The Fluid Project
An Open Source Community for Inclusive Design

Presented by: Mara Hancock
Interim Director, Educational Technology Services, UC Berkeley
Fluid Project board member

Project Funded by Andrew W. Mellon Foundation
Topics We'll Cover

- Fluid Vision
- UX Toolkit: walkthroughs, patterns, and more
- Components
- Project roadmap
- How to get involved
What Is Fluid?

An community source project which creates a user experience, tools, and software capable of addressing the needs of a diverse set of demanding and creative users and projects.

**Core Institutions:** University of Toronto | University of California, Berkeley | York University | University of British Columbia | Cambridge

**Other Participating Institutions:** Michigan State | University of Colorado | University of Michigan | Georgia Tech | UK Open University | Your University Here

**Participating Projects:** Sakai | uPortal | Kuali Student | Moodle | Open Collections | ATutor

**Corporate Partners:** Sun Microsystems | IBM | Mozilla Foundation | Unicon
Who Are We?

- Fluid is intentionally open and distributed
- Fluid is Sakai, Moodle, Kuali, OpenCollection, uPortal, Etc..
- We’re members of many communities
- All solving UX issues together
- Collaborate as an integral part of development teams
- Engage across projects -- what is of cross-cutting importance?
- Consult
Fluid Vision

• Foster user experience design and development within academic community source projects
  …so that they can fulfill their potential as platforms for innovation

• Build a presentation layer that supports:
  – Cross-application sharing of designs
  – The diversity of needs are unique within higher education

• Support, evangelize, and educate about the precarious values of usability, accessibility, quality assurance, and more

• Create a broad and deep community of UX expertise

• Encourage and model investment in these values and expertise
The Fluid Approach to UX in Community Source

- UX is a challenge for all open source projects and all institutions
  - (UX is a challenge for most software projects)
- Cross-project collaboration:
  - Share scarce UX resources across projects
  - Solve common & high priority challenges
  - Recognize recurring user interface idioms and needs
- Breaking down cultural & geographic barriers:
  - Can non-technical people get involved and to gain respect in OSS?
  - Can designers and developers speak a shared language?
  - How do you conduct UX activities in a distributed environment?
- A two-fold approach:
  - Social: build a community around UX
  - Technical: new UI development tools
Encompassing Principle

focus on user goals:

*Tools must be usable and undemanding*

- Users should focus on **teaching, learning, research and administrative tasks**… not on operating the tools
- Institutions should invest in furthering research and learning not in…ballooning support needs
- Institutions should not be concerned with cost of tool rejection and difficult implementations
- Tools should be platforms for innovation
“You say tomato, I say tomato, lets call the whole thing off”

- Academic communities are very diverse
- They differ greatly in their preferences, needs, habits, concepts, comforts, convictions….
  - Institutional preferences and branding
  - Conventions of academic discipline
  - Linguistic differences
  - Perspectives vary related to role (student, staff, faculty)
  - Different teaching approaches
  - Different learning approaches
  - Different research approaches

The academic community fosters and thrives on diversity!
Accessibility & Usability challenge today

- Legal commitment to equal access (Rehab 508, Section 255, ADA, state commitments, institutional policies)
- “Consistency will solve the problem”
- “One size fits all” has significant limitations
  - Constrain user experience for those meant to help
  - Accessibility guidelines seen to constrain creativity
  - “Accessible for everyone, optimal for no one”

We can’t afford to be this limited!
Fluid: "Flexible User Interface"

- Swappable styles
- Modular, reusable UI components
- Web 2.0 -- engage me -- focus
- Either runtime transformation for unique needs of the individual
- Or customization through configuration
Community source UX in higher education has a unique opportunity

- Co-located with our users
  - Our workplace is our laboratory
  - Gather ideas and feedback
  - Observe and involve our users
- Have explicit mission to meet teaching, learning, research, and public service needs (we have permission!)
- Highly motivated staff
State of the Union

Usability/Accessibility in Open Source Software

- Systemic problem of poor and inconsistent user interface
- Poor UX an impediment to adoption & support challenge
- Tackled at the end or in-production
- Redundantly developed
- Inadequately tested and refined
- Often left to programmers
- UX designers not well integrated into development culture
- Range of design roles not understood nor invested in
Systemic problem of poor and inconsistent user interface
Poor UX an impediment to adoption & support challenge
Tackled at the end or in-production
Redundantly developed
Inadequately tested and refined
Often left to programmers
UX designers not well integrated into development culture
Range of design roles not understood nor invested in
Systemic problem of poor and inconsistent user interface

Poor UX an impediment to adoption & support challenge

Tackled at the end or in-production

Redundantly developed

Inadequately tested and refined

Often left to programmers

UX designers not well integrated into development culture

Range of design roles not understood nor invested in
State of the Union

Usability/Accessibility in Open Source Software

- Systemic problem of poor and inconsistent user interface
- Poor UX an impediment to adoption & support challenge
- Tackled at the end or in-production
- Redundantly developed
- Inadequately tested and refined
- Often left to programmers
- UX designers not well integrated into development culture
- Range of design roles not understood nor invested in
State of the Union
Usability/Accessibility in Open Source Software

- Systemic problem of poor and inconsistent user interface
- Poor UX an impediment to adoption & support challenge
- Tackled at the end or in-production
- Redundantly developed
- Inadequately tested and refined
- Often left to programmers
- UX designers not well integrated into development culture
- Range of design roles not understood nor invested in
Systemic problem of poor and inconsistent user interface
- Poor UX an impediment to adoption & support challenge
- Tackled at the end or in-production
- Redundantly developed
- Inadequately tested and refined
- Often left to programmers
- UX designers not well integrated into development culture
- Range of design roles not understood nor invested in

State of the Union

Usability/Accessibility in Open Source Software

User Experience
Interaction Design
Interface Design
Visual Design
UI Developer
Application Developer
Software Architect

Accessibility Expertise
State of the Union

Usability/Accessibility in Open Source Software

- Systemic problem of poor and inconsistent user interface
- Poor UX an impediment to adoption & support challenge
- Tackled at the end or in-production
- Redundantly developed
- Inadequately tested and refined
- Often left to programmers
- UX designers not well integrated into development culture
- Range of design roles not understood nor invested in
What are we Building?

- Rich, flexible, reusable user interface components
- Lightweight JavaScript development tools
- User Experience Toolkit
- Great Interaction Designs

Interconnected activities
<table>
<thead>
<tr>
<th>Silo-delete</th>
<th>Fluid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sakai</td>
<td>uPortal</td>
</tr>
<tr>
<td>Moodle</td>
<td>OpenCollection</td>
</tr>
<tr>
<td>Kuali</td>
<td></td>
</tr>
</tbody>
</table>
Silo-delete

Sakai | uPortal | Moodle | OpenCollection | Kuali

Fluid
Designing software that works - for everyone
Silo-delete

Sakai   uPortal   Moodle   OpenCollection   Kuali

CONTENT MANAGEMENT
Silo-delete

File Uploading
Reworked, lightweight File Picking
Tagging and Tag Clouds
Smart folders, “playlists,” contextual filtering
Favourites and Clipboard/File Basket
Infrastructure: Accessible Thick Box, Tree, Sortable Tabs
Drag and drop portlets

Sakai | uPortal | Moodle | OpenCollection | Kuali
UX Toolkit

• User Research Tools
  – Contextual Inquiry
  – Persona development

• UI Design Patterns:
  – Open Source Design Patterns Library
  – Shared design advice and guidance

• UX Walkthroughs:
  – Tools for assessing your user experience

• Testing techniques and guidelines
  – How to test your designs and talk to users

• User profile library:
  – Understanding higher education users and beyond

All the stuff you need to design great interfaces in OSS!
UX Toolkit

User Research Tools
• Contextual Inquiry
• Persona development

UI Design Patterns:
• Open Source Design Patterns Library
• Shared design advice and guidance

UX Walkthroughs:
• Tools for assessing your user experience
• Testing techniques and guidelines

User profile library:
• Understanding higher education users and beyond

All the stuff you need to design great interfaces in OSS!
Contextual inquiry is a cross between interviews and observation and combines the strengths of both. In a contextual inquiry, the interviewer goes to the user and interviews them where they do their work -- sometimes referred to as "in the wild." The idea is to interview users in the context of their work, while they are performing their tasks, asking them questions about what they are doing and why (when necessary) along the way.
UI Design Patterns

• A pattern is a proven solution to a common problem in a specified context

• Practical tool to help designers and developers choose the right interface for the job

• Open Source Design Patterns Library:
  – The first truly open, collaborative pattern repository
  – Map to components

• Share patterns across communities
  – Tag, customize, adapt for your context

• Advice on how to use Fluid components
UI Design Patterns

- A pattern is a proven solution to a common problem in a specified context.

- Practical tool to help designers and developers choose the right interface for the job.

- Open Source Design Patterns Library:
  - The first truly open, collaborative pattern repository
  - Map to components
  - Share patterns across communities
  - Tag, customize, adapt for your context

- Advice on how to use Fluid components
UX Walkthroughs

• What are they?
  – Assess where you are: identify user pain points
  – Identify “componentizable” solutions
    • do the same thing across apps
  – Drive development priorities

• What is Fluid providing?
  – Simple, approachable techniques for usability and accessibility assessment
  – A tool that communities can use to assess their own usability and accessibility
    • Anyone can do a UX walkthrough!
    • Try out our checklists and heuristics
UX Walkthroughs

• What are they?
  – Assess where you are: identify user pain points
  – Identify "componentizable" solutions
do the same thing across apps
  – Drive development priorities

• What is Fluid providing?
  – Simple, approachable techniques for usability and accessibility assessment
  – A tool that communities can use to assess their own usability and accessibility
  – Anyone can do a UX walkthrough!
  – Try out our checklists and heuristics

Fluid UX Walkthrough General Protocol and Approach

Introduction

This document provides background information on performing a UX walkthrough. Detailed directions for performing an evaluation on a particular system can be found on the appropriate Protocols and Checklists pages.

The Fluid Approach

The Fluid approach to UX walkthroughs is a kind of combination of heuristic evaluation and cognitive walkthrough. Heuristic evaluation is done by examining the interface to a system as whole, and hence isn't specific to any particular user task. Cognitive walkthroughs are done by tracing the user actions, and associated cues and feedback, for one or more particular tasks, and as such, don't attempt to cover the entire interface for a nontrivial system. On the other hand, a cognitive walkthrough can pick up issues in working through a task that can be hard to detect when examining the interface as a whole, relating especially to the adequacy of cues and feedback in context.

Our aim for Fluid, "software that works - for everyone", takes in accessibility as well as usability. Rather than having two separate inspections, we want to have a unified inspection that addresses both areas, if we can.

Choosing what to examine

For a product, we want to examine tasks and subtasks that use product, that have some functional affinity, that is, parts that do the same kind of thing. We believe
Designing Components

- Components are recurring interactions:
  - Navigation: wizards, sequences, workflows
  - Content: file management, uploading, finding content
  - Direct manipulation of objects
- They are often larger than “widgets”
- Choosing components is based on:
  - Talking with users (all that user research!)
  - Analysis of existing applications across projects
  - Solving the most frequent and severe problems
- Some Examples...
Designing Components

- Components are recurring interactions:
  - Navigation: wizards, sequences, workflows
  - Content: file management, uploading, finding content
  - Direct manipulation of objects

- They are often larger than "widgets"

- Choosing components is based on:
  - Talking with users (all that user research!)
  - Analysis of existing applications across projects
  - Solving the most frequent and severe problems

- Some Examples...
Designing Components

• Components are recurring interactions:
  – Navigation: wizards, sequences, workflows
  – Content: file management, uploading, finding content
  – Direct manipulation of objects

• They are often larger than “widgets”

• Choosing components is based on:
  – Talking with users (all that user research!)
  – Analysis of existing applications across projects
  – Solving the most frequent and severe problems

• Some Examples...
File Upload

1) Choose Images

Add image to All Images:

Also add images to the current collection: <collection name>

File Queue:

<table>
<thead>
<tr>
<th>File Name</th>
<th>Size</th>
<th>Status</th>
<th>Remove</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image File 1.jpg</td>
<td>4.78MB</td>
<td>Ready to Upload</td>
<td></td>
</tr>
<tr>
<td>Image File 1.jpg</td>
<td>4.78MB</td>
<td>Ready to Upload</td>
<td></td>
</tr>
<tr>
<td>Image File 1.jpg</td>
<td>4.78MB</td>
<td>Ready to Upload</td>
<td></td>
</tr>
<tr>
<td>Image File 1.jpg</td>
<td>4.78MB</td>
<td>Ready to Upload</td>
<td></td>
</tr>
<tr>
<td>Image File 1.jpg</td>
<td>4.78MB</td>
<td>Ready to Upload</td>
<td></td>
</tr>
<tr>
<td>Image File 1.jpg</td>
<td>4.78MB</td>
<td>Ready to Upload</td>
<td></td>
</tr>
<tr>
<td>Image File 1.jpg</td>
<td>4.78MB</td>
<td>Ready to Upload</td>
<td></td>
</tr>
<tr>
<td>Image File 1.jpg</td>
<td>4.78MB</td>
<td>Ready to Upload</td>
<td></td>
</tr>
</tbody>
</table>

Add more files...  

Upload
<table>
<thead>
<tr>
<th>Name</th>
<th>User ID</th>
<th>Email Address</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Campeau, Patrick</td>
<td>15234314</td>
<td><a href="mailto:pcampe@berkeley.edu">pcampe@berkeley.edu</a></td>
<td>Student</td>
</tr>
<tr>
<td>Capovilla, Megan</td>
<td>19269508</td>
<td><a href="mailto:megan@berkeley.edu">megan@berkeley.edu</a></td>
<td>Student</td>
</tr>
<tr>
<td>Catania, Natalie</td>
<td>19435570</td>
<td><a href="mailto:ncatania@berkeley.edu">ncatania@berkeley.edu</a></td>
<td>Student</td>
</tr>
<tr>
<td>Chalhoub, George</td>
<td>19206726</td>
<td><a href="mailto:gchalhoub@berkeley.edu">gchalhoub@berkeley.edu</a></td>
<td>Student</td>
</tr>
<tr>
<td>Chan, Julie</td>
<td>19206726</td>
<td><a href="mailto:jchan@berkeley.edu">jchan@berkeley.edu</a></td>
<td>Instructor</td>
</tr>
<tr>
<td>Choi, Daniel</td>
<td>16984247</td>
<td><a href="mailto:dchoi@berkeley.edu">dchoi@berkeley.edu</a></td>
<td>Student</td>
</tr>
<tr>
<td>Clarkson, Ian</td>
<td>19286942</td>
<td><a href="mailto:iclarkson@berkeley.edu">iclarkson@berkeley.edu</a></td>
<td>Student</td>
</tr>
<tr>
<td>Cole, Paul</td>
<td>19446940</td>
<td><a href="mailto:pcole@berkeley.edu">pcole@berkeley.edu</a></td>
<td>Student</td>
</tr>
<tr>
<td>Colgate, Abbey</td>
<td>14644786</td>
<td><a href="mailto:acolgate@berkeley.edu">acolgate@berkeley.edu</a></td>
<td>Instructor</td>
</tr>
<tr>
<td>Colville, Tony</td>
<td>19283731</td>
<td><a href="mailto:tcolville@berkeley.edu">tcolville@berkeley.edu</a></td>
<td>Student</td>
</tr>
</tbody>
</table>
Time/Date Picker

Click

February 21, 2008
Open on: 2/21/2008 11:30 AM

February 22, 2008
Due on: 2/22/2008 7:30 PM

Click

February 21, 2008
Open on: 2/21/2008 11:30 AM

February 22, 2008
Due on: 2/22/2008 7:30 PM
Component Roadmap

- Focus on file management and navigation
- “My files available from anywhere”
- Embeddable components:
  - File Uploading
  - Reworked, lightweight File Picking
  - Tagging and Tag Clouds
  - Smart folders, “playlists,” contextual filtering
  - Favourites and Clipboard/File Basket
  - Infrastructure: Accessible Thick Box, Tree, Sortable Tabs
  - Drag and drop portlets
1) Choose Images

Choose Images from:
- Your computer
- Images in this site
- Images in another site

Browse to upload images. Use the control and shift keyboard shortcuts to choose multiple images at once.

Flash Version 8 required for optimal viewing. Download now or go to the basic upload view.
Choose Images from:

- Your computer
- Images in this site
- Images in another site

Browse to upload images. Use the control and shift keyboard shortcuts to choose multiple images at once.

Flash Version 8 required for optimal viewing. Download now or go to the basic upload view.
Choose Images from:
- Your computer
- Images in this site
- Images in another site

Browse to upload images. Use the control and shift keyboard shortcuts to choose multiple images at once.

Flash Version 8 required for optimal viewing. Download now or go to the basic upload view.
U-Camps

• Our main educational effort
• Everyone should have a basic UX vocabulary
• Share a repertoire of viable UX techniques
• Opportunity for designers and developers to collaborate
• Loose agenda, open participation
U-Camps

- Our main educational effort
- Everyone should have a basic UX vocabulary
- Share a repertoire of viable UX techniques
- Opportunity for designers and developers to collaborate
- Loose agenda, open participation

2. 80/20 rule
   - Less is more
   - Don't try to fit it all in
     - Leave out edge cases
     - Hide advanced or less often used functionality
U-Camps

• Our main educational effort
• Everyone should have a basic UX vocabulary
• Share a repertoire of viable UX techniques
• Opportunity for designers and developers to collaborate
• Loose agenda, open participation

4. Find Out What’s Generally True About Your Users

• Learn about enough users to separate out quirks from common behavior patterns
• Loudest users likely have unusual requirements
• Don’t include idiosyncratic requirements
U-Camps

- Our main educational effort
- Everyone should have a basic UX vocabulary
- Share a repertoire of viable UX techniques
- Opportunity for designers and developers to collaborate
- Loose agenda, open participation
U-Camps

- Our main educational effort
- Everyone should have a basic UX vocabulary
- Share a repertoire of viable UX techniques
- Opportunity for designers and developers to collaborate
- Loose agenda, open participation
Distributed Design
Distributed Design
Distributed Design

Camera and Voice

To Everyone

Call Phones or Send SMS

Fluid Work

Designing software that works - for everyone
Distributed Design
Virtual Usability Lab

- Open source distributed usability testing
- Competition to expensive tools like Morae
- Powerful tool for usability testing
  - Before and after survey questions
  - Remote screen recording
  - No installation required
  - Mouse and keyboard tracking
  - Designed within a community that needs it!

http://www.vulab.ca/
Technical Goals

- Make it easier for developers to build better, more accessible user interfaces
- Support collaboration with designers
- Foster sharing of design and code
- Adaptable for a variety of tools and workflows
- Embrace the Web
- Support diverse presentation frameworks
- Don't reinvent the wheel: leverage good existing technologies and fill the gaps
Component Composition (Flexible and reusable)

Easy to modify!
Fluid Framework Illustrated

Components

UI Adaptation Engine
Reorderer
Text Editing Service
Dependency Management
Template Renderer
Accessibility Plugins
DOM Binding
Universal View Bus (AJAX)
jQuery
Roadmap

We welcome participation!

June 2008
Fluid 0.4
- toolkit accessibility
- framework definition
- ui accessibility adoption
- file management components

December 2008
Fluid 0.8
- sustainable, active community

Early Adoption; Help Shape the Direction of Fluid

Build and Share Your Own Fluid Components

March 2008
Fluid 0.3
- framework development
- open design patterns library

September 2008
Fluid 0.6
- new components, framework refinements
- expand UX toolkit

March 2009
Fluid 1.0

We welcome participation!
How You Can Help

• Join our mailing lists
• Share code
• Help with design effort
  – UX Walkthroughs are fun and easy
  – Contextual inquiry
  – Component design teams
• Use and extend Fluid components in your tools
• QA: design test plans, help with testing
• User testing
• Share design patterns
Join in….

- Fluid Project Web Site: [http://fluidproject.org](http://fluidproject.org)
- Our wiki: [http://wiki.fluidproject.org](http://wiki.fluidproject.org)
- Our source code: [https://source.fluidproject.org/svn](https://source.fluidproject.org/svn)
- Our mailing lists: fluid-work@ for community collaboration fluid-talk@ for anything you’re interested in