Fluid

Designing software
that works – for everyone

U-Camp: Hands-on Accessibility

Colin Clark, Fluid Project Technical Lead, Adaptive Technology Resource Centre, University of Toronto

Topics We'll Cover

Fluid

Designing software
that works – for everyone

- What is accessibility?
- Standards and techniques
- A simple checklist for evaluating accessibility
- DHTML accessibility techniques

USER EXPERIENCE

Fluid

Designing software
that works – for everyone

What is Accessibility?

A New Definition

Fluid
Designing software
that works – for everyone

- Technology provides us with an opportunity to rethink disability and accessibility
- Accessibility can be defined as the *ability of the system to accommodate the needs of the user*
- Inaccessibility is a *mismatch between the user and the interface offered by the system*
- In this model, we all experience disability

USER EXPERIENCE

Fluid's Accessibility Vision

Fluid
Designing software
that works – for everyone

- Embrace diversity:
  - Users all have different needs
  - Under different circumstances
- One size never fits all
- Build systems that can bend and adapt to meet the users' individual needs

USER EXPERIENCE

Models for Web Accessibility

Fluid
Designing software that works – for everyone

1. "Text-only Site" Approach: So 1997!
   - Media-rich site and a separate "accessible" alternative
   - Hard to maintain, falls out of date easily
2. Single Compliant Site Approach:
   - One site accessible site for all
   - Avoids currency gap
   - May not meet all needs
3. Adaptable, Personalizable Approach:
   - Extend the latter approach: build flexibility into system
   - One underlying model, but render personalized views
   - Aligns well with the portal philosophy

USER EXPERIENCE

Motivations for Accessibility

Fluid
Designing software that works – for everyone

- Legislation: you *have to be accessible*
  - Section 508
  - Target Lawsuit
    - Basic question: is a website a “public accommodation?”
- Ethics: it's the right thing to do
  - Technology opens up opportunities
  - Access to social networks, shopping, etc.
- ... but there's more!

USER EXPERIENCE

Accessible Software is Better

Fluid
Designing software
that works – for everyone

- The “curb cut effect:” everyone benefits
- Accessible technology tends to be...
  - More interoperable
  - Easier to re-purpose and reuse
  - More future proof
  - More robust
  - Easier to use on a variety of devices

USER EXPERIENCE

Fluid

Designing software
that works – for everyone

Standards and Techniques

Standards & Guidelines

Fluid
Designing software
that works – for everyone

- W3C Web Content Accessibility Guidelines:
  - WCAG 1.0 vs. 2.0: which one to choose?
  - The tension between specificity and obsolescence
- Section 508
  - Federal agencies must provide equal access
  - Often affects higher education institutions, too

USER EXPERIENCE

Section 508 in a Nutshell

Fluid
Designing software
that works – for everyone

- Text alternatives for graphics & multimedia
- Provide alternatives to color-coding
- Allow pages to work without stylesheets (huh?!?)
- [Obsolete requirements for server-side image maps]
- Label tables sensibly
- Frames suck
- Use text-only pages if you really have to, but keep them up to date
- Make scripts usable with keyboard and assistive technologies
- Avoid evil forms and inaccessible DHTML
- Skip repetitive navigation links
- Give users extra time

USER EXPERIENCE

WCAG 2.0 Concepts

Fluid
Designing software that works – for everyone

- Perceivable
  - Text & multimedia alternatives
  - Design for alternative presentations
  - Layout, colour, and audio flexibility
- Operable
  - Make it work with the keyboard
  - Provide extra time
  - Help the user orient themselves
- Understandable
  - Readable
  - Consistent
  - Help users avoid mistakes

USER EXPERIENCE

Fluid

Designing software
that works – for everyone

Assessing Your Accessibility

Fluid UX Walkthroughs

Fluid
Designing software
that works – for everyone

- A combination of *heuristic evaluation* and *cognitive walkthrough*
  - In translation: a checklist and scenarios for looking at your application's usability and accessibility
- Step into the shoes of your users
- With a bit of help, anyone can do a UX Walkthrough

USER EXPERIENCE

Simple Accessibility Checklist

Fluid
Designing software
that works – for everyone

1. Assess the layout, structure and content of the page
2. Play around with the layout:
    - enlarge the font size
    - change the size of the window
    - adjust your resolution
3. Use the Tab key to navigate through the entire page.
4. Check for alternative text for all images
    - Roll over with Internet Explorer
    - Use Popup Alt Attributes Extension for FireFox

USER EXPERIENCE

Layout and Structure

Fluid
Designing software
that works – for everyone

- Is the page structured into logical sections?
- Are the sections clearly labeled?
- Are there sufficient non-visual cues for site structure?
- Are there sufficient visual cues?
- Is the most important information prominent?
- Is navigation consistent from page to page?

USER EXPERIENCE

Screen Enlargement

Fluid
Designing software
that works – for everyone

- Play around with increasing the font size, changing resolution, and resizing the window
- Is all the text visible? Does it overlap?
- Are headers & labels still correctly associated?
- Do columns shift or realign as expected?

USER EXPERIENCE

Keyboard Navigation

Fluid
Designing software
that works – for everyone

- Conventions:
  - Tab key cycles between widgets
  - Arrow keys navigate within a control
  - Spacebar controls selection
  - Enter actives the control
- Do all links and controls receive focus?
- Can controls be correctly activated?
- Are shortcuts provided to quickly access content?
- Are there any areas where you get stuck or need to use the mouse?

USER EXPERIENCE

Fluid

Designing software
that works – for everyone

Web 2.0 Accessibility

Web 2.0 & Accessibility

Fluid
Designing software
that works – for everyone

- Just when we thought we had Web accessibility in hand...
  - Opaque user interface markup: not enough semantics
  - Non-mouse accessibility
  - Live regions

USER EXPERIENCE

DHTML Accessibility Advice

Fluid

Designing software
that works – for everyone

- Embrace JavaScript
- Use emerging standards: ARIA, tabindex, etc.
- Degrade gracefully in the interim
- Think about the use case for accessibility
- Start with accessibility, don’t add it at the end

USER EXPERIENCE

Assistive Technologies

Fluid
Designing software
that works – for everyone

- Used by people with disabilities to perceive and control the user interface:
- Examples:
  - Screen reader
  - Screen magnifier
  - On-screen keyboard
- Most assistive technologies use built-in operating system APIs for reflecting the user interface

USER EXPERIENCE

Opaque Markup

Fluid
Designing software
that works – for everyone

- Cool new Web 2.0 interfaces push the semantic abilities of DHTML
- Complex UI behaviour is typically attached to generic HTML elements (eg. and)
- Assistive technologies attempt to read the underlying document markup
- Problem: how do assistive technologies represent DHTML interfaces to the user?

USER EXPERIENCE

Opaque Markup: An Example

Designing software that works – for everyone

- A DHTML menu bar without semantics:

```
<div id="myMenuBar">
    <div id="Edit"/>
      <div id="Cut"'/>
        <div id="Paste"/>
</div>
```

USER EXPERIENCE

Opaque Markup: Solution

Fluid
Designing software that works – for everyone

- Provide additional semantics or metadata that describe the role, function, and states of DHTML user interfaces
- How? **ARIA (Accessible Rich Internet Application)**

http://www.w3.org/TR/aria-roadmap/
http://www.w3.org/TR/aria-role/
http://www.w3.org/TR/aria-state/

- Working standard from the W3C, led by Fluid partner Rich Schwerdtfeger

USER EXPERIENCE

ARIA

Fluid
Designing software
that works – for everyone

- Attributes added to your HTML markup that describe the function and states of your UI components
- These map to all your familiar types of UI widgets:
  - Dialog
  - Slider
  - Progress Bar
  - Tab Panel
  - Menu bar

USER EXPERIENCE

Opaque Markup: A Solution

Designing software that works – for everyone

- A DHTML menu bar with ARIA semantics:

```
<div id="myMenuBar" role="wairole:menubar" >
    <div id="Edit" role="wairole:menuitem"
                   haspopup="true" />
        <div id="Cut"' role="wairole:menuitem" />
        <div id="Paste" role="wairole:menuitem" />
</div>
```

USER EXPERIENCE

The Value of ARIA

Fluid
Designing software
that works – for everyone

- DHTML accessibility is a short-term problem
- Long-term, it has the potential to make web accessibility much better
- Assistive technology developers have had a decade to get desktop GUI accessibility right
- By mapping rich-client interfaces with ARIA, web interfaces can leverage this support

USER EXPERIENCE

Non-mouse accessibility

Fluid
Designing software
that works – for everyone

- Most rich Web 2.0-type interactions *require* the mouse
- Standard tabbing strategy in browsers is tedious
- Keyboard bindings will enable almost all of the non-mouse control strategies:
  - On-screen keyboard
  - Single switch
  - Voice control

USER EXPERIENCE

Tabbing and tabindex

Fluid
Designing software
that works – for everyone

- Browsers used to only allow you to use tab to focus form elements and links
- There is an HTML attribute called “tabindex” that allows you to tell the browser how to handle tabbing
- Strategy:
  - allow the user to tab to user interface widgets
  - use the arrow keys allow selection within

USER EXPERIENCE

An Example of Tabbing

Fluid
Designing software
that works – for everyone

- Allow focus to arbitrary DOM elements:

  <div id="myMenuBar" **tabindex="0"**>

- Prevent focus on contained elements:

  <div id="myMenuItem" **tabindex="-1"**>

- Add JavaScript handlers for arrow keys
- Use a toolkit for keyboard events and DOM manipulation, it will make your life much easier!
- This is supported in FireFox 1.5+ and IE 6+

USER EXPERIENCE

Fluid

Designing software
that works – for everyone

Winding Down

USER EXPERIENCE

Accessibility Meta Concepts

Fluid
Designing software
that works – for everyone

1. Label everything
2. It has to scale
3. It has to work with the keyboard

References

Designing software that works – for everyone

- WCAG 2.0:
  - http://www.w3.org/TR/WCAG20/
- WebAIM's Section 508 Checklist:
  - http://www.webaim.org/standards/508/checklist.php
- Accessible Rich Internet Applications (ARIA):
  - http://www.w3.org/TR/aria-roadmap/
- Dojo Toolkit:
  - http://dojotoolkit.org/
- Fluid Project:
  - http://fluidproject.org/

USER EXPERIENCE