1 – Use of Colour..... 34

1.4.2 – Audio Control..... 35

1.4.3 – Contrast (Minimum)..... 36

1.4.4 – Resize Text..... 37

1.4.5 – Images of Text..... 38

1.4.6 – Contrast (Enhanced)..... 39

1.4.7 – Low or No Background Audio..... 40

1.4.8 – Visual Presentation..... 41

1.4.9 – Images of Text (No Exception)..... 43

**Principle Two – Operable**..... **44**

**Guideline 2.1 – Keyboard Accessible**..... **44**

2.1.1 – Keyboard..... 44

2.1.2 – No Keyboard Trap..... 45

2.1.3 – Keyboard (No Exception)..... 46

**Guideline 2.2 – Enough Time**..... **47**

2.2.1 – Timing Adjustable..... 47

2.2.2 – Pause, Stop, Hide..... 48

2.2.3 – No Timing..... 49

2.2.4 – Interruptions..... 50

2.2.5 – Re-authenticating..... 52

**Guideline 2.3 – Seizures..... 53**

2.3.1 – Three Flashes or Below..... 53

2.3.2 – Three Flashes..... 54

**Guideline 2.4 – Navigable..... 55**

2.4.1 – Bypass Blocks..... 55

2.4.2 – Page Titled..... 56

2.4.3 – Focus Order..... 57

2.4.4 – Link Purpose (In Context)..... 58

2.4.5 – Multiple Ways..... 60

2.4.6 – Headings and Labels..... 61

2.4.7 – Focus Visible..... 62

2.4.8 – Location..... 63

2.4.9 – Link Purpose (Link Only)..... 64

2.4.10 – Section Headings..... 66

**Principle Three – Understandable..... 68**

**Guideline 3.1 – Readable..... 68**

3.1.1 – Language of Page..... 69

3.1.2 – Language of Parts..... 70

3.1.3 – Unusual Words..... 71

3.1.4 – Abbreviations..... 73

3.1.5 – Reading Level..... 75

3.1.6 – Pronunciation..... 77



**Guideline 3.2 – Predictable**..... **78**

3.2.1 – On Focus..... 78

3.2.2 – On Input..... 79

3.2.3 – Consistent Navigation..... 80

3.2.4 – Consistent Identification..... 81

3.2.5 – Change on Request..... 82

**Guideline 3.3 – Input Assistance**..... **84**

3.3.1 – Error Identification..... 84

3.3.2 – Labels or Instructions..... 85

3.3.3 – Error Suggestion..... 86

3.3.4 – Error Prevention (Legal, Financial Data)..... 87

3.3.5 – Help..... 88

3.3.6 – Error Prevention (All)..... 89

**Principle Four – Robust**..... **90**

**Guideline 4.1 – Compatible**..... **91**

4.1.1 – Parsing..... 91

4.1.2 – Name, Role, Value..... 92

**What to do next**..... **94**

**What’s wrong with the Web Content Accessibility Guidelines 2.0?**..... **97**

**Level A Checklist**..... **102**

**Level AA Checklist**..... **104**

**Level AAA Checklist**..... **106**



# Disability and the Internet

---

**B**efore we delve into the murky and often confusing world of web accessibility, spare a thought for the brilliant advances the World Wide Web has already made. Content placed on the internet has the potential to be more accessible than any physical content could ever hope to be.

Think about it; can you see *The Times* bringing out a Braille version every day? Or an mp3 of every single story? Of course not, but by putting content on the internet, it can be read by assistive technologies designed for all types of disabilities. The internet, if used to its full potential, can be a wholly inclusive tool for sharing information and services.

## What is Disability?

When I use the term “disability” with regards to the internet, I really mean four different things:

- Sight disability – for example, blind, partially sighted or colour blind
- Hearing disability – for example, deaf or hard of hearing
- Motor disability – for example, problems with motor skills or slow movement
- Cognitive disability – for example, learning difficulties or problems with concentration or reading

This guide does not seek to define disability in any stronger terms than is useful to making a website accessible to all users. It does not suggest that all disabilities fall into these categories or that they are mutually exclusive.

## Accessibility and Design

Building an accessible website (or modifying an existing website) does not have to be contrary to building a good-looking website. Too many designers take the easy way out when it comes to accessibility and simply strip out features rather than adapting them. This guide will explain ways in which websites can conform to the Web Content Accessibility Guidelines 2.0 (WCAG2) without compromising on design or content.

One of the great strengths of the internet is its diversity, yet web designers can forget that disabled users value this just as much as anyone else. Making websites accessible is not an exercise in making all websites look the same<sup>1</sup>. If you're interested in accessibility (and I assume you are if you're still reading), then you've probably seen so-called "accessible" websites and are familiar with their incredibly "texty" interface<sup>2</sup>. Images are often the first to go during an accessibility exercise, yet only sight-disabled users have a major problem with them. You'll see in the first chapter of this guide that there are much better options than avoiding images altogether and that good images can really help users with a cognitive disability.

## Accessibility and Search Engine Optimisation

Search Engine Optimisation (SEO) is the practice of making a website as attractive to search engines like Google and Bing as possible. Websites with good SEO will appear at the top of search results for keywords associates with your business.

SEO and accessibility go hand in hand because search engines are a lot like disabled users. They can only read certain content, can't listen to or watch media and don't use a mouse. Their understanding of a website is based on very limited features and they do not have the capacity to reason.

One of the most powerful elements of SEO is creating good, machine readable content. This is the same as content that can be read by humans and assistive technologies like screen readers. The days of 'stuffing' content full of keywords to rank higher have gone. Search engines now look for in-depth content with semantically linked keywords that appear more naturally. Good descriptive alternative text for images helps both search engines and disabled users.

Many other areas overlap that I won't cover in detail here, but include descriptive page titles; descriptive link text; logical structure, text transcripts and readable URLs.

## Accessibility and the Law

Accessibility takes on different forms across different territories but most have some kind of anti-discrimination law that prevents discrimination based on ability. In the US, there is the relatively well-known Section 508 that applies to government websites as well as national anti-discrimination law. In the UK we have the Equality Act 2010 that requires equal access to public or private services. There are many [international accessibility laws](#) that apply in different territories and can be mapped to the WCAG2.

The extent to which online anti-discrimination law is enforced is minimal as things stand. It's incredibly hard to see a time when the police will have the resources to take every blogger to court on the basis that their 'service' is not accessible to disabled users. However, if you are

---

1. <http://www.daii.org/> is a great example of an interesting but accessible website (accessed 21/12/11)

2. <http://www.w3.org/> is itself a prime example (accessed 21/12/11)

a relatively successful business (or hope to be), it's not so hard to imagine a letter coming through the door requiring you to bring your website into line with the law. As the case for accessibility on the internet grows (and I'll make sure it does) it is a matter of time before the biggest businesses get an uncomfortable tap on the shoulder. Logically, the same must be true for local councils, government departments, schools and universities.

Without naming any names, try Googling any famous online shop. Chances are that they have no accessibility measures in place – not hard to argue that they are discriminating by effectively blocking some disabled users. With the UK government pledging to get the whole of Britain on super-fast broadband before the Olympics<sup>3</sup>, it's easy to see that a progressive stance on web content is important.

Take the recent introduction of the 'cookie law'<sup>4</sup> as another example. Websites in Europe are now required to have a strategy in place to prompt users to accept cookies before they use any. The confusion this caused has set the enforcement of the law in the United Kingdom back until 2012, a full year after it was intended to take effect. Though only the most prominent websites will be targeted at first, every business should have a plan for change. What would happen to your website if a Web Accessibility Act was enacted tomorrow? Would you have to take it offline for weeks, trawling through every page and fixing errors all while losing custom?

Planning for increased accessibility isn't just a legal requirement; it is a moral imperative and makes business sense. Discrimination is illegal in many countries because it is evil; doing your bit now shows that you care about all of your users, without being forced to. As an added bonus, think of how your online customer base could grow. Say you run BassFishing.com and your main competitor is BassMasters.com, you have an equal market share. One day you announce that BassFishing.com is now built for accessibility. Not only do you get new custom from disabled users but the moral high ground will attract some of your competitor's customers over too (why do you think some coffee shops offer Fairtrade coffee?). It may not be long before BassMasters.com follows suit, but you will always be the website that moved first (the fact that disabled bass fishers have two accessible websites to choose from is a bonus for the rest of us). This really is a case of leading and the rest will have to follow.

## What are the Web Content Accessibility Guidelines 2.0?

While there isn't a specific law that defines web accessibility (and there may never be), the WCAG2 are the closest thing we have to a set of rules. Compiled by the World Wide Web Consortium (W3C), the guidelines are based around four principles of Perceivable, Operable, Understandable and Robust.

Their merits have been long discussed (and you can read my criticisms at the end of this guide) but the WCAG2 are the most in-depth standards we have to measure accessibility

---

3. <http://www.guardian.co.uk/technology/2010/jan/07/broadband-digital-britain> (accessed 21/12/11)

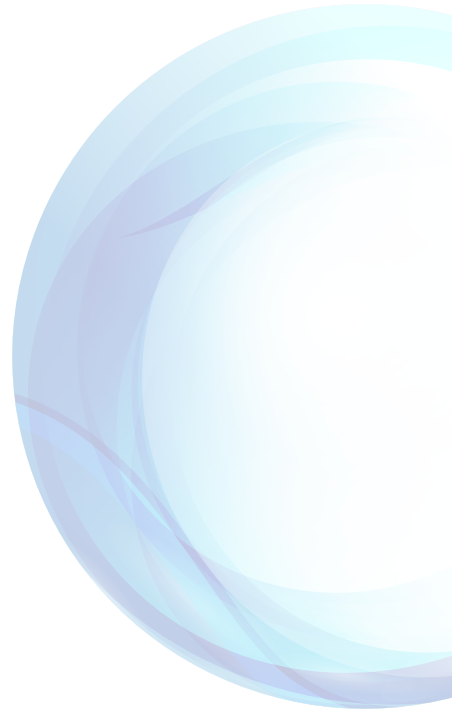
4. <http://www.bbc.co.uk/news/technology-13541250> (accessed 21/12/11)

against. The W3C has developed three levels of accessibility that disabled users recognise as a way of determining whether they can fully use a website. The levels themselves are staggered from A to AAA:

- **Level A** is a standard that websites must achieve for even the most basic accessibility
- **Level AA** is a standard that websites should achieve for good accessibility
- **Level AAA** is the highest form of accessibility but cannot be achieved by all websites and should not be aimed for by many

The guidelines themselves are often hard to understand and worse to implement. They are written in technical jargon and mix techniques that are outdated, not supported or mutually exclusive.

A quick note for American readers; complying with Level AA equates to Section 508 compliance.



# How to use this Guide

---

**T**he intent behind this guide is to simplify the process of making your website accessible to WCAG2 standards. I don't pretend that the WCAG2 are perfect, nor that they are the only suggestions to consider when designing for accessibility, merely that they are the most widely-used and recognised.

You can use this guide whether you are designing a new website, updating or auditing an existing website, or just doing some groundwork before you hire someone to do one of the other two things for you. In fact, if you're in that final category, I hope this guide shows you that you might well be able to make the changes yourself (at the very least you'll be able to demand a much higher standard of work from your designer).

To get the most out of this guide, ignore all of the text in the middle and head straight to the checklists. Wading through every guideline is as dull as it is pointless, unless you know what your problems are. Start at 1.1.1 and work forwards, making notes on what affects your website and what doesn't. Don't have any video content? Tick off all the guidelines that don't apply with a big smile on your face.

Use the checklists as the backbone of your accessibility strategy. Decide which level of accessibility you want to aim for (I'd suggest AA with any AAA that seem reasonable to you) and make your way progressively through the guidelines that affect your website. Where one or more guidelines overlap, or I've spotted a shortcut, you'll find a list of linked guidelines to speed you on your way.

If you want more information about adding accessibility to your regular workflow, make sure you read the Planning for Accessibility section below.

One final thought: pay close attention to the *Issues* section of guidelines you are adopting. As exciting as ticking off every single requirement can be, you must always ask whether it is truly appropriate to your disabled users. In this section I highlight problems that I can see with guidelines, whether they are too strict or too lenient, whether they might lead to later

issues and whether they might soon be devalued in their utility. Pay close attention and formulate an opinion based on knowledge of your website and your customers.

### What this Guide isn't

Following this guide is not a guarantee that your website meets the WCAG2. Neither is this guide an exhaustive list of all the issues that can arise from accessibly designing a website. The great strength of the internet and website design is the almost limitless amount of options open and accessibility does not need to put a lid on that possibility. One thing I will repeat in the guide is the importance of actually testing a website with disabled web users and asking for their feedback.

### What this Guide is

This guide is my interpretation of the WCAG2, it is written in good faith that following its techniques can enable a website to gain a level of accessibility recognised by the W3C.

This guide attempts to explain the WCAG2 by using simple English and straightforward examples. I do not reproduce every technique the WCAG2 suggest as some are outdated or too obscure. Instead, I explain the simplest and most design-friendly techniques that have forward compatibility with HTML5 and CSS.

One of the biggest differences you will find is that I have left out all references to providing an alternative version of a webpage in order to comply. I do not believe that the answer to web accessibility lies in duplicating websites. All websites have the capacity to comply with all guidelines (well, except the impossible ones – more of that later) without doubling website owners' workload and without further marginalising disabled users with ugly, featureless "alternate" versions for their own good. I find this approach lazy, disrespectful and a complete waste of time – the antipathy of good web design.

### How this Guide Works

I have done my best to dissect, explain and advise on each of the WCAG2's guidelines. Each guideline is given a full page or more, made up of:

- An explanation of why the guideline exists
- A *What this means* section that explains what you must do to meet the guideline
- A set of *Tips* for maximising accessibility
- Any relevant *HTML* you will need
- Any *Issues* that I have with a particular guideline.
- A *See also* section so you can easily find related guidelines
- A statement of the guideline's level (A, AA or AAA)

### Planning for Accessibility

Implementing all this accessibility into either a new or existing website isn't going to be easy (it won't be as hard as you might think, it just needs a little effort). Don't suddenly throw all your resources behind converting your website in a day or two, make a plan of small,

incremental changes that you can reasonably stick to.

Remember, the goal is to make an accessible website for the benefit of disabled users. Far better to take your time and get it right than any kind of rush job where no-one will benefit from your hard work. The next two sections should give you a rough idea of how to plan your changes.

## Designing a New Website

Starting from scratch? What luck! You're taking the easiest route to web accessibility.

Whether designing yourself or contracting out, make sure that you:

- Decide which level of accessibility you want to achieve
- Compare the types of content you are planning to the checklists at the back of this guide – tick off the ones that don't apply
- Start with your template (the header, navigation and footer) – get these right once and you're most of the way there
- Add static pages one at a time, checking them for compatibility with the level you have chosen
- Think about the content you will be adding to your website over time – draw up a set of rules for yourself/your writers so that all new pages meet the accessibility level you want
- Put in place an editing stage so that someone else is checking pages for accessibility before they are published

## Auditing an Existing Website

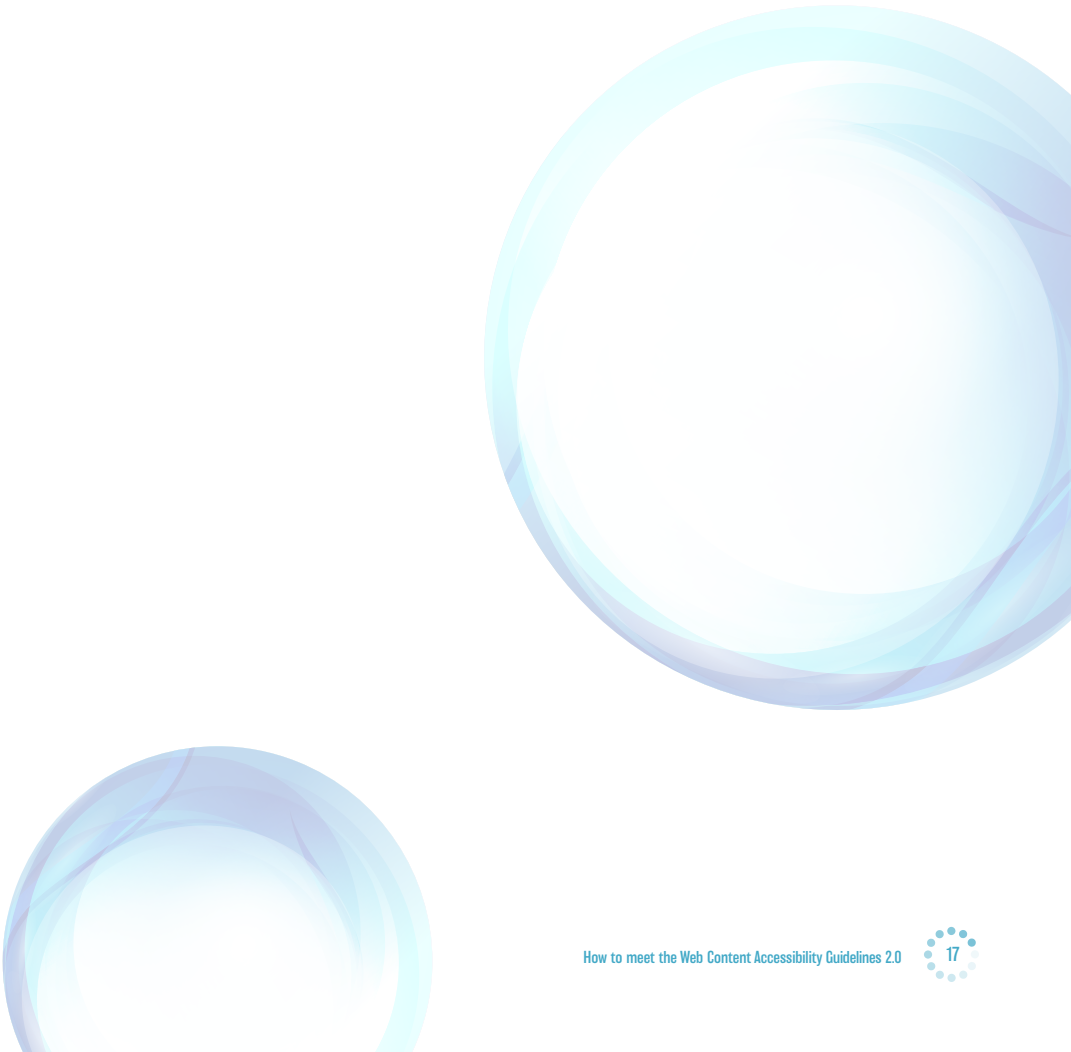
Not as easy as starting from scratch but achievable with good planning. In this case there may be parts of your website that cannot meet some accessibility criteria, it's up to you whether to remove or replace these. Either way, ensure that you:

- Decide which level of accessibility you want to achieve
- Start with your template (the header, navigation and footer) – do you need to change your design or can you amend existing elements?
- Draw up a list of your pages and rank them in order of importance – a combination of most visited and value to your business is best
- Draw up a reasonable time frame to edit these pages – one a day, one a week, etc... depending on your resources
- Write up a set of rules for yourself/your writers so that all new pages meet the accessibility level you want
- Put in place an editing stage so that someone else is checking pages for accessibility before they are published

## Here we go

You've read the waffle, now it's time to let loose on that website of yours. Remember, take things slow and make full use of the checklists. Above all, have fun with the process and make an awesome website that everyone can enjoy.





# 1

---

## Principle One – Perceivable

**“Information and user interface components must be presentable to users in a way they can perceive.”**

### What this means

Disabled users read websites in different ways, often making use of assistive technology (for example, screen readers, text-only browsers and Braille). All content on a website should be understandable by these methods.

This is really the most basic form of accessibility; making sure that content can be read and understood. There are, of course, several layers to claiming full accessibility under Principle One, but the first level is relatively easy to achieve.

### Tips

One of the most important things to take into account when following the WCAG2 is the availability of assistive technology. Many disabled users will visit your website using assistive technology and interact with your webpages in ways you might not expect. Throughout this guide I will highlight the best methods for including such technologies in your design.

A few popular assistive technologies that you might want to research are:

- [BrowseAloud](#)
- [JAWS \(Job Access With Speech\)](#)
- [Lynx](#)
- [Microsoft Magnifier](#)
- [Window-Eyes](#)

It will be worth your time to test your website with a few of these products as you go. While the techniques in this guide are designed to satisfy WCAG2 criteria, there is no substitute for proper testing. If possible, find a group of disabled users and ask them to try out your website – a local college or night class on basic computer skills might be a good place to ask. Use [Next Step's course finder](#) to search for courses online.

## Guideline 1.1 – Text Alternatives

### 1.1.1 – Text Alternatives

Some disabled users browse websites with images turned off (as do some users with slow internet connections). You must provide text alternatives for non-text content (for example, images and form controls). These must be true alternatives to the content – they must provide the same information.

#### What this means

- Add descriptive alt tags to all images (for example, “fisherman holding a bass”)
- Add the same to media such as audio or video (you do not have to describe the whole video: “video about bass fishing” will do)
- Add the same to any icons (for example, RSS feed, print or sharing icons)
- Add a name to any buttons you use (for example, “Search” next to a search box)

#### Except

- Purely decorative text
- If the non-text content is a test that would be undermined by providing a description
- If the non-text is a CAPTCHA image

#### Tips

- If you do use CAPTCHA, consider providing contact details for someone who can override the process as some disabled users cannot pass this kind of test
- Where alt text is not required, use a null alt attribute (see below)

#### HTML

A correctly inserted image has two required attributes: source and alternate text. The source is the URL of the image and the alternate text is what we use for accessibility:

```
➤ 
```

Adding a null alt attribute:

```
➤ 
```

Adding labels and names into a form:

```
➤ <form>
  <input type="radio" name="type" id="bass" />
  <br />
  <label for="notbass">Not bass</label>
  <input type="radio" name="type" id="notbass" />
</form>
```

## Issues

The W3C demand for the “same” information to be included in an alternative text as an image is practically difficult to achieve. For starters, who’s to say what the same is? If I showed you a picture of a sunrise, is it enough to use “sunrise” as alternative text? What if there was a tree in the background of the picture too? Does “sunrise over a tree in the background” work? At what point does a description become too long to be truly useful?

The best bet here is to be honest. If the image has been used to convey a sunrise to regular users, then “sunrise” is what disabled users need to know. If you have a series of different pictures of sunrise, then mark them out as different in the alternative text.

## See Also

1.4.5 – Images of Text  
1.4.9 – Images of Text  
(No Exception)

***Implementing 1.1.1 is a  
Level A standard***

## Guideline 1.2 – Time-based Media

In real life we are talking about audio or video media (it's called "time-based" by the nerds at W3C because the information is only available for the duration of the recording).

Media like this is hard for some disabled users to understand – the key is to provide a range of alternatives.

### 1.2.1 – Audio-only and Video-only (Pre-recorded)

Some disabled users will find it difficult to understand things like podcasts and silent videos or animations. An alternative must be provided that presents the same information as the regular experience.

#### What this means

- Audio-only – provide a link to a full text transcript
- Video-only – provide a link to a full text transcript **or**
- Video-only – provide an audio track

#### Except

If the content is itself an alternative (for example, you don't have to provide a transcript of the audio track you provided to explain the silent video you used)

#### Tips

A full text transcript is a document that includes all information present in the audio or video source. This means including any visual cues (for example, "The fisherman holds up a large bass.") as well as dialogue attribution and description

#### Issues

Already we can see that even gaining Level A can be hard. Many users will simply avoid using media content to pass 1.2.1, but this makes the internet a poorer place.

#### See Also

1.2.3 – Audio Description or Media Alternative (Pre-recorded)  
1.2.5 – Audio Description (Pre-recorded)  
1.2.7 – Extended Audio Description (Pre-recorded)  
1.2.8 – Media Alternative (Pre-recorded)

***Implementing 1.2.1 is a Level A standard***

### 1.2.2 – Captions (Pre-recorded)

When a video does have audio (which is most of the time) you must provide closed captions. YouTube is currently running a beta auto-captioning service but it is not yet good enough to pass. You only have to watch a few videos with the sound off to see the problems – imagine the confusion a deaf user will have when your bass fishing video starts talking about mass killing.

#### What this means

Add handwritten captions to any videos you use

#### Except

If the video itself is an alternative

#### Tips

Google has written a very clear [how-to guide on adding captions to YouTube](#) videos which I'm not going to try and better.

#### Issues

The main issue here is having enough time to insert the captions, especially if you rely on video as a large part of your content. The key is to build the time into your workflow from the start (the same goes for guidelines that require transcribing such as 1.2.1 if you have written a full text transcription then you have the basis for your captions already).

If you feel you can't build in the time, consider cutting the number of videos you upload. One fully-accessible video that all your potential visitors can enjoy is better than two videos that alienate some of your audience.

#### See Also

1.2.1 – Audio-only and Video-only (Pre-recorded)  
1.2.4 – Captions (Live)

***Implementing 1.2.2 is a Level A standard***

### 1.2.3 – Audio Description or Media Alternative (Pre-recorded)

Video with audio must have a second alternative – because closed captions aren't always helpful to a user with sight difficulties.

#### What this means

- Provide a link to a full text transcript (see below) **or**
- Make an alternative version of the video where the regular soundtrack is replaced by one with audio description and link to this from near the regular content

#### Except

- If the video is itself an alternative
- No audio description is needed if all of the information in the video is provided in the regular soundtrack

#### Tips

- Providing audio description at this stage will fulfil 1.2.5 for Level AA but you will need both audio description and a transcript for 1.2.8 and Level AAA
- If you're going to the length of audio description for 1.2.3 you can satisfy 1.2.5 & 1.2.7 by recording extended audio description tracks wherever necessary
- Something like a straight face-to-face interview, or a speech-to-camera would probably not need audio description
- A full text transcript is a document that includes all information present in the audio or video source. This means including any visual cues (for example, "The fisherman holds up a large bass.") as well as dialogue attribution and description.

#### See Also

1.2.5 – Audio Description (Pre-recorded)  
1.2.7 – Extended Audio Description (Pre-recorded)  
1.2.8 – Media Alternative (Pre-recorded)

***Implementing 1.2.3 is a Level A standard***

### 1.2.4 – Captions (Live)

Disabled users should be able to access closed captions to help them understand live audio in videos.

**Implementing 1.2.4 is a Level AA standard**

#### What this means

- Using a content management system that allows captions to be added live and probably a professional subtitler to do it for you
- Live video is mainly used by large broadcasting companies but you might want to provide a live stream of a seminar for people who can't physically attend, or perhaps you're a university and want to put graduation ceremonies online

#### Except

If the video is an alternative for text

#### Tips

Remember, this is only for audio that is also part of visual content – it doesn't apply to live radio-style podcasting

#### Issues

Finding a system and the staff to provide live captions is not going to be easy or cheap. Companies like the BBC employ professionally trained subtitlers for their programmes, including live broadcasts.

As I noted above, many universities now stream live broadcasts of graduation ceremonies. As institutions funded by student fees (and parents' savings), they have a stronger duty than most to provide this function.

Overall, this guideline is aimed at the right level as anyone providing live video has a duty to include deaf users. Broadcasters will probably have the resources to succeed but smaller companies will struggle even if the live content is occasional or a one-off.



### 1.2.5 – Audio Description (Pre-recorded)

Sight-impaired users should have access to audio description for video content.

#### What this means

Make an alternative version of your video with an audio description soundtrack and link to it from near the original content

#### Except

No audio description is needed if all of the information in the video is provided in the regular soundtrack

#### Tips

- By doing this for 1.2.3 you would already have fulfilled 1.2.5
- Something like a straight face-to-face interview, or a speech to-camera would probably not need audio description

#### Issues

It's a shame that audio description only comes in at the second level of accessibility in the W3C's mind. The idea that a full text transcript is useful is something of a fudge when you think about it; blind users don't need it and deaf users have closed captions (1.2.2).

That said, the W3C is really just reflecting a society where closed captions are far more prevalent than audio description (think of television and movies). I can understand that providing an alternative soundtrack to a video is much more onerous than typing out a full text transcript. But we're not doing this because it's easy; we're doing it because we believe accessibility is important. Frankly, audio description is better than full text transcript and we should be guiding society, not accepting it as it is or racing to the bottom.

#### See Also

- 1.2.2 – Captions (Pre-recorded)
- 1.2.3 – Audio Description or Media Alternative (Pre-recorded)
- 1.2.7 – Extended Audio Description (Pre-recorded)
- 1.2.8 – Media Alternative (Pre-recorded)

#### ***Implementing 1.2.5 is a Level AA standard***

### 1.2.6 – Sign Language (Pre-recorded)

You can help some disabled users by providing sign language translations for videos, media content and animations with audio.

**Implementing 1.2.6 is a Level AAA standard**

#### What this means

- Make an alternative version (linked from near the original) with a sign language interpreter embedded in the corner of the video – you'll need some serious video editing software and knowledge for this (and someone who can sign) **or**
- You could make a video of just the sign language translation and place it next to the original content (sadly, most browsers would not yet support having two videos running simultaneously)

#### Issues

The most obvious issue is the question of which sign language to use. Forms of sign language are often exclusive to a particular country; even British and American sign languages are completely distinct, despite the similarities of the written language.

The best bet is to provide sign language in the language of the country that you are targeting, or if you are multi-national, in the language of the country of the highest proportion of your visitors. Hardly a precise answer.

Secondly, current technology only really allows for one solution to this issue. Providing a video of a sign language translation that runs next to the original content simply will not work most of the time.

It is possible that, in the future, an assistive technology will be invented that reads text and provides a signed version in the user's language of choice.

It almost seems like this guideline is pre-empting websites becoming fully fledged broadcasters. In reality, only large corporations can currently afford the expense of signing every video they provide.

### 1.2.7 – Extended Audio Description (Pre-recorded)

When pauses in a video's audio are too short to provide all the information to partially-sighted users, an extended audio description can be provided.

#### What this means

- Providing an alternative version of the video with a soundtrack where you pause the visual until the audio description track has caught up with the information in the video
- Probably hiring someone with strong editing knowledge to get the timings right and synchronise the audio and video

#### Except

No audio description is needed if all of the information in the video is provided in the regular soundtrack

#### Tips

- If you're going to the length of audio description for 1.2.3 you can satisfy 1.2.5 & 1.2.7 by recording extended audio description tracks wherever necessary
- Something like a straight face-to-face interview, or a speech-to-camera would probably not need audio description

#### See Also

1.2.3 – Audio Description or Media Alternative (Pre-recorded)  
1.2.5 – Audio Description (Pre-recorded)  
1.2.8 – Media Alternative (Pre-recorded)

***Implementing 1.2.7 is a Level AAA standard***

### 1.2.8 – Media Alternative (Pre-recorded)

Video-only content (video, media content or animation without sound) is inaccessible to some disabled users. You can provide a full text transcript for this type of media as well as ensuring that media with video and sound has the same alternative.

#### What this means

Provide a link to a full text transcript (see below) for all video, media content and animation, whether it has audio or not

#### Except

If the video is itself an alternative

#### Tips

- You may have already done this if you chose transcript for 1.2.3 and added an audio description for 1.2.5
- A full text transcript is a document that includes all information present in the audio or video source. This means including any visual cues (for example, “The fisherman holds up a large bass.”) as well as dialogue attribution and description.

#### See Also

1.2.3 – Audio Description or Media Alternative (Pre-recorded)  
1.2.5 – Audio Description (Pre-recorded)  
1.2.7 – Extended Audio Description (Pre-recorded)

***Implementing 1.2.8 is a Level AAA standard***

### 1.2.9 – Audio-only (Live)

You can assist some disabled users by providing an accessible alternative to live audio content. This includes live radio-style podcasts or plays.

#### What this means

- All live content with an audio element is captioned by a professional subtitler and displayed near the original content **or**
- If the audio is live but from a prepared script, the script is made available by a link near the original content

#### See Also

1.2.4 – Captions (Live)

***Implementing 1.2.9 is a Level AAA standard***

## Guideline 1.3 – Adaptable

Some disabled users will need to view a website as simply as possible to understand it. When doing so, they should still have all the information available to regular users.

### 1.3.1 – Info and relationships

One of the ways we understand content is by its structure or format. For example, you probably understand that the large text at the top of this page is a heading, and that smaller text is a subheading, and the rest is the main content.

We also understand content by knowing what symbols like “ ” and bullet points mean, how tables work, and that **bold** and *italics* give emphasis (and that ***bold italics*** is too much). Some disabled users may not understand these elements; they must be able to determine them from a webpage’s HTML markup so that screen readers can interpret correctly.

Forms can be just as complex to understand; how many times have you seen one that uses an asterisk for required fields without explaining what it means?

#### What this means

- Break up content with headings **and**
- Use header tags to determine the relative importance of different headings **and**
- Use correct HTML markup to denote paragraphs, lists, tables, emphasised text and quotes **and**
- Ensure markup is valid (no odd <p>s floating around) **and**
- Use clear labels on forms (for example, a statement before a form that specifies “All required fields are marked in red and with an asterisk (\*)”) **or**
- Add an icon with a text alternative next to required fields and explain this above the form

#### Tips

Use the W3C’s HTML validator to check your markup (but checking by eye is best)

## HTML

Header tags go from:

► `<h1>The Most Important Header</h1>`

to

► `<h6> The Least Important Header</h6>`

W3schools' [full list of valid HTML5](#)

The W3C's [HTML validator](#)

### See Also

1.3.2 – Meaningful Sequence  
2.4.3 – Focus Order  
2.4.6 – Headings and Labels  
2.4.10 – Section Headings  
4.1.1 – Parsing

***Implementing 1.3.1 is a Level A standard***

### 1.3.2 – Meaningful Sequence

The meaning of a text is usually reliant on the order it is presented. For example, in English a left-hand column is read before a right-hand column. Content must be presented in a way that preserves meaning for disabled users using assistive technology.

#### What this means

- Text is always presented in a meaningful sequence
- Correct HTML markup preserves meaningful sequence (for example, all the elements of an image are nested together)
- Separate navigation menus from content with CSS

#### Except

Where the meaning of content is not affected by its sequence (for example, an unordered list when the order of items is irrelevant)

#### Tips

Use header tags in descending order (H1 > H6) to show the hierarchy of content

#### HTML

Header tags go from:

► `<h1>The Most Important Header</h1>`

to

► `<h6> The Least Important Header</h6>`

#### See Also

1.3.1 – Info and Relationships  
2.4.3 – Focus Order  
2.4.6 – Headings and Labels

***Implementing 1.3.2 is a Level A standard***



### 1.3.3 – Sensory Characteristics

Some disabled users may not understand references to shape, position or size. For example, how many times have you seen a website say “Use the search box on the right” – what if you don’t know where “right” is? You must use more than one indicator when giving instructions.

#### What this means

- Instructions use more than one indicator – using location, colour, shape and clear labelling is best. (for example, “Search by using the green rectangle box labelled ‘Search’ on the right.”)
- Instructions do not rely on sound (for example, a beep sound for mistakes on a form)

#### Tips

Using colour correctly in instructions fulfils 1.4.1

#### See Also

1.4.1 – Use of Color

***Implementing 1.3.3 is a Level A standard***

## Guideline 1.4 – Distinguishable

Visually impaired users may need extra help to understand websites; they may need to zoom in on text or have a higher contrast between content and background colours.

### 1.4.1 – Use of Colour

Colour must not be the only means of identification – “Required fields are in red.” on a form is no use to a user who cannot understand “red”.

#### What this means

- You should already have detailed descriptions for instructions from 1.3.3
- If you haven’t, think: label, colour, size, shape, position
- If you use a graphic element on a webpage (for example, a chart), ensure that different areas are indicated by more than just colour (for example, add different patterns to the bars on a bar chart)

#### Tips

Print off a webpage in black and white and make sure you can still understand all the information

#### See Also

1.3.3 – Sensory Characteristics

***Implementing 1.4.1 is a Level A standard***

### 1.4.2 – Audio Control

Some disabled users have difficulty focusing on content. Adding distractions that play automatically can undermine their experience of a website, especially if they rely on screen reader technology. Any potential audio distraction must have options to be stopped or paused.

#### What this means

- Any audio that plays automatically when a page loads has a stop or pause function **or**
- Any audio that plays automatically when a page loads has a volume control or mute function

#### Except

The audio plays for three seconds or less

#### Tips

Ideally, don't add items like this to websites because they annoy and distract all users (especially me)

**Implementing 1.4.2 is a Level A standard**

### 1.4.3 – Contrast (Minimum)

Some users need a large contrast between content and background to understand information fully. You should ensure that the minimum contrast ratio is 4:5:1.

#### What this means

- Use a light background and dark text **or**
- Use a dark background and light text **and**
- Use WebAim's [colour contrast checker](#) to verify your choice

#### Except

- Text that is 18 points upwards (or 14 points upwards if bold) must have a contrast of at least 3:1
- Where text is purely decorative – so you could have a block of different coloured words on a blue background if it didn't do anything and their order was random
- Where the text is an incidental part of an image (for example, a man who is reading a newspaper or a view that happens to include a street sign)
- Brand logos

#### Tips

- Remember to ensure that all colours used conform, this includes links which may change colour after being used once and headings in menus and sidebars as well as main content
- Picking a contrast of at least 7:1 will also fulfil 1.4.6
- This guideline also applies to images of text, such as 1.4.5 and 1.4.9 (but you shouldn't really be using them)
- Make sure that any embedded charts or images of charts have the minimum contrast between elements (for example, bars, axes and labels)

#### See Also

1.4.5 – Images of Text  
1.4.6 – Contrast (Enhanced)  
1.4.9 – Images of Text (No Exception)

***Implementing 1.4.3 is a Level AA standard***

### 1.4.4 – Resize Text

Some sight-impaired users will need to zoom closer to content in order to understand it fully – this should be accomplished up to 200% without losing content or functions.

#### What this means

- As all modern browsers allow for resizing text, a site based on good HTML/CSS should comply **and/or**
- Add a feature to your website that allows the user to change text size (by CSS) based on three or four predetermined options including 200%
- Check your website by resizing up to 200% in a variety of browsers

#### Except

- Images of text (but don't use these because they don't resize well)
- Captions

#### Tips

- Resizing text is different to zooming (which resizes text, images and layout)
- Make sure your resized text doesn't require the user to scroll horizontally and you fulfil part of 1.4.8

#### See Also

1.4.5 – Images of Text  
1.4.8 – Visual Presentation  
1.4.9 – Images of Text (No Exception)

***Implementing 1.4.4 is a Level AA standard***

### 1.4.5 – Images of Text

Users with a text browser cannot process images of text; they may also find it hard to understand images of text that have been resized, as these do not scale well. While using alt tags is useful, you should use plain text when the technology allows.

#### What this means

- Do not use an image of text when you can use text
- Use CSS to style headings
- Use CSS to style navigation menus as text
- Display quotes as text rather than images

#### Except

- Brand logos
- If using an image of text is essential and cannot be achieved with text (for example, presenting a particular example of typography)
- The presentation can't be achieved with the technology used to design the website

#### Tips

- Remember that images of text can be subject to guidelines on contrast (1.4.3 and 1.4.6)
- Remember to add alt tags for any images you do use (see 1.1.1)

#### See Also

- 1.1.1 – Text Alternatives
- 1.4.3 – Contrast (Minimum)
- 1.4.6 – Contrast (Enhanced)
- 1.4.9 – Images of Text (No Exception)

***Implementing 1.4.5 is a Level AA standard***

### 1.4.6 – Contrast (Enhanced)

You can help some disabled users by making your website's contrast as clear as possible – at least 7:1.

#### What this means

- Choose a light background and dark text **or**
- Choose a dark background and light text
- Use WebAim's [colour contrast checker](#) to verify your choice

#### Except

- Text that is 18 points upwards (or 14 points upwards if bold) must have a contrast of at least 4:5:1
- Where text is purely decorative – so you could have a block of different coloured words on a blue background if it didn't do anything and their order was random
- Where the text is an incidental part of an image (for example, a man who is reading a newspaper or a view that happens to include a street sign)
- Brand logos

#### Tips

- Remember to ensure that all colours used conform, this includes links which may change colour when used and headings in menus and sidebars as well as main content
- You may have fulfilled this guideline when you tackled 1.4.3
- This guideline also applies to images of text, such as 1.4.5 and 1.4.9 (but you shouldn't really be using them)
- Make sure that any embedded charts or images of charts have the minimum contrast between elements (for example, bars, axes and labels)

#### See Also

1.4.3 – Contrast (Minimum)  
1.4.5 – Images of Text  
1.4.9 – Images of Text (No Exception)

***Implementing 1.4.6 is a Level AAA standard***

### 1.4.7 – Low or No Background Audio

Users with hearing difficulties can benefit if audio-only content on your website is as clear as possible. This mainly applies to interviews and presentations.

#### What this means

- Pre-recorded audio-only content does not contain any background noise or
- If there has to be some background noise, it is generally 20 decibels lower than the foreground noise

#### Except

- When the audio isn't mainly speech – an action scene, for example
- If the audio is part of a CAPTCHA element
- The audio is you 'singing or rapping' as the W3C nerds put it. They so street

#### Tips

- Record your own audio in a place you know will be quiet (a room with lots of soft furnishings is best if you haven't got a studio set-up)
- Only embed good quality audio clips (you should be doing this anyway if you care about your website)

**Implementing 1.4.7 is a Level AAA standard**



### 1.4.8 – Visual Presentation

Sight-impaired users may have a variety of preferences that best allow them to understand content on a website. There's no way you can guess every combination of colours and sizes they might collectively want, but you can provide a decent range with a few features.

The following five features are cumulative, as all must be in place to pass 1.4.8

#### What this means

1. Provide a tool that enables the user to select from a number of background and foreground colours – a designer familiar with CSS will be able to implement this **and**
2. Text blocks must be no wider than 80 characters – set your template accordingly, making sure that narrowed windows do not require horizontal scrolling (by using CSS) **and**
3. Text is not justified to both sides of the webpage **and**
4. Provide a tool that enables the user to select from a number of line and paragraph spacing options – one of the options must give line spacing that is at least 1.5 in text blocks and spaces between paragraphs are at least 1.5x line spacing (2.25 if using normal word processor standard) – again in CSS - **and**
5. Text can be resized in a browser up to 200% without requiring the user to scroll horizontally on an average monitor or laptop screen (more CSS)

#### Tips

- Use [W3C's in-depth colour picker](#)
- The [BBC's accessibility page](#) has a good range of colour and spacing options

#### Issues

The scale of this guideline means that it can only exist at the AAA level, however many of its features are vitally important to disabled users (especially those with cognitive disabilities, who are poorly supported by the WCAG2) and could happily be separated and moved down a level or two.

1. A colour picker will open up your website to users who simply cannot see regular (black on white) colours. The number of options you need to provide is unclear but a safe bet is to match the W3C's example (except think of

a way to include it the top navigation of your website, up with Skip Navigation if you really want to help – it has to be one of the first things users see to have any use). This also means thinking of a good-looking design for the picker that fits with your website

2. The W3C could have easily made this a lower level, standalone requirement. Having scanned popular websites in the news, sports, tech and entertainment sectors, most websites already come close to this (80-100 characters is common) due to having a thickish sidebar. Making sure that a narrowed window doesn't require scrolling is just good design
3. Again, no reason why this couldn't exist as a lower level, standalone guideline. Justifying text to both sides doesn't look great and many websites already justify to the left
4. One of the worst written guidelines. Text height must be changeable by the user to 150% of the default line spacing and spaces between paragraphs to 150% of the new line spacing... Best bet is to provide options (like in the [BBC accessibility page](#)) of space and a half and double (with their equivalent paragraph spacing of course!). In fairness, it's not easy to see how this can be simplified (would people understand "Provide 1.5x and 2x spacing" to have meant lines and paragraphs?) but adding it in the middle of an already complicated guideline is lunacy
5. This could really have been addressed fully back in 1.4.4

## See Also

1.4.4 – Resize Text

***Implementing 1.4.8 is a Level AAA standard***

### 1.4.9 – Images of Text (No Exception)

You can take care of partially-sighted users by minimising the use of images of text, allowing for them to be interpreted by assistive technologies. This builds on 1.4.5 by removing the excuse of using unsupportive technology for your website.

#### What this means

- If you can't use text for a function then you should forget that function and come up with something that does use text
- Use CSS to style headings
- Use CSS to style navigation menus as text
- Display quotes as text rather than images

#### Except

- Where text is purely decorative – so you could have a block of different coloured words on a blue background if it didn't do anything and their order was random
- Brand logos
- If using an image of text is essential and cannot be achieved with text (for example, presenting a particular example of typography)

#### Tips

- Remember that images of text can be subject to guidelines on contrast (1.4.3 and 1.4.6)
- Remember to add alt tags for any images you do use (see 1.1.1)

#### See Also

1.1.1 – Text Alternatives  
1.4.3 – Contrast (Minimum)  
1.4.5 – Images of Text  
1.4.6 – Contrast (Enhanced)

***Implementing 1.4.9 is a Level AAA standard***

# 2

## Principle Two – Operable

**“User interface components and navigation must be operable”**

### What this means

Disabled users may have a range of difficulties with motor skills (for example, they may navigate by keyboard only). This category also includes older users who may have deteriorating motor skills and who prefer to switch between mouse and keyboard. These users should be able to use all the functions a regular user would.

### Guideline 2.1 – Keyboard Accessible

A keyboard is the most popular accessible form of navigating a website.

#### 2.1.1 – Keyboard

All functions of a website must be operable through a keyboard, without requiring specific timings for keystrokes.

### What this means

- Clean HTML/CSS based websites will generally comply
- Check your website by unplugging the mouse and testing every function
- Make sure no function needs a timing, such as double enter within 2 seconds to submit

### Except

Functions that require pointer input such as free-hand drawing and some games (although 2D keyboard games are loads of fun)

### Tips

Don't use access keys (assigning a navigation link to a particular key) or page-specific key commands as they can conflict with assistive technology

### See Also

2.4.3 – Focus Order  
2.4.7 – Focus Visible

***Implementing 2.1.1 is  
a Level A standard***

## 2.1.2 – No Keyboard Trap

If a function of a website can be reached by a keyboard, it must be possible to move away from that function via a keyboard too. If focus can be moved by non-standard keyboard commands, instructions must be provided.

### What this means

- Test your website to make sure you can navigate away from as well as to functions **and**
- Make sure all navigation is controllable by either tab or arrow keys, which is standard for clean HTML/CSS based websites

### Tips

- Don't make things hard by using non-standard navigation and trying to crowbar in some instructions
- Ensure any third party widgets or plugins that you use comply

### Issues

The W3C doesn't ask for enough with this guideline. All navigation should be accessible by standard keyboard commands (tab, arrow keys). Leaving an exception, even at the lowest level, is a poor choice (made even poorer by the idea that throwing in instructions is a good design or accessible option – where do you put the instructions? how do you fit them in with other guidelines?).

**Implementing 2.1.2 is a Level A standard**

### 2.1.3 – Keyboard (No Exception)

Websites that have an element of pointer-controlled content cannot conform to AAA. You can make all content accessible if you choose to avoid this type of content.

#### What this means

W3C states that 2.1.3 isn't about making sites with pointer-controlled content conform – such sites simply can't be AAA

#### Issues

How this helps disabled users who presumably want to draw or play games I do not know. The W3C is asking for something that would make the internet a far less interesting place, and penalise all kinds of users.

The W3C does, of course, state that:

"It is not recommended that Level AAA conformance be required as a general policy for entire sites because it is not possible to satisfy all Level AAA Success Criteria for some content."<sup>1</sup>

Why try to promote such a self-defeating standard? Games that rely on mouse pointers are, by definition, not suitable for users who cannot use a mouse. It seems insane that a forward-thinking, accessible website designer who makes such games cannot get the highest level. As long as the website itself is perfectly accessible (not an easy task in the first place), how can it then be marked down for providing something specifically designed not to be used by a keyboard?

Disabled users I have met have unanimously agreed on this issue. They understand that there are some things that they cannot take part in. A few mentioned to me that disability away from the internet is far more frustrating than disability using it, not being able to play a certain type of game is par for the course and shouldn't be a black mark against a website that otherwise does everything right.

#### See Also

2.1.1 – Keyboard

***Implementing 2.1.3 is a Level AAA standard***

---

1. <http://www.w3.org/TR/WCAG20/> at Conformance Requirements (accessed 19/12/2011)

## Guideline 2.2 – Enough Time

Some disabled users may need longer to read information on a website than regular users. A time limit on content impedes their ability to understand. A good example of the correct use of time limits is on a banking website to make sure that user details are secure.

### 2.2.1 – Timing Adjustable

When a time limit is in place, one or more options must be available to alter the amount of time given to a user.

#### What this means

- Give the user an option to turn off the limit before it begins – a landing webpage before a secure webpage explaining this and offering the choice would work **or**
- Give the user an option to adjust the time limit before it begins, over a range of at least 10x the default – again, use that landing webpage **or**
- The user has an option to extend the time limit at least 20 seconds before it expires with a simple action (like pressing enter) and this function can be used at least 10 times
- A site with a moving text animation must have a pause button so slower readers can understand the information
- A site with a function that automatically updates with the latest headlines has an option to delay the time between updates 10x

#### Except

- If the time limit is due to real time events – like bidding on eBay or buying an offer on Groupon
- Live video streams
- The time limit is essential – for example, when buying tickets for a concert you may only have two minutes to confirm your order but demand for tickets is high so the limit is necessary for the business to function
- If the time limit is greater than twenty hours

#### Tips

- Take as much content outside of time limits as possible
- Make sure all these options are keyboard accessible!
- If you use a pop-up that gives an option to extend a time limit, consider 2.2.4

## See Also

2.2.3 – No Timing  
2.2.4 – Interruptions  
2.2.5 – Re-authenticating

***Implementing 2.2.1 is a Level A standard***

### 2.2.2 – Pause, Stop, Hide

Moving, blinking, scrolling and auto-updating information can be difficult for some disabled users to understand. They must be able to pause the movement in order to fully read the content.

#### What this means

- Moving, blinking or scrolling information must have an option to pause or stop or hide it
- Auto-updating information must have the same options or an option to control the frequency (this is an overlap with 2.2.1)

#### Except

- Any of the above types of content are used but don't start automatically
- Moving, blinking or scrolling content of five seconds or less
- The information is not parallel with content but on a standalone webpage – for example a full screen advert displayed to all users before a website is loaded
- An animation that runs as a website or content on a website (for example, a video or file) is loaded, if not indicating the progress would make a user think the webpage had frozen

#### Tips

“Blinking” is not the same as “flashing” from 2.3.1 and 2.3.2. “Blinking” content can cause a distraction but “flashing” content can cause a seizure – as a rough guide “blinking” means less than three times a second and “flashing” means more than three times a second

#### See Also

2.2.1 – Timing Adjustable  
2.3.1 – Three Flashes or Below  
2.3.2 – Three Flashes

***Implementing 2.2.2 is a Level A standard***



### 2.2.3 – No Timing

When a time limit is in place, some disabled users are at a disadvantage. Time limits can be limited to a small number of exceptions.

#### What this means

Timing is never an essential element to the content – so ticket sales websites where you have two minutes to confirm (see 2.2.1) are out

#### Except

- Real time events like auctions
- Live video streams

#### See Also

2.2.1 – Timing Adjustable  
2.2.5 – Re-authenticating

***Implementing 2.2.3 is a Level AAA standard***

## 2.2.4 – Interruptions

Some disabled users have difficulty with focus and attention; interrupting their experience may stop them understanding content. You can avoid this by eliminating interruptions.

### What this means

- Don't use an automatic redirect or refresh function based on a time delay (for example, if a webpage has moved, do not redirect users to the new page after a certain amount of time (use a 301 redirect in this situation))
- Provide an option for turning off all but essential interruptions

### Except

Warnings about a user's health, safety or the security of their data or property as they are considered essential (I'm not sure how a website would know if your house was about to explode but let's not ban it from trying to tell you!).

### HTML

The ideal way to redirect a user from one webpage to another is to do it without them noticing. One of the simplest ways to do this is to edit a website's htaccess file which sits on the root directory (not all servers will allow this file to be edited so check with your hosting provider)

Implement a 301 redirect for individual webpages in .htaccess:

```
➤ redirect 301 /olddirectoy/oldpage.htm http://  
www.newsite.com/newpage.htm htaccess:
```

Implement a 301 redirect for an entire website in .htaccess:

```
➤ Options +FollowSymLinks  
RewriteEngine on  
rewritecond %{http_host} ^oldsite.com [nc]  
rewriterule ^(.*)$ newsite.com/$1 [r=301,nc]
```

Create htaccess text without learning technical jargon at the [htaccess file generator](#)

## Issues

There is an overlap with 2.2.1, which allows for some kind of warning to interrupt a user to tell them that a time limit is approaching. It isn't clear whether this kind of warning would pass 2.2.4. I'm not given to quoting the WCAG2, but in this case it seems necessary. Interruptions are therefore allowed if they are:

"civil emergency alert messages or any other messages that warn of danger to health, safety, or property, including data loss, loss of connection, etcetera."<sup>2</sup>

In fairness, the pop-up option is seen as a less useful solution than others for 2.2.1, but why suggest it without clarifying its impact on 2.2.4? Perhaps a time out is considered a "loss of connection"? Your best bet is to avoid using pop-ups to warn about time limits if you really want AAA, but if your users tell you they find them helpful then give them what they want.

## See Also

2.2.1 – Timing Adjustable

***Implementing 2.2.4 is a Level AAA standard***

---

2. <http://www.w3.org/TR/UNDERSTANDING-WCAG20/time-limits-postponed.html>  
(accessed 19/12/2011)

## 2.2.5 – Re-authenticating

For some websites it is essential to require users to re-authenticate for certain functions (for example, login expiring after a certain time on an e-commerce website). While these websites use this function for a user's security (see 2.2.1), you can retain accessibility by saving all information entered by the user.

### What this means

- A user can continue exactly as before, after re-authenticating, by saving their data (for example, their shopping basket contents, input into forms or accessibility options)
- Surveys and questionnaires can be saved and completed at a later date

### Tips

Consider whether your use of a time limit is justified under 2.2.1 and 2.2.3

### See Also

2.2.1 – Timing Adjustable  
2.2.3 – No Timing

***Implementing 2.2.5 is a Level AAA standard***

## Guideline 2.3 – Seizures

Some users can be affected by repetitive flashing, especially sufferers of epilepsy. You should minimise occurrences of these.

### 2.3.1 – Three Flashes or Below

To make websites as safe as possible for all users, you must limit the amount of flashing media.

#### What this means

Do not add anything to your website that flashes more than 3 times a second

#### Except

There is a ridiculously complicated exception based on the user's visual field which you don't need to worry about if you follow this guidance

#### Tips

Remember, flashing is different to blinking (2.2.2)

#### Issues

Here is that exception, use it if you can:

“the combined area of flashes occurring concurrently occupies no more than a total of .006 steradians within any 10 degree visual field on the screen (25% of any 10 degree visual field on the screen) at typical viewing distance”<sup>3</sup>

See what I mean?

#### See Also

2.2.2 – Pause, Stop, Hide  
2.3.2 – Three Flashes

***Implementing 2.3.1 is a Level A standard***

---

3. <http://www.w3.org/TR/UNDERSTANDING-WCAG20/seizure-does-not-violate.html> at general flash and red flash thresholds (accessed 19/12/2011)

## 2.3.2 – Three Flashes

To make websites as safe as possible for all users, you can minimise the risk of causing seizures by avoiding flashing content.

### What this means

Do not add any content to your website that flashes more than 3 times a second

### Tips

- All flashing content is annoying and distracting
- Remember, flashing is different to blinking (see 2.2.2)

### Issues

Having a guideline that is designed to reduce the amount of seizures caused by websites seems like it should be more important than Level AAA, which few people aim for. There's no reason why this can't replace 2.3.1 and exist at Level A. One of the many problems with the current three-tier structure is that if people know they can't attain AAA, they won't even look through the guidelines to see where they can improve accessibility.

### See Also

2.2.2 – Pause, Stop, Hide  
2.3.1 – Three Flashes or Below

***Implementing 2.3.2 is a Level AAA standard***

## Guideline 2.4 – Navigable

Some disabled users may need extra help in moving around a website, navigating to certain sections, following links and may get “lost” if there is no assistance.

### 2.4.1 – Bypass Blocks

Websites typically have a block of content at the top of a webpage that contains a header and the main navigation. Users with a screen reader will have to listen to these features being read aloud for every webpage they visit on your website. You must provide a way for users to skip this content.

#### What this means

Add a Skip to Content link as the very first bit of content on a webpage – above the header in the top left is ideal

#### Tips

HTML5 allows navigation menus to be wrapped in the `<nav>` element. Screen readers will eventually recognise this and make Skip to Content links redundant

#### HTML

The Skip to Content link must point to a bookmark next to the main content:

```
➤ <a href="#maincontent">Skip to Content</a>
```

Add a bookmark at the start of your main content:

```
➤ <a id="maincontent"></a>
```

The `<nav>` element wrapped around a navigation menu:

```
➤ <nav>
  <ul>
    <li><a href="/bass">Bass</a></li>
    <li><a href="/exotic-fish">Exotic fish</a></li>
  </ul>
</nav>
```

## See Also

3.2.3 – Consistent Navigation

***Implementing 2.4.1 is a Level A standard***

## 2.4.2 – Page Titled

Users must understand where they are on a website and what a webpage is about without having to read through the entire content. Page titles are a simple way to achieve this.

### What this means

Make sure every webpage on your website has a unique and descriptive page title

### Tips

Use these page titles in your sitemap (2.4.5 and 2.4.8) to make it more accessible

### HTML

Page titles are controlled by the title element in the head section of a HTML document:

```
➤ <head>
  <title> Welcome to BassFishing </title>
</head>
```

### See Also

2.4.5 – Multiple Ways  
2.4.8 – Location

***Implementing 2.4.2 is a  
Level A standard***



### 2.4.3 – Focus Order

This overlaps with 1.3.2 on sequence of content and 2.1.1 on keyboard navigation. Users must be able to navigate (by keyboard if they choose) a website's content in a sequential order, so that meaning of information is preserved.

#### What this means

A clean HTML/CSS website will typically be navigable in the sequence intended for regular users – test with a keyboard to check

#### Except

- Complex items like tree diagrams do not have to be exactly programmed – as long as a keyboard user can reach every element
- If your website doesn't need to be navigated sequentially to be understood by regular users (for example, a collage of links or images)

#### See Also

1.3.2 – Meaningful Sequence  
2.1.1 – Keyboard  
2.4.7 – Focus Visible

***Implementing 2.4.3 is a Level A standard***

## 2.4.4 – Link Purpose (In Context)

Links can be problematic for some disabled users if it is not clear where they lead to. Think of all the links you've seen that only use the text "[Click here](#)" – this isn't very useful to a user who has difficulties with focus or attention. Such users might be using an assistive technology that presents a list of links or be reading content heavily magnified. The purpose of a link must be clear from its context.

### What this means

- The purpose (destination or action) of the link must be clear from either the link text **or**
- The purpose of the link must be clear from the surrounding context (for example, the same sentence, paragraph or a cell in a table) **or**
- The purpose of the link must be clear from a title attribute **or**
- If an image is used as a link, the alternative text describes the purpose of the link

### Except

If the purpose of the link is ambiguous to regular users – for example "One of my hobbies is [bass fishing](#)" because the link could be to a photo of a bass, a Wikipedia page or a personal website – no user would know before clicking it

### Tips

- Keep the explanatory context before the link: "Visit my bass fishing [website](#)" is better than "Visit my [website](#) about bass fishing" ("Visit [my bass fishing website](#)" is best, and conforms to 2.4.9)
- Links with the same destination should have the same description, but links should not share a description if they lead to different destinations (3.2.4)
- For icons as links, try to use text with the icons (otherwise add an alternative text description to the icon)
- "[Read more](#)" links can pass this guideline if they follow explanatory text (for example, a few sentences of an article than a link to the full article)
- Inform users when a link will open in a new window

## HTML

A hyperlink with good link text:

```
➤ <a href="http://www.mybassfishingsite.com">
  Visit my bass fishing website</a>
```

Adding alternative text to images when they are used as links:

```
➤ <a href="http://www.BassFishing.com">
  </a>
```

Adding a title element to a link:

```
➤ <a href=http://www.BassMasters.com title="Bass
  fishing information">my hobby</a>
```

Opening a link in a new window (with a title element explaining):

```
➤ <a href=http://www.BassMasters.com
  target="_blank"
➤ title="This link opens in a new window">
  bass fishing</a>
```

## Issues

Context is hard to justify as an excuse when you consider the severity of issues caused by poorly defined links. Many users who browse with screen reader software access a list of links on a webpage and use that list to navigate – links that do not make sense out of context are useless to those users.

The main function of this guideline should be to move web designers away from “Click here” and “Read more” links.

## See Also

1.4.9 – Link Purpose (Link Only)  
3.2.4 – Consistent Identification

***Implementing 2.4.4 is a Level A standard***

## 2.4.5 – Multiple Ways

Disabled users who have difficulty understanding website navigation should be given a variety of ways of finding webpages.

### What this means

- Add an HTML sitemap webpage and link to it after the “Skip to Content” link from 2.4.1 **and**
- Include a search function on every webpage (by adding it to the header) – some disabled users will not understand a sitemap’s layout so a search is more useful to them

### Except

For pages that users only reach after a process (for example, a receipt or confirmation page)

### Tips

- A good HTML sitemap will depend on the structure of your website. Design your page so that it best reflects the structure of your webpages and highlights the most important sections
- A good main menu with descriptive and logical headings is useful
- A related pages section is a good way of helping users navigate around a theme

### See Also

1.4.2 – Page Titled  
2.4.1 – Bypass Blocks  
2.4.8 – Location

***Implementing 2.4.5 is a Level AA standard***

## 2.4.6 – Headings and Labels

Some disabled users benefit from content being broken into chunks with descriptive headings; it helps those with poor memory or concentration as well as visually impaired users who need to skip between headings. You should break up content with headers and you should label all elements on a webpage to make it easy to understand.

### What this means

- Use informative headings and subheadings where appropriate (a change in topic or purpose) to aid navigation through content
- Label all elements (for example, sidebar widgets, forms, search boxes, tables)

### Tips

- A single letter can be enough (for example, in an alphabetical index)
- If you produce similar content, keep headings consistent (for example, a website about films might have “Plot”, “Characters” and “Verdict” on each individual webpage)

### HTML

Header tags go from:

➤ `<h1>The Most Important Header</h1>`

to

➤ `<h6>The Least Important Header</h6>`

### See Also

1.3.1 – Info and Relationships  
1.3.2 – Meaningful Sequence

***Implementing 2.4.6 is a Level AA standard***

## 2.4.7 – Focus Visible

When navigating by keyboard, users should be able to see which element they are focused on.

### What this means

- The keyboard focus indicator must be visible on all elements – this is often a border around the element
- Use CSS to apply either a change in background colour or add a border when focus is received

### Tips

A HTML/CSS website should automatically conform to this guideline – check yours to be sure

### See Also

2.1.1 – Keyboard  
2.4.3 – Focus Order

***Implementing 2.4.7 is a Level AA standard***

## 2.4.8 – Location

As some disabled users may have difficulty understanding the structure and navigation of a website, they can be shown where they are (in terms of the whole website) on every webpage or from an easily-findable index page.

### What this means

- Use a breadcrumbs feature for navigation (something that has links at the top of the page highlighting current position: Home -> Fish -> Bass, for example)
- Add an HTML sitemap (see 2.4.5) webpage and link to it after the Skip to Content link from 2.4.1
- If your website uses a process (for example, shopping for products), show all the steps in the process with breadcrumbs and highlight the current step to help users with cognitive disability

### Tips

- Use full page titles for breadcrumbs when they are 1-3 words long (otherwise they get too long to be displayed)
- Abbreviate longer titles to make them easier to read (for example, “A Guide To Exotic Fish” could just as well be “Exotic Fish”)
- If a page has a ridiculous number of breadcrumbs, perhaps your website is not very well organised
- Use these page titles in your sitemap, organised under headings where useful, to make it more accessible

### HTML

Add breadcrumb navigation to pages with > separators:

```
➤ <a href="home.html">Home</a> >
  <a href="fish.
html">Fish</a> > <a href="bass.html">
Bass</a>
```

### See Also

2.4.1 – Bypass Blocks  
2.4.5 – Multiple Ways

***Implementing 2.4.8 is a  
Level AAA standard***

## 2.4.9 – Link Purpose (Link Only)

You can ensure that link purpose is identifiable from only the text of the link.

### What this means

- The purpose (destination or action) of the link must be clear from the link text alone (for example, “Visit [my bass fishing website](#)”)
- If an image is used as a link, the alternative description describes the purpose of the link

### Except

If the purpose of the link is ambiguous to regular users – for example “One of my hobbies is [bass fishing](#)” because the link could be to a photo of a bass, a Wikipedia article or a personal website – no user would know before clicking

### Tips

- Links with the same destination should have the same description, but links should not share a description if they lead to different destinations (3.2.4)
- For icons as links, try to use text with icons (otherwise add an alternative description to the icon)
- “[Read more](#)” or “[Click here](#)” links will not pass this guideline
- Inform users when a link opens in a new window
- The title attribute (which was an exception in 2.4.4) is not “sufficient” to pass this guideline, only “advisory” (which means you can use it if you want but you can’t rely on it to pass the guideline)



## HTML

A hyperlink with good link text:

```
➤ <a href="http://www.mybassfishingsite.com">Visit my  
bass fishing website</a>
```

Adding a title element to a link:

```
➤ <a title="Link to my bass fishing website" href=  
"http://www.mybassfishingsite.com">bass fishing</a>
```

Adding alternative text to images when they are used as links:

```
➤ <a href="http://www.BassFishing.com"></a>
```

Images where the text is also present can have null alt attributes:

```
➤ <a href="http://www.BassFishing.com">My bass fishing website</a>
```

## See Also

1.1.1 – Text Alternative  
2.4.4 – Link Purpose (In Context)  
3.2.4 – Consistent Navigation

***Implementing 2.4.9 is a Level AAA standard***

## 2.4.10 – Section Headings

Adding section headings to all content will help disabled users who have difficulty focusing or remembering where they are on a webpage as well as visually impaired users who may navigate by skipping between headings. You can help these users by ensuring that all content on your website is broken up by informative headings.

### What this means

Every new and unrelated thought or topic has a heading (for example, website about films might have “Plot”, “Characters” and “Verdict” on each individual webpage)

### Except

A webpage that is a single block of content about one thought or topic

### Tips

- A section is defined by the W3C nerds as “a self-contained portion of written content that deals with one or more related topics or thoughts”, but can be more than one paragraph.<sup>4</sup>
- Beware of making your content harder to read for regular users. Although a well broken up film review might look good, it’s not hard to see how a blog that runs on connected, free-associated thoughts could be made worse by too many headings

### HTML

Header tags go from:

► `<h1>The Most Important Header</h1>`

to

► `<h6> The Least Important Header</h6>`

---

4. <http://www.w3.org/TR/WCAG20/#sectiondef> (accessed 19/12/11)

5. <http://www.w3.org/TR/UNDERSTANDING-WCAG20/navigation-mechanisms-headings.html> at Intent of this Success Criterion (accessed 19/12/11)

## Issues

This is a difficult guideline to understand; 1.3.1 and 2.4.6 already say that websites should be broken up with headings that describe a change in topic or purpose.

This is another case of the W3C adding something at Level AAA (like 2.1.3) where some websites simply cannot conform – they give an example of a website that reproduces a pre-existing document that does not have section headings.<sup>5</sup>

If we take that example given in the explanation to the guideline by the W3C we can see the problems. A website that is otherwise perfectly accessible would fail if it reproduced a letter from Abraham Lincoln to Queen Victoria that did not use headings. Presumably, disabled users are aware that neither the President nor the Queen would have been writing for them, but might be interested in an AAA rated website about history.

Similarly, publishing a modern letter would fail even if you could edit the content – letters don't have headings but often contain distinct topics. It would seem common sense that the original format of a document should be made an exception to this guideline. Guidelines 1.4.5 and 1.4.9 (restriction on using images of text) allow for an exception when presenting a text that must be an image (for example, on a page about fonts that aren't available as text). In those guidelines, the purpose of the webpage is given weight and balanced against absolute accessibility. Why can't that be achieved here?

Providing an accessible alternative (for example, a transcript that inserts headings) may go some way to helping but will inevitably alter the meaning of the original document which was intended by its authors to be read in a certain format.

## See Also

- 1.3.1 – Info and Relationships
- 1.3.2 – Meaningful Sequence
- 1.4.5 – Images of Text
- 1.4.9 – Images of Text (No Exception)
- 2.1.3 – Keyboard (No Exception)
- 2.4.6 – Headings and Labels

***Implementing 2.4.10 is a Level AAA standard***

# 3

---

## Principle Three – Understandable

**S**ome disabled users have difficulty with things that regular users take for granted, however many of the techniques that ensure websites are accessible are also helpful to the overall user experience. These are things like identifying the language of a page, explaining unusual or technical words, making website functions predictable and providing assistance when errors occur.

### **Guideline 3.1 – Readable**

The most basic expectation of any website is that it should be readable. However, there are various degrees of “readability” defined by the W3C, a few of which place huge burdens on website owners.

### 3.1.1 – Language of Page

A website must have a language assigned, so that assistive technology can be as accurate as possible, down to the pronunciation given to words (see: Tomato v Tomato).

#### What this means

Your website has a language meta tag

#### Tips

For a website in English, the “en” tag is usually enough – use “en-us” or “en-gb” if you really want to be geographically specific, but remember that an American user might not understand the “en-gb” pronunciation of a word

#### HTML

Inside the <html> tag, declare your website’s language:

```
➤ <html lang="en">
  </html>
  ...
```

Check [w3schools’ list of language codes](#)

**Implementing 3.1.1 is a  
Level A standard**

### 3.1.2 – Language of Parts

Some websites present content in languages other than the main website language; you should mark each change accordingly so that users with assistive technology can understand the full text.

#### What this means

- Your website has a language meta tag (see 3.1.1)
- Add a language attribute to content that is not in the main website language

#### Except

If you are using a single word that has become part of the main language (for example, “rendezvous” is used in English but is of French origin)

#### Tips

If you have an alternative language version or translation of some content, link to it with the name of the language in that language (for example, “Francais”, “Deutsch”) and add a language tag to the link

#### HTML

In the body of your content, wrap the language attribute around the text:

Quoting another language:

```
➤ <q lang="fr"> un pêcheur tenant une basse </q>
```

A paragraph in another language:

```
➤ <p lang="fr"> un pêcheur tenant une basse </p>
```

Linking to an alternate language version:

```
➤ <a href="http://www.BassFishing.com/fr" hreflang="fr">Francais</a>
```

Check [w3schools' list of language codes](#)

#### See Also

3.1.1 – Language of Whole

**Implementing 3.1.2 is a Level AA standard**

### 3.1.3 – Unusual Words

Some disabled users may have difficulty understanding unusual uses of words (for example, figurative language), words used in very specific ways, idioms and jargon. You can implement a technique to help explain these uses.

#### What this means

- Explain the use of the word inline (for example, “I like bass. A bass is a fish.” or “I like bass (a type of fish).” with definition HTML markup **or**
- Link the word to a definition on a webpage/glossary **or**
- Link the word to a definition at the end of the webpage
- If the use always means the same thing, you only have to define the first occurrence but if the use changes (“The common European freshwater perch” v “A voice, instrument, or sound of the lowest range”) you must define the word on every occurrence. I hope you don’t lure bass with bass

#### Tips

- It’s always helpful to define technical terms and jargon that any user might not understand depending on their familiarity with the subject
- The most accessible way of creating content is to avoid jargon and idioms altogether. This not only helps disabled users but also users who might be novices in your industry or users from a country where the meaning of the words are different

#### HTML

Explain the word inline with definition markup:

```
➤ <dfn>A bass</dfn> is a fish
```

Link to a glossary entry bookmark on another webpage:

```
➤ <a href="/glossary.html#bass">bass</a>
```

Link to a definition bookmark at the end of a webpage:

```
➤ <a href="#bass">bass</a>
```

Add a bookmark to a webpage:

```
➤ <a id="bass">Bass</a>
```

## Issues

The main problem with defining unusual words is finding agreement on what is unusual. Quite generally, using definitions for phrases that some disabled users might not understand (“It was raining cats and bass”) or technical words makes a lot of sense.

As a rule, if in doubt, spell it out. If you have to think whether not to define a word or phrase, chances are you should.

That said, what a ridiculous burden to put on website owners. Going through existing content, agreeing what needs defining, marking it up, making links, creating a glossary all seems too onerous to get people onside with accessibility. Yes, it’s at the highest level but it’s really just an example of why AAA fails because it is too hard to justify spending resources on. In terms of accessibility, a disabled user may actually have their experience degraded by constant annotation (either by `<dfn>` or links) – this is especially crucial for users with a cognitive disability who need help with attention or focus.

A better solution would be to work with browsers and assistive technology to allow them to supply users with an option to define words they are unfamiliar with. I’m confident that most of the markup used for this (and 3.1.4 and 3.1.6 for that matter) will not be used by disabled users, who will each have knowledge of different words and may struggle with words regular users would not imagine difficult. The browser solution would allow a user to define any word they choose, rather than just the selection of words that regular users thought they might need help with. It also cuts out a hugely time consuming and fairly fruitless exercise.

## See Also

3.1.4 – Abbreviations  
3.1.6 – Pronunciation

***Implementing 3.1.3 is a Level AAA standard***



### 3.1.4 – Abbreviations

Abbreviations may not be known to some disabled users and can be explained.

#### What this means

- Expand the abbreviation inline: “Federal Bureau of Investigation (FBI)” **or**
- Link to an expansion on a definition webpage **or**
- Link to an expansion at the end of the webpage **or**
- Use the abbreviation HTML tag to expand any abbreviation (Dr), acronym (NATO) or initialism (FBI)
- If the use always means the same thing, you only have to define the first occurrence but if the use changes (Doctor v Drive) you must define the word on every occurrence. I hope you’re not a Dr working on Mulholland Dr

#### Except

If the abbreviation is part of the language (for example, laser or CD)

#### HTML

Expand the abbreviation inline with markup:

➤ `<abbr title=“Bass Fishing Society”>BFS</abbr>`

Link to a glossary entry on another webpage:

➤ `<a href=“glossary.html#bfs”>BFS</a>`

Link to a definition at the end of a webpage:

➤ `<a href="#bfs">BFS</a>`

Add a bookmark to a webpage:

➤ `<a id="bfs">BFS</a>`

### Tips

- Avoid using abbreviations wherever possible (for example, instead of “the FBI” every time you could say “Federal Bureau of Investigation” once then refer to them as “the Bureau”
- The tidiest solution is the `<abbr>` option, which creates a hidden expansion that appears on mouse hover and can be understood by screen readers.

### Issues

Abbreviations can be annoying and frustrating to users, and the solutions to explaining them are generally quite complex and not necessarily helpful. The idea that providing a definition on first occurrence is useful is a fallacy. Cognitively disabled users may be just as confused by later occurrences as they cannot remember the explanation – except now there is no definition help. Similarly, not everyone will notice or be able to read definitions and if they can it will interrupt their reading flow (another big no-no for cognitive disability).

Absolute best practice is to avoid using abbreviations.

### See Also

3.1.3 – Unusual Words  
3.1.6 – Pronunciation

***Implementing 3.1.4 is a Level AAA standard***

### 3.1.5 – Reading Level

Some disabled users can only understand content up to a certain reading age; the W3C standard is someone who has had 7-9 years of school. You can aim at this level as you produce content.

#### What this means

- Write content that a person with no more than 9 years of schooling can understand **or**
- Supplement complex content with summaries, diagrams or an audio version **or**
- Provide a version of the content written to the lower age, linked from the regular content

#### Except

Names of things (for example, people, types of fish, films, companies)

#### Tips

- Short sentences are easiest to understand
- Microsoft Word can check for readability against the Flesch-Kinkaid scale, aim for a score of 60+
- Don't rely on automated tests. Think about structure, tense and voice

#### Issues

Though the W3C don't say it, this is another guideline that some websites simply cannot comply with. Their advice to supplement content will only work for some websites, and some content creators.

Most websites rely on unique content, often written by authors that users enjoy reading. To meet this guideline, there is a risk that author individuality might be lost. However from experience, writing pages that meet the minimum 60 on the Flesch scale is not too onerous. Some more complex words have to be simplified and sentences broken up more than you might naturally do. Mostly this is just good practice for writing copy in the first place. Nevertheless, there will be some writers whose content cannot be shaped to a conforming level.

The W3C's other suggestion, providing an alternative version of the same webpage but written in simple language, is at best a waste of time to these types of authors and at worst a complete misunderstanding of authorship.

This guideline reminds me of the issues I have with 2.1.3, which penalises perfectly valid content that disabled users might enjoy. Why wouldn't a disabled user visit a website because they like the way the content is written? Some cognitively disabled users have a much more advanced understanding of complex technical terms and language than regular users. It's like forcing novelists to provide a simple version of their work – if the user (of whatever ability) enjoys it, they will read it. If the content is too complicated then the user will not read it – the only problem with this is when the content is on a public service or product website, where all users have a right to be included.

So, government websites and businesses can learn from 3.1.5. They have a duty to make their products and services as accessible as possible. For businesses this means holding back on sales rhetoric and providing clear and useful information. Now there's something we can all get behind.

For the first time in the WCAG 2.0, I can see a rational distinction between Level AA and Level AAA. Could AA be appropriate for all websites and AAA for websites whose main purpose is to provide information to users about services or products for sale? That way websites whose main purpose is authored content (for example, reviews or news stories) could gain a top level but government and sales websites would be asked to do more. I expand on this thought more in an essay at the end of the guide called Error! Reference source not found.

### ***Implementing 3.1.5 is a Level AAA standard***

### 3.1.6 – Pronunciation

Hard to pronounce words, or words that are spelled the same but pronounced differently (for example, “bass” v “bass”) depending on their use, may not be known to some disabled users. You can explain them.

#### What this means

- Avoid using such words wherever possible **or**
- Provide the pronunciation of hard to pronounce **or** ambiguous words immediately after the word **or**
- Link hard to pronounce or ambiguous words to a definition with pronunciation on a glossary webpage **or**
- Link hard to pronounce or ambiguous words to a definition with pronunciation at the end of the webpage

#### Except

- When the correct form is not clear from context
- Words (for example, Latin) that are just as likely to be unknown to regular users

#### HTML

Link to a glossary entry on another webpage:

```
➤ <a href="glossary.html#bass1">bass</a>
```

Link to a definition at the end of a webpage:

```
➤ <a href="#bass1">bass</a>
```

Add a bookmark to a webpage:

```
➤ <a id="bass1">Bass (bæs)</a>
```

#### See Also

3.1.3 – Unusual Words  
3.1.4 – Abbreviations

***Implementing 3.1.6 is a Level AAA standard***

## Guideline 3.2 – Predictable

As a rule, most websites should be made with predictability in mind. Changes in layout can confuse any user, as will change in function between webpages (for example, on some pages you must click a link to follow it but on others just hovering over the link has the same outcome).

### 3.2.1 – On Focus

When an element on a webpage receives focus (for example, when navigated to with a keyboard) it must not automatically change context as this could disorientate some disabled users.

#### What this means

- No button or link opens on focus **and**
- No forms submit without positive user selection **and**
- No automatic pop-ups **and**
- Focus does not 'jump' to another element automatically **and**
- No other action that substantially alters the webpage occurs on focus alone

#### Tips

- Good HTML/CSS will not do any of these things but check your website with a keyboard
- A dynamic menu, that expands to provide more options on focus, will be fine as it is not a change in context

**Implementing 3.2.1 is a Level A standard**

### 3.2.2 – On Input

Changes of context when entering data into a form or selecting settings can be confusing for users with attention or cognitive difficulties. They must not happen automatically unless the user has been advised before using the element.

#### What this means

- Forms do not auto-submit when all fields are filled **and**
- Focus does not automatically change when a field has been filled **and**
- Using a control does not automatically perform the action

#### Except

If the user is advised of any automatic actions before beginning the form

#### Tips

- Avoid any problems by using “Submit” buttons on forms and controls
- Where this is impractical, make the explanatory text precise (for example, a good accessibility control might say “Select a size to change the text size on the page.” followed by a few options which, when clicked, automatically change the text size)

#### See Also

3.2.5 – Change on Request

***Implementing 3.2.2 is a Level A standard***

### 3.2.3 – Consistent Navigation

As we've already seen in 2.4.5 and 2.4.8 (among others), some disabled users have difficulty navigating websites and are more likely to get “lost” than regular users. In addition, users with impaired sight may navigate using spatial sense. This should be minimised by maintaining a consistent navigation.

#### What this means

- Navigation menus are in the same location **and**
- Navigation menus present their elements in the same order **and**
- All other standard elements (search box, Skip to Content link) are in the same location

#### Tips

Using a template for your website will fulfil the majority of this guideline

#### See Also

2.4.1 – Bypass Blocks  
2.4.5 – Multiple Ways  
2.4.8 – Location

***Implementing 3.2.3 is a Level AA standard***



### 3.2.4 – Consistent Identification

Users with screen readers and screen magnifiers often use familiarity as a means of understanding and selecting functions. Elements with the same function should be labelled consistently across a website.

#### What this means

- Any icons used are consistent (for example, print or RSS link) **and**
- Elements with the same function are labelled consistently **or**
- Elements with the same function are named consistently **or**
- Elements with the same function have a consistent text alternative (see 1.1.1)

#### Tips

- Consistent is not identical (for example, an arrow might link to the next page but depending on the page the text alternative would be “Go to page X”)
- An image can have different meanings, so need different text alternatives to pass (for example, a tick can mean “included” or “approved” or “bass found”)

#### HTML

Adding a text alternative to an image:

```
➤ 
```

Adding labels and names to a form:

```
➤ <form>
  <label for="bass">Bass</label>
  <input type="radio" name="type" id="bass" />
  <br />
  <label for="notbass">Not bass</label>
  <input type="radio" name="type" id="notbass" />
</form>
```

#### See Also

- 1.1.1 – Text Alternatives
- 2.4.4 – Link Purpose (In Context)
- 2.4.9 – Link Purpose (Link Only)
- 4.1.2 – Name, Role, Value

#### ***Implementing 3.2.4 is a Level AA standard***

### 3.2.5 – Change on Request

For some disabled users, any automatic change of context can impede accessibility. They can be limited so that they only occur when requested by the user.

The following five features are cumulative, as all must be in place to pass 3.2.5.

#### What this means

- Websites with automatic updates provide a function to turn this off and update only when requested **and**
- Links open in the same window/tab unless it's essential to open a new window/tab **and**
- If a link must open in a new window/tab, the user is told the link opens in a new window/tab (for example, "link (link opens in a new tab)") **and**
- Forms do not auto-submit when all fields are filled **and**
- Webpages that redirect to newer versions use the correct 301 redirect

#### Tips

You might have fulfilled this guideline if you didn't use the exception in 3.2.2 (which this does not allow for) and addressed auto-updating content in 2.2.2

#### HTML

Use the target attribute to open a link in a new window:

```
➤ <a href="http://www.bassfish.com/" target="blank">Bass Fishing Website</a>
```

The ideal way to redirect a user from one webpage to another is to do it without them noticing. One of the simplest ways to do this is to edit a website's htaccess file which sits on the root directory (not all servers will allow this file to be edited so check with your hosting provider)

Implement a 301 redirect for individual webpages in .htaccess:

```
➤ redirect 301 /olddirectoy/oldpage.htm http://www.newsite.com/newpage.htm
```

Implement a 301 redirect for an entire website in .htaccess:

```
➤ Options +FollowSymLinks
RewriteEngine on
rewritecond %{http_host} ^oldsite.com [nc]
rewriterule ^(.*)$ newsite.com/$1 [r=301,nc]
```

## Issues

Like 1.4.8, the scale of this guideline means that it can only exist at the AAA level, however many of its features are vitally important to disabled users (especially those with cognitive disabilities, who are poorly supported by the WCAG2) and could happily be separated and moved down a level or two.

Content that updates automatically was dealt with at the Level A guideline 2.2.2. The only difference seems to be that 3.2.5 wants a button that allows the user to choose when to update the content, and 2.2.2 asks for a pause function and frequency control. I can't help but see that as backwards logic; adding a simple "refresh" or "update" button is both simpler to code and simpler to use than a frequency control. There's no reason why the lower guideline can't have pause, stop and update as its requirements and remove this from AAA altogether.

Having links open in the same window is just good web design. There's no magic needed here, new windows (or more likely tabs) break the "back" function on a browser (something that cognitively disabled users rely on more than most to navigate). No reason why this (with the exception of "essential") can't exist alone at Level A.

I'd bundle this in with the previous point. Should exist at Level A, because the "back" button is important to all users and website owners should leave the option of whether to leave their website up to users.

This can easily be merged with 3.2.2 at Level A by dropping the exception first time around. Adding warning text clutters up forms and, once again, does not well serve cognitively disabled users who may not be able to remember the warning. A "Submit" button is hardly a burden.

Another guideline I'm not convinced needs to exist at this level. 301 redirects are a must for anyone developing a good website in any event. No user likes to land on a page that says "You are being redirected, please wait or click this link to proceed" so why not stick it in at Level A with an exception of being essential.

## See Also

2.2.2 – Pause, Stop, Hide

3.2.2 – On Input

3.3.5 – Help

***Implementing 3.2.5 is a Level AAA standard***

## Guideline 3.3 – Input Assistance

All users are going to make mistakes when filling out forms on your website, but some disabled users need more help to understand how to avoid and fix these mistakes. You must provide that help.

### 3.3.1 – Error Identification

Errors that are automatically detected (like an invalid postcode) must be identified and described to a user in text

#### What this means

On finding an error, a form should explain in text (as close to the error as possible) what is wrong (for example, a mandatory field not filled in or specific format not used)

#### Tips

- Be as specific as possible with error identification and cover the basics
- Highlight the errors with colours and symbols
- Provide a list of links that the user can follow to jump to the incorrect fields
- Make it easy to resubmit incorrect forms by retaining correct data

#### See Also

1.4.1 – Use of Colour

***Implementing 3.3.1 is a Level A standard***

### 3.3.2 – Labels or Instructions

When content requires user input, the kind of input needed must be described with clear labels and instructions so that disabled users can understand the requirements.

#### What this means

- Label input fields clearly and helpfully **and**
- If a field needs a specific format, use placeholder text in the field showing this format (for example, “Date: 01/01/2000”) **and**
- Mark required fields with more than just colour (see 1.4.1) **and**
- Provide text descriptions for required fields that have not been completed (see 3.3.1)

#### Tips

Keep your labels short and useful – too much description can be a hindrance

#### HTML

Adding labels and names to a form:

```
➤ <form>
  <label for="bass">Bass</label>
  <input type="radio" name="type" id="bass" />
  <br />
  <label for="notbass">Not bass</label>
  <input type="radio" name="type" id="notbass" />
</form>
```

Adding placeholder text to a (search) field:

```
➤ <input type="search" name="user_
  search" placeholder="Search BassMasters.com" />
```

Making a form field required:

```
➤ <input type="text" name="user_name"
  required="required" />
```

#### See Also

1.4.1 – Use of Colour  
3.3.1 – Error Identification

***Implementing 3.3.2 is a Level A standard***

### 3.3.3 – Error Suggestion

Errors that are automatically detected (like an invalid postcode) are identified (see 3.3.1) and if suggestions are known, they should be provided so that disabled users can understand where mistakes have occurred and how to correct them.

#### What this means

- A form identifies an error (see 3.3.1) **and**
- Where the error is missing out a required field, a text suggestion notes this **or**
- If the error is in the format of the input, the suggestion shows the correct format (for example, “The date must be in the form DD/MM/YYYY”) **or**
- If the error is because the input needed to be from a limited list of values, provide these values and explain

#### Except

- If making a suggestion would break security **or**
- Making a suggestion goes against the purpose of the field

#### Tips

- Be as specific as possible with error identification and cover the basics
- Highlight the errors with colours and symbols too
- Provide a list of links that the user can follow to jump to the incorrect fields
- Make it easy to resubmit incorrect forms by retaining correct data

#### See Also

3.3.1 – Error Identification

**Implementing 3.3.3 is a Level AA standard**

### 3.3.4 – Error Prevention (Legal, Financial Data)

Some disabled users are more likely to make mistakes than regular users. Commitments to legal or financial agreements, test responses and changes to user controlled data should have some form of confirmation.

#### What this means

- Legal commitments, financial transactions, test responses and changes to user controlled data are reversible **or**
- Legal commitments, financial transactions, test responses and changes to user controlled data are checked for input errors and the user is given a chance to correct any mistakes **or**
- A confirmation webpage is added to such actions that summarises the input and the outcome, with an option to correct or discontinue. This webpage has an input field (for example, a “confirm” button or checkbox) that acts as confirmation of the submission

#### Tips

- Legal commitments: “transactions where the person incurs a legally binding obligation or benefit”<sup>1</sup>
- User controllable data: for example, “Name and address fields for a user’s account”<sup>2</sup>
- This guideline applies to deletion of data as well as submission
- The most useful means of fulfilling this guideline is by providing a confirmation webpage. This gives the user the chance to read a summary of everything they have requested (which may have been split over several pages in the process) and positively confirm their actions

---

1. <http://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-reversible.html#legalcommitmentsdef> at Legal Commitments (accessed 20/12/11)

2. <http://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-reversible.html#user-controllabledef> at User-controllable (accessed 20/12/11)

#### See Also

3.3.6 – Error Prevention (All)

**Implementing 3.3.4 is a Level AA standard**

### 3.3.5 – Help

Some disabled users need more help than others with user input. Additional help can be provided if a submission field's label does not describe all functionality.

#### What this means

- Labels on input buttons describe the nature of the submission **or**
- Add a help with this field link to further information next to input fields which require more detailed explanation than a label provides **and**
- If a field needs a specific format, use a label, adjacent text or placeholder text showing this format (for example, "Date: DD/MM/YYYY")

#### Tips

- This is an "essential" exception against opening a link in a new window (3.2.5)
- An example of when further information would help would be a field to answer the question: "Why are you a good candidate for this job?" The help could offer hints and tips on a good answer

#### HTML

Use the target attribute to open a link in a new window:

```
➤ <a href="http://www.bassfish.com/" target="_blank">Bass Fishing Website</a>
```

Add labels and names to a form:

```
➤ <form>
  <label for="date">Date (DD/MM/YYYY)</label>
  <input type="text" name="date" id="date" />
</form>
```

Add placeholder text to a (search) field:

```
➤ <input type="text" name="date" placeholder="DD/MM/YYYY" />
```

#### See Also

3.2.5 – Change on Request

**Implementing 3.3.5 is a Level AAA standard**



### 3.3.6 – Error Prevention (All)

Some disabled users are more likely to make mistakes than regular users. All submissions of data can have some form of confirmation.

#### What this means

- All submissions of data are reversible **or**
- All submissions of data are checked for input errors and the user is given a chance to correct any mistakes **or**
- A confirmation webpage is added to the process that summarises the input and the outcome, with an option to correct or discontinue. This webpage has an input field (for example, a “confirm” button or checkbox) that acts as confirmation of the submission

#### Tips

- This guideline applies to deletion of data as well as submission so this process will also need one of the three methods
- The most useful means of fulfilling this guideline is by providing a confirmation webpage for all submissions. This gives the user the chance to read a summary of everything they have requested (which may have been split over several pages in the process) and positively confirm their actions.

#### See Also

3.3.4 – Error Prevention  
(Legal, Financial Data)

***Implementing 3.3.6 is a  
Level AAA standard***

# 4

---

## Principle Four – Robust

**“Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies.”**

Disabled users make use of a far wider range of browsers and assistive technologies than regular users. A website must be compatible with as many of these as possible.

## Guideline 4.1 – Compatible

### 4.1.1 – Parsing

All browsers benefit from correct HTML, but assistive technology may encounter issues that regular browsers can ignore. HTML must be checked for errors.

#### What this means

- All HTML elements have complete start (< >) and end (</ >) tags
- All HTML elements are nested correctly
- Any IDs used are unique
- No HTML elements contain duplicate attributes

#### Except

Where HTML allows any of the above features

#### Tips

- Use HTML5 standards
- Use W3C's HTML validator but check your markup by eye for best results

#### HTML

- [W3schools' full list of valid HTML5](#)
- The [W3C's HTML validator](#)

#### Issues

For some reason the W3C restricts valid HTML to mean only that elements are correctly closed and correctly nested, so

➤ `<p></p>`

or

➤ `<p><p></p>`

would both be failures, but completely made-up or outdated HTML would be fine. The W3C had a chance to set a level playing field of valid HTML but ducked the responsibility. Far better would have been to demand that only HTML5 elements would pass for accessibility, or at the least added a Level AA or Level AAA equivalent to that effect.

**Implementing 4.1.1 is a Level A standard**

#### 4.1.2 – Name, Role, Value

This is a rule aimed at people authoring their own script outside of standard HTML markup. Basically, a user interface element (for example, a form element or link) must conform to a set of standards similar to HTML so it can be understood by all browsers and assistive technologies.

##### What this means

- Use HTML specifications for any script you author for your website **and**
- If you use a plugin or other element authored by a third party, make sure it uses valid HTML markup

##### Tips

Names and labels are especially important here; assign them according to HTML specifications so that assistive technologies can understand them

##### HTML

- [W3schools' full list of valid HTML5](#)
- The [W3C's HTML validator](#)

**Implementing 4.1.2 is a Level A standard**



# What to do next

---

**S**o, you made a website that conforms to one (or more) of the W3C's levels, what's next? In truth, the real work is done. The whole point of this guide was to show you how to make an accessible website that doesn't compromise your design or content. That's the most important thing you can do with your website.

The next step is making sure that your website truly does comply, and there are a few things to double check before you make any claims. This is pretty easy, but reading the WCAG2 advice seems to make it look like a more complicated journey than walking to Mount Doom.

Technically, conformance claims are made for individual webpages, but this is really just another example of the W3C being out-of-touch with the way the internet works. To have made all your efforts worthwhile and to make a claim useful to a disabled user, it must cover your entire website. I have adapted the W3C guidance to reflect this below.

## Conformance Checklist

To ensure you are making a valid claim you must:

- Have fulfilled all the guidelines for the level you are claiming (including all of any lower level)
- Make sure that all parts of your website conform (for example, you're not ignoring your footer or a few pages with uncaptioned videos)
- If your website has a process (for example, buying a product), every page in that process conforms to the level you are claiming
- Be sure that you are claiming success based on accessible technologies (all the techniques in this guide are accessible technologies)

## Conformance Claims

What you need to do next is tell people about your work. This is not part of becoming accessible; you do not have to be recognised by the W3C or any other body in order to claim accessibility.

You should send out a press release, write a blog post or tell your customers by whatever means you usually employ that you've made your website A, AA or AAA accessible.

You can also tell the W3C about your claim by yet another drawn out process, including the following:

### The date of the claim

The title, version and URL of the guidelines you have followed (this guide helps you follow: "Web Content Accessibility Guidelines 2.0 at [www.w3.org/TR/2008/REC-WCAG20-20081211](http://www.w3.org/TR/2008/REC-WCAG20-20081211)")

### Conformance Level (A, AA or AAA)

A description of the webpages for which the claim is made (if you are claiming for your entire website as I suggest, your main domain will do)

A list of the web technologies relied on (for example, HTML, CSS, JavaScript, Flash)

If you do make a claim, there have a number of well-recognised [WCAG2 conformance badges](#) you can place on your website.

## Maintaining Standards

Making a great-looking, accessible website is one thing, but keeping it that way is another altogether. If you run a news website or blog, you're going to be adding new content on a regular basis. Even largely static websites will have to be updated from time to time. No-one said taking on accessibility was going to be easy, but only the W3C said it would be this hard.

In truth, it's somewhere in the middle; you're building an extra level of editing into your workflow. A few examples of things you might need to consider:

- Are new videos being captioned, transcribed or audio described?
- Is new text content being edited for HTML markup like abbreviation and definition?
- Are new pages being checked for valid HTML?
- Are new links clear from their text alone or are you adding titles to them?

---

1. <http://www.w3.org/TR/WCAG/#conformance-claims> at Statement of Partial Conformance - Third Party Content (accessed 21/12/11)

Where you run a website with third party content like comments or submitted articles, you can continue to claim conformance as long as you monitor new content and ensure it complies within two working days of it being uploaded to your website.<sup>1</sup>

### Partial Conformance

For websites that use third party content (video, audio, text or whatever), a “statement of partial conformance” can be made. This tells disabled users that the vast majority of a website conforms to a certain level but some specific elements mean that it cannot claim full conformance.

You can put a statement on an accessibility page, alongside any badges, that reads something like:

“This website does not conform, but would conform to the Web Content Accessibility Guidelines 2.0 at Level ..... if..... were removed.”

On one of my websites I sometimes have to use videos of movie trailers that are not captioned or audio described so I fill the blank with:

“all third party video content in film reviews”

You could also do this on a page-by-page basis but, I will repeat, it is my strong belief that accessibility standards must be website-wide to be useful.



# What's wrong with the Web Content Accessibility Guidelines 2.0?

---

**T**hroughout this guide I've highlighted specific areas where the W3C can improve the guidelines; in this final part I'll discuss a few more general problems I have with the WCAG2. Before I begin, let it be remembered that I applaud the efforts of any organisation that cares about accessibility and it is only through continued criticism and dialogue that any improvements can be made.

## Technology

Due to the huge time lag between iterations of the WCAG, the second set of guidelines has been written as “not technology-specific”<sup>2</sup>. In doing so, the W3C have managed to strip out most of the useful techniques they could have promoted and left a document that bares little resemblance to modern web design practices.

Far better would be a set of techniques for each guideline that describe HTML, CSS, Flash, JavaScript, etc. methods of meeting them. This would necessitate a much more regularly updated WCAG, but even an annual update is quite reasonable. We could have WCAG2.1 and so on, without adding excessively to the burden of the W3C. Such a set of rules would be much more useful and recognisable to web designers.

A prime example of the problem is the upcoming launch of HTML5. I say “launch”, but of course most of the techniques are already in use and have been around for years. What I really mean is the upcoming acceptance of HTML5 by the W3C<sup>3</sup>. Many of the guidelines currently recommend techniques that are not supported in pure HTML5, things like the acronym element which has been folded into abbreviation. Even small changes like these confuse web designers who do not recognise suggested techniques. At the very least, an annual edition could tidy up these disused elements.

---

2. Web Content Accessibility Guidelines 2.0, in Abstract on [www.w3.org/TR/WCAG20/](http://www.w3.org/TR/WCAG20/)

3. See <http://www.w3.org/TR/html5/> for an up-to-date status report

## Language and Style

In an incredibly ironic move, the WCAG2 are written in some of the least accessible language I have ever read. It's hard to see how any user, let alone a cognitively disabled user, could understand what they mean from a single reading. They are filled with technical jargon, so much so that every guideline has a few links off to definition pages. In many cases it is impossible to read through an entire guideline without leaving the webpage more than once.

The definitions themselves are hardly clear; in this example links are bolded:

“non-text content any content that is not a sequence of characters that can be **programmatically determined** or where the sequence is not expressing something in **human language**

Note: This includes **ASCII Art** (which is a pattern of characters), emoticons, leetspeak (which uses character substitution), and images representing text”<sup>4</sup>

That's a definition which itself requires three definitions, after which I doubt many readers will fully understand its implications anyway. Interestingly they don't think any further definition that the one in brackets is required for “leetspeak” which is one of the more obscure words in the example.

The definitions are included in the main WCAG2 document, which runs to 72 pages. The two supporting documents Understanding WCAG2<sup>5</sup> and Techniques for WCAG2<sup>6</sup> weigh in at a combined 527 pages. They are incredibly time-consuming to read, not to mention barely understandable and astoundingly boring. What web designers need is a concise document that uses clear language to explain guidelines and techniques. Hopefully I have contributed in some small way to that idea with this guide.

## Cognitive Disability

Much of the criticism of the WCAG2 surrounds the issue of accessibility for users with a cognitive disability. People like Lisa Seaman<sup>7</sup>, Joe Clark<sup>8</sup> and Gian Sampson-Wild<sup>9</sup> have all bemoaned the lack of robust support for these users. All three mention the issue of testability as the main reason why support for cognitively disabled users is not particularly strong in the guidelines. The basic premise being that rules like “have clear and understandable writing” cannot be tested reliably by either machines or humans.

Personally, I agree with the consensus that cognitively disabled users are poorly served by the WCAG2. The W3C's desired reading level isn't really helpful when it comes to accessibility; while some cognitively disabled users need very simple text, others have a

4. From <http://www.w3.org/TR/2008/REC-WCAG20-20081211/#non-text-contentdef>

5. <http://www.w3.org/TR/UNDERSTANDING-WCAG20/>

6. <http://www.w3.org/TR/WCAG-TECHS/>

7. <http://lists.w3.org/Archives/Public/w3c-wai-gl/2006AprJun/0368.html>

8. <http://www.alistapart.com/articles/tohellwithwcag2>

9. <http://www.alistapart.com/articles/testability>

much higher than average reading ability. Much more effective would be to enforce a certain style of presentation in the WCAG2, such as “clear and understandable”, broken into sections with useful headings.

As it stands the W3C wants a mish-mash of headings and explanations. While no-one running a website needs to write clearly, they must ensure that every single “unusual” word is explained with a link (and we’ve seen how effective that technique is for the guidelines themselves) or some HTML markup like definition.

The same goes for words that need a certain pronunciation to be understood, if they can’t be understood in context. In 3.1.6, I used the example of bass (bæs) v bass (bays) but it’s hard to see how that would ever confuse a reader. Indeed, it’s hard to imagine when a word’s pronunciation would not be clear from its context. If all we are worried about is screen readers, then we have to make sure the technology in those technologies is good enough, not add ridiculously to a content writer’s burden by asking them to add HTML markup every time a word has a twin.

### Two Levels not Three

The final issue I have with the WCAG2 is the way that its levels are split. In essence we have accessibility (A and AA) and accessibility-plus (AAA). Accessibility is reachable by all websites, with a usually proportionate difference between A and AA, but accessibility-plus is impossible for some websites by definition.

Take 2.1.3, which disallows websites running games or other programs that work by mouse pointer input to gain AAA. I mentioned in the Issues that this doesn’t make any sense; the WCAG should not act as some kind of creativity police, criticising perfectly relevant technologies. Part of being disabled means that some things in life are not available, this is undoubtedly sad and we have a duty to level the playing field wherever possible. However, we already know that a person with low motor skills will have difficulty playing a game that requires a mouse (most kinds of 3D games), and if we know then that disabled person will certainly know. Find me one such person that has gone to play a mouse-based game and then felt personally marginalised by the creator of the website. Some things cannot be made accessible to everyone and disabled users understand this, the WCAG2 has no business in stopping creators of these things gaining a top level. This kind of guideline does not add to the accessibility of the internet and should be amended to allow specifically-designed content to pass.

A similar criticism can be levelled at 2.4.10, which effectively prevents any website that reproduces historical content from passing AAA. Essentially this means that no website can teach users about historical documents and get a top rating. As with 2.1.3 this principle has no use for accessibility and should be amended to allow reproduction.

Instead of this two-tiered system of accessibility and accessibility-plus, there should be a three-tier tree of:

- accessibility-minus – replacing A with features that every website must have as the most basic forms of accessibility
- accessibility – replacing AA with features that every website should have as a good balance between accessibility, features and efficiency
- accessibility-plus – replacing AAA with features that some websites (government, publicly owned) must have and any other website can choose to implement

The current AAA has some very good guidelines (like 2.3.2) which will not be considered by web designers who are not aiming for that level. The WCAG2 does mention that, when making a compliance claim, people are encouraged to fulfil parts of a level above the level they are claiming, but that is a self-defeating strategy. When the way that levels are claimed is by a series of badges, there is no benefit to a website owner to spend resources on guidelines they can't really claim credit for. Sure they can add a note to an accessibility page, but who reads them? Disabled users will see the badge and assume no more than that level entails.

For this reason it is not only essential to remove guidelines that websites cannot meet by definition and create a new highest level where the most demanding practices are kept for public service websites. With a more regular update schedule, guidelines can be dropped down a level as the technology to implement them (or discover them) improves. Take sign language (1.2.6) as an example; no small website can afford to meet this guideline today but websites like [www.direct.gov.uk](http://www.direct.gov.uk) and [www.hmrc.gov.uk](http://www.hmrc.gov.uk) have a moral duty to meet it. In time, a technology will be developed that translates audio and text into a sign language of the user's choice. The guideline should therefore be kept in accessibility-plus and reviewed annually until it can be more easily met.

Rather than going on and on explaining my system further, I will leave the job to the wolves of the World Wide Web. I remain confident that the internet is the greatest tool for accessibility and inclusion ever invented. I remain sure that institutions like the W3C have users' best interests at heart when they sit down to create guidelines. Most of all, I remain convinced that the true accessible potential of the internet has yet to arrive.



# A

---

## Level A checklist

## Level A checklist

Guideline	Completed?
1.1.1 – Text Alternatives	
1.2.1 – Audio-only and Video-only (Pre-recorded)	
1.2.2 – Captions (Pre-recorded)	
1.2.3 – Audio Description or Media Alternative (Pre-recorded)	
1.3.1 – Info and Relationships	
1.3.2 – Meaningful Sequence	
1.3.3 – Sensory Characteristics	
1.4.1 – Use of Colour	
1.4.2 – Audio Control	
2.1.1 – Keyboard	
2.1.2 – No Keyboard Trap	
2.2.1 – Timing Adjustable	
2.2.2 – Pause, Stop, Hide	
2.3.1 – Three Flashes or Below	
2.4.1 – Bypass Blocks	
2.4.2 – Page Titled	
2.4.3 – Focus Order	
2.4.4 – Link Purpose (In Context)	
3.1.1 – Language of Page	
3.2.1 – On Focus	
3.2.2 – On Input	
3.3.1 – Error Identification	
3.3.2 – Labels or Instructions	
4.1.1 – Parsing	
4.1.2 – Name, Role, Value	

# AA

---

## Level AA checklist



### Level AA checklist

Guideline	Completed?
1.2.4 – Captions (Live)	
1.2.5 – Audio Description (Pre-recorded)	
1.4.3 – Contrast (Minimum)	
1.4.4 – Resize Text	
1.4.5 – Images of Text	
2.4.5 – Multiple Ways	
2.4.6 – Headings and Labels	
2.4.7 – Focus Visible	
3.1.2 – Language of Parts	
3.2.3 – Consistent Navigation	
3.2.4 – Consistent Identification	
3.3.3 – Error Suggestion	
3.3.4 – Error Prevention (Legal, Financial Data)	

# AAA

---

## Level AAA checklist

## Level AAA checklist

Guideline	Completed?
1.2.6 – Sign Language (Pre-recorded)	
1.2.7 – Extended Audio Descriptions (Pre-recorded)	
1.2.8 – Media Alternative (Pre-recorded)	
1.2.9 – Audio-only (Live)	
1.4.6 – Contrast (Enhanced)	
1.4.7 – Low or No Background Audio	
1.4.8 – Visual Presentation	
1.4.9 – Images of Text (No Exception)	
2.1.3 – Keyboard (No Exception)	
2.2.3 – No Timing	
2.2.4 – Interruptions	
2.2.5 – Re-authenticating	
2.3.2 – Three Flashes	
2.4.8 – Location	
2.4.9 – Link Purpose (Link Only)	
2.4.10 – Section headings	
3.1.3 – Unusual Words	
3.1.4 – Abbreviations	
3.1.5 – Reading Level	
3.1.6 – Pronunciation	
3.2.5 – Change on Request	
3.3.5 – Help	
3.3.6 – Error Prevention (All)	

**How to meet the Web Content Accessibility Guidelines 2.0 will teach you how your website can be accessible to everyone on the planet. Using the W3C's guidelines as a template, this book will walk you through exactly how to comply with WCAG 2.0 and Section 508. You'll quickly learn how, when and where each guideline applies to your site and what you need to do about them.**

This book breaks down over 700 pages of dry, technical guidance into 100 pages of succinct advice. Learn the easiest ways to increase accessibility without having to learn to code or a whole new vocabulary.

The internet is waking up to the importance of accessibility. Isn't it time you learnt How to meet the Web Content Accessibility Guidelines 2.0?