

Introduction to Accessibility

Feb. 22, 2008
Dave Katten
Rice Web Services
Web Forum

Outline

- Issues in Accessibility
- Tips for Improving Accessibility
- Resources

What is Accessibility?

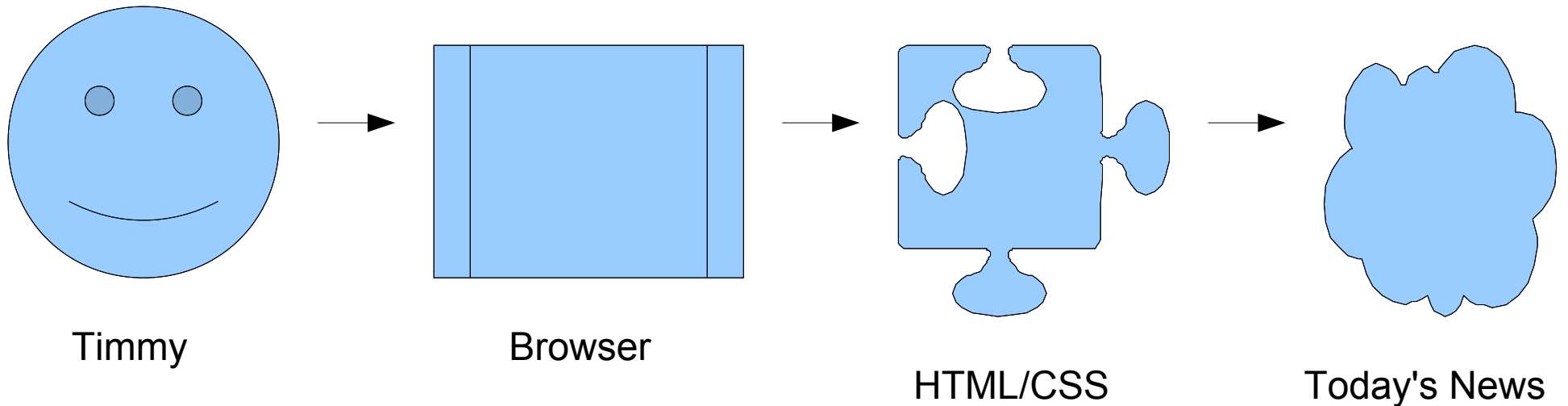
- “Content is accessible when it may be used by someone with a disability.” - W3C Web Content Accessibility Guidelines 1.0
- Make content consumable regardless of disability
 - Sensory (visual, auditory)
 - Cognitive/Neurological (Dyslexia, Epilepsy)
 - Motor
 - Technological (?)

•Standards

- 508 – Minimum standards in the US
- Web Content Accessibility Guidelines (WCAG 1.0)
- WCAG 2.0

•Technology and User Agents

- A *user* employs an *agent* that leverages a *technology* that encompasses content



• **Accessibility Supporting Technologies and Agents**

- Screen readers
- Screen magnifiers
- Captioning formats
- Flash
- PDF

Three Domains of Tips

- Separate Content from Presentation
- Love HTML
- Look for Guidance

Separate Content And Presentation

- Love CSS
- Never Use Tables, Except When You Should
- Image Replacement
- Unobtrusive JavaScript
- Background Check
- Fonts

Love CSS

- Flexible
- Easy to learn
- Difficult to master
- Can be used for page, paragraph, or characters
- Consider a CSS framework

Never Use Tables, Except When You Should

- Tables are for presenting tabular data
 - A page is not tabular data
- `<div>`s and ``s (and other HTML elements) can do anything a table can
- Don't be afraid to use a table when your content is rows and columns
- But seriously, don't use tables
 - But if you do, use `<th>`

Image Replacement

- CSS Technique for decorative text
- “display:none” doesn't cut it
- Often require *some* empty semantics, but worth the cost

IR Example

- `Dave`
- CSS
 - `.ir {position:relative; overflow:hidden;}`
 - `.ir span {position:absolute; top:0; left:0}`
 - `#dave {width: 200px; height: 200px;}`
 - `#dave span {width:200px; height:200px; background: transparent url(path/to/image.jpg) no-repeat;}`

Unobtrusive JavaScript

- Don't hide links in JS
- Your page should have equal functionality with JS off
 - `<noscript>`
- **But** you don't have to avoid JS to have an accessible page
- Using a library can help crystallize the unobtrusive concept
 - JQuery/YUI

ARIA

- Web 2.0, AJAX, RIA
- W3C spec for widgets
- Only supported by Firefox, and not quite a standard
- The future

•**Background Check**

- Ultimately, text and background color is determined at the browser.
- Easy fix: specify both.

•**Fonts**

- Mac vs. PC
- Font size can have impact on design
- Test!

Love HTML

- `<q>` `<abbr>` `<thead>` `<dl>` `<kbd>`
- Semantic Markup

Love HTML, cont.

- Write good alt text
- Write descriptive links
- Use Fieldsets
- Use Labels
- Use Headers
- Use Lists for Navigation

Write Good Alt Text

- Alt text is the alt attribute of the `` tag
- When a screenreader announces an ``, it says "image"
 - Don't repeat!
- Keep it short
- Every `` has alt.
 - Even if it's null

Writing Good Alt Text, cont.

- Good
 - “bridge”
 - “graph: as guns increase, butter decreases”
 - “empire state building”
 - “dave – he gets things done”
 - “”
- Bad
 - “a picture of me and my sister at her wedding”
 - “image”
 - “guns and butter have a complex relationship. Guns are a limited commodity, while butter is seemingly unlimited”

• **Write Descriptive Links**

- "Click here"
- Optimize for scanning
- Title isn't everything

Use Fieldsets

- Group related form elements
- Nest nicely
- `<legend>` give a coherent grouping to form controls

Use Labels

- Give controls a name
- Implicit
 - `<label ... >Label Text<input ... /></label>`
- Explicit
 - `<label for="control">Label Text</label>
<input id="control" />`

Use Headers

- More than just big text!
- Every page should read like an outline
 - Don't use a `<h3>` if you haven't used a `<h1>` and an `<h2>`
- Can be used to separate navigation from main content

Use Lists for Navigation

- Navigation = list of links
- Easily nested
- Easily traversed

•Using “Skip to content”

- “Skip repetitive links”
- Bookmarks
- Mobile users

Look For Guidance

- Choose a Standard
- Tell people how your site works

Choose A Standard

- WCAG 2.0 isn't official, but it's good
 - Targeted for the future
 - Testable
 - Techniques document that documents success and failure strategies
- Validation

Tell People How Your Site Works

- Towards the top of the page, link to a page that describes your accessibility strategy
- Short, simple
- Check out the library's page

Wrapup

- Making an effort is important, but it's easy to hinder instead of help
- Resources are out there
- HTML and CSS are your friend

Questions?

Introduction to Accessibility

Feb. 22, 2008
Dave Katten
Rice Web Services
Web Forum

Outline

- Issues in Accessibility
- Tips for Improving Accessibility
- Resources

What is Accessibility?

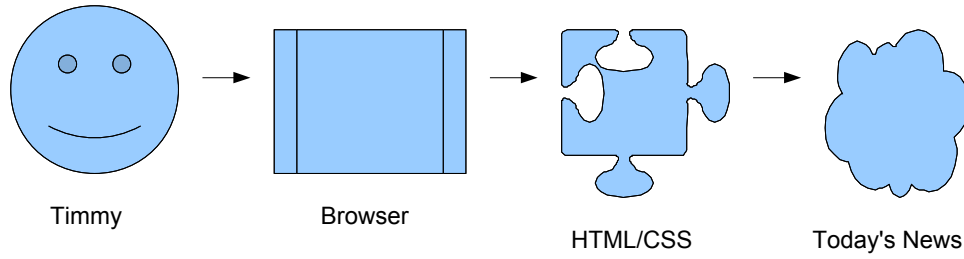
- “Content is accessible when it may be used by someone with a disability.” - W3C Web Content Accessibility Guidelines 1.0
- Make content consumable regardless of disability
 - Sensory (visual, auditory)
 - Cognitive/Neurological (Dyslexia, Epilepsy)
 - Motor
 - Technological (?)

•Standards

- 508 – Minimum standards in the US
- Web Content Accessibility Guidelines (WCAG 1.0)
- WCAG 2.0

•Technology and User Agents

- A user employs an *agent* that leverages a *technology* that encompasses content



•Accessibility Supporting Technologies and Agents

- Screen readers
- Screen magnifiers
- Captioning formats
- Flash
- PDF

Three Domains of Tips

- Separate Content from Presentation
- Love HTML
- Look for Guidance

Separate Content And Presentation

- Love CSS
- Never Use Tables, Except When You Should
- Image Replacement
- Unobtrusive JavaScript
- Background Check
- Fonts

Love CSS

- Flexible
- Easy to learn
- Difficult to master
- Can be used for page, paragraph, or characters
- Consider a CSS framework

Never Use Tables, Except When You Should

- Tables are for presenting tabular data
 - A page is not tabular data
- `<div>`s and ``s (and other HTML elements) can do anything a table can
- Don't be afraid to use a table when your content is rows and columns
- But seriously, don't use tables
 - But if you do, use `<th>`

Image Replacement

- CSS Technique for decorative text
- “display:none” doesn't cut it
- Often require *some* empty semantics, but worth the cost

IR Example

- `Dave`
- CSS
 - `.ir {position:relative; overflow:hidden;}`
 - `.ir span {position:absolute; top:0; left:0}`
 - `#dave {width: 200px; height: 200px;}`
 - `#dave span {width:200px; height:200px; background: transparent url(path/to/image.jpg) no-repeat;}`

Unobtrusive JavaScript

- Don't hide links in JS
- Your page should have equal functionality with JS off
 - <noscript>
- **But** you don't have to avoid JS to have an accessible page
- Using a library can help crystallize the unobtrusive concept
 - JQuery/YUI

ARIA

- Web 2.0, AJAX, RIA
- W3C spec for widgets
- Only supported by Firefox, and not quite a standard
- The future

•**Background Check**

- Ultimately, text and background color is determined at the browser.
- Easy fix: specify both.

•**Fonts**

- Mac vs. PC
- Font size can have impact on design
- Test!

Love HTML

- `<q>` `<abbr>` `<thead>` `<dl>` `<kbd>`
- Semantic Markup

Love HTML, cont.

- Write good alt text
- Write descriptive links
- Use Fieldsets
- Use Labels
- Use Headers
- Use Lists for Navigation

Write Good Alt Text

- Alt text is the alt attribute of the `` tag
- When a screenreader announces an ``, it says "image"
 - Don't repeat!
- Keep it short
- Every `` has alt.
 - Even if it's null

Writing Good Alt Text, cont.

- Good
 - “bridge”
 - “graph: as guns increase, butter decreases”
 - “empire state building”
 - “dave – he gets things done”
 - “”
- Bad
 - “a picture of me and my sister at her wedding”
 - “image”
 - “guns and butter have a complex relationship. Guns are a limited commodity, while butter is seemingly unlimited”

•**Write Descriptive Links**

- "Click here"
- Optimize for scanning
- Title isn't everything

Use Fieldsets

- Group related form elements
- Nest nicely
- `<legend>` give a coherent grouping to form controls

Use Labels

- Give controls a name
- Implicit
 - `<label ... >Label Text<input ... /></label>`
- Explicit
 - `<label for="control">Label Text</label>`
`<input id="control" />`

Use Headers

- More than just big text!
- Every page should read like an outline
 - Don't use a `<h3>` if you haven't used a `<h1>` and an `<h2>`
- Can be used to separate navigation from main content

Use Lists for Navigation

- Navigation = list of links
- Easily nested
- Easily traversed

•Using “Skip to content”

- “Skip repetitive links”
- Bookmarks
- Mobile users

Look For Guidance

- Choose a Standard
- Tell people how your site works

Choose A Standard

- WCAG 2.0 isn't official, but it's good
 - Targeted for the future
 - Testable
 - Techniques document that documents success and failure strategies
- Validation

Tell People How Your Site Works

- Towards the top of the page, link to a page that describes your accessibility strategy
- Short, simple
- Check out the library's page

Wrapup

- Making an effort is important, but it's easy to hinder instead of help
- Resources are out there
- HTML and CSS are your friend

Questions?

- Click to add an outline

Click to add title

- Click to add an outline