Accessibility for Rich Internet Applications:

Techniques & Toolkits

Colin Clark, Fluid Project Technical Lead, Adaptive Technology Resource Centre
Topics We’ll Cover

• A quick introduction
• The Fluid Community
• Challenges of rich internet accessibility
• Solutions and standards
• Dojo
• jQuery
• Fluid Infusion
Introductions
Links

• http://fluidproject.org/
• http://wiki.fluidproject.org
• http://build.fluidproject.org
• http://uidesignpatterns.org/
The Fluid Community

An international open source community of interaction designers, developers, and accessibility experts dedicated to improving the open web.
Challenge for Open Source

UX is a challenge for all open source projects

- How do we address the systemic problems of poor usability and accessibility?
- How do non-technical people get involved in OSS?
- How can we help designers and developers speak the same language?
- How do you do user testing in a distributed environment?
What is Fluid?

An open source project that creates user experience tools and software capable of addressing the needs of diverse users.

- Flexible, reusable user interface components
- Development tools to support inclusive design
- Design Handbook
- Accessible Interaction Designs
- Design Pattern Community
Rich Internet Accessibility
What is Accessibility?
A New Definition

• Accessibility is the ability of the system to accommodate the needs of the user

• Disability is the mismatch between the user and the interface provided

• We all experience disability

• Accessible software = better software
DHTML: A New Can of Worms

- The shift from documents to applications
- Familiar a11y techniques aren’t enough
- Most DHTML is completely inaccessible
- New techniques are still being figured out
Assistive Technologies

• Present and control the user interface in different ways
• Screen readers
• Screen magnifiers
• On-screen keyboards
• Use built-in operating system APIs to understand the user interface
The Problem

• Custom widgets often look, but don’t act, like their counterparts on the desktop

• HTML provides only simple semantics

• Not enough information for ATs

• Dynamic updates require new design strategies to be accessible
The Solution

• Describe user interfaces with ARIA
• Add consistent keyboard controls
• Provide flexible styling and presentation
Supporting Assistive Technology
Opaque Markup

// These are tabs. How would you know?
<ul>
  <li>Cats</li>
  <li>Dogs</li>
  <li>Gators</li>
</ul>
<div>
  <div>Cats meow.</div>
  <div>Dogs bark.</div>
  <div>Gators bite.</div>
</div>
Opaque Markup: Tabs

Dogs *bark.*
ARIA

- Accessible Rich Internet Applications
- W3C specification in the works
- Fills the semantic gaps in HTML
- Roles, states, and properties
- Live regions
Roles

• Describe widgets not present in HTML 4
• slider, menubar, tab, dialog
• Applied using the role attribute
States and Properties

- Added to elements within the DOM
- Properties describe characteristics:
  - draggable, hasPopup, required
- States describe what’s happening:
  - busy, disabled, selected, hidden
- Applied using custom aria- attributes
Using ARIA

// Now *these* are Tabs!
<ul id="animalTabs" role="tablist" tabindex="0">
  <!-- Individual Tabs shouldn’t be focusable -->
  <!-- We’ll focus them with JavaScript instead -->
  <li id="cats" role="tab" tabindex="-1">Cats</li>
  <li id="dogs" role="tab" tabindex="-1">Dogs</li>
  <li id="gators" role="tab" tabindex="-1">Gators</li>
</ul>
<div id="panels">
  <div role="tabpanel" aria-labelledby="cats">Cats meow.</div>
  <div role="tabpanel" aria-labelledby="dogs">Dogs bark.</div>
  <div role="tabpanel" aria-labelledby="gators">Gators bite.</div>
</div>
Keyboard Accessibility
Keyboard Navigation

• Everything that works with the mouse should work with the keyboard

• ... but not always in the same way

• Support familiar conventions
Keyboard Conventions

- **Tab** key focuses the control or widget
- **Arrow keys** select an item
- **Enter** or **Spacebar** activate an item

- Tab is handled by the browser. For the rest, you need to write code.
Tabbing and Tabindex

• Each focusable item can be reached in sequence by pressing the **Tab** key

• **Shift-Tab** moves backwards

• The **tabindex** attribute allows you to customize the tab order

• `tabindex=“-1”` removes element from the tab order: useful for custom handlers
Tabindex examples

<!-- Tab container should be focusable -->
<ul id="animalTabs" tabindex="0">
  <!-- Individual Tabs shouldn’t be focusable -->
  <!-- We’ll focus them with JavaScript instead -->
  <li id="tab1" tabindex="-1">Cats</li>
  <li id="tab2" tabindex="-1">Dogs</li>
  <li id="tab3" tabindex="-1">Alligators</li>
</ul>
Arrow Key Navigation

// Make the tabList focusable with Tab.
var tabList = jQuery("#animalTabs").attr("tabindex", 0);

// Make the tabs selectable with the arrow keys.
var tabs = jQuery("li", tabList);
tabs.selectable(tabList, {
    willSelect: function(aTab) {
        aTab.addClass("highlight");
    }
});
Things to Think About

• What kind of UI are you building?
• Does it resemble something familiar?
• What states or modes does it have?
• Can you reuse an existing widget?
Accessibility Resources

http://codetalks.org

http://wiki.fluidproject.org/display/fluid/DHTML+Developer+Checklist

http://wiki.fluidproject.org/display/fluid/UX+Accessibility+Walkthrough+Protocols

http://developer.mozilla.org/en/docs/Accessible_DHTML

http://developer.mozilla.org/en/docs/Key-navigable_custom_DHTML_widgets

Toolkit Accessibility
Dojo

• The first to tackle DHTML accessibility

• Accessibility features include:
  • Support for Windows High Contrast
  • Comprehensive ARIA library
  • Keyboard navigation

• Most core widgets are covered
Dojo High Contrast

Sample Dialog
Name: 
Location: 
Description: 
OK

Sample Dialog
Name: 
Location: 
Description: 
OK
jQuery

• A truly community-driven effort
• A step-by-step approach
• Features include:
  • Several accessible widgets, more to come
  • Comprehensive ARIA library
  • Keyboard navigation in progress
jQuery

Dialog

Basic dialog

This is the default dialog which is useful for displaying information. The dialog window can be moved, resized and closed with the 'x' icon.

text input

- checkbox
- radio
- select

textarea
Fluid Infusion
Infusion is...

- A collection of rich UI components
- A framework for building your own
- Built on top of jQuery
- Built from the ground up for accessibility
- Personalizable and adaptable
- Ideal for portals, mashups, and CMS
Components

- Encompass familiar Web activities
- Working with files, finding stuff, authoring
- Rich interactions: drag and drop, etc.
- Activities and contexts, not just widgets
- Designed to be extended and adapted
The Reorderer Family

layouts

grids

lists
Uploader

![Uploader Dialog]

- LICENSE.txt: 12.1 KB
- maven.xml: 1.8 KB
- pom.xml: 0.9 KB
- project.properties: 0.1 KB
- project.xml: 1.2 KB
- README.txt: 5.4 KB

Uploading: 5 of 6 files (15.9 KB of 21.3 KB)
## Site Settings

### Members

<table>
<thead>
<tr>
<th>Members</th>
<th>Email</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe I. Instructor</td>
<td><a href="mailto:joe@sakai.university.edu">joe@sakai.university.edu</a></td>
<td>Instructor</td>
</tr>
<tr>
<td>Mike A. Smith</td>
<td><a href="mailto:msmith@yahoo.com">msmith@yahoo.com</a></td>
<td>TA</td>
</tr>
<tr>
<td>Jane Doe</td>
<td><a href="mailto:j.doe@sakai.university.edu">j.doe@sakai.university.edu</a></td>
<td>Student</td>
</tr>
<tr>
<td>Laura S. Roberts</td>
<td><a href="mailto:roberts_laura@gmail.com">roberts_laura@gmail.com</a></td>
<td>Guest</td>
</tr>
<tr>
<td>Erin A. Wilson</td>
<td><a href="mailto:thwilsones@hotmail.com">thwilsones@hotmail.com</a></td>
<td>Student</td>
</tr>
<tr>
<td>Ben Singer</td>
<td><a href="mailto:singer21@hotmail.com">singer21@hotmail.com</a></td>
<td>Student</td>
</tr>
<tr>
<td>Bhaktavatsalam Bhayakridbhyanashanacher</td>
<td><a href="mailto:bb@local.host">bb@local.host</a></td>
<td>Student</td>
</tr>
<tr>
<td>Dept Admin</td>
<td><a href="mailto:do1@local.host">do1@local.host</a></td>
<td>Instructor</td>
</tr>
<tr>
<td>Ángeoio Hasip</td>
<td><a href="mailto:ah@local.host">ah@local.host</a></td>
<td>Student</td>
</tr>
</tbody>
</table>
UI Options & FSS
UI Options: High Contrast Theme

My Dashboard

User Interface Options

- Easier to see
  - Font style: Serif
  - Minimum text size: 12 pt
  - Text Spacing: Regular
  - Line Spacing: None
  - Contrast: High Contrast
  - Background Images: Yes
  - Simplified Layout: Yes

- Easier to find

Preview window (updates automatically)

More Tools
- Home
- Profile
- Membership
- Schedule
- Resources
- Worksite Setup
- Preferences
- Account

Add Tools
- Toggle View

Settings Close

Reset | Save and apply | Cancel

© 2004-2008 The Sakai Foundation
Portions of Sakai are copyrighted by other parties as described in the Acknowledgments screen

Fluid"
Infusion Framework

• Puts you back in charge of your UI
• No black boxes: everything is configurable
• Totally open architecture
• Models, views, but not gluey controllers
• Make your UIs transformable
Getting Involved

- Join our mailing lists
- Share code or designs
- Use Infusion
Links

• http://fluidproject.org/
• http://wiki.fluidproject.org
• http://build.fluidproject.org
• http://uidesignpatterns.org/