The Fluid Project: Overview and Update

Colin Clark, Fluid Project Technical Lead, Adaptive Technology Resource Centre, University of Toronto
Daphne Ogle, Senior User Interaction Designer, UC Berkeley
Allison Bloodworth, Senior User Interaction Designer, UC Berkeley
Topics We'll Cover

• What is Fluid?
• Overview of the Fluid 0.1 release
• UX Toolkit: walkthroughs, patterns, and more
• Component architecture
• Project roadmap
• How to get involved
What is the Fluid Project?
Vision

• Foster user experience design and development within academic community source projects
• …so that they can fulfill their potential as platforms for innovation
• Create a community of UX expertise
• Build a presentation layer that supports:
  – Cross-application sharing of designs
  – The diversity of needs within higher education
• Support the precarious values of usability, accessibility, and more
UX in Community Source

• UX is a challenge for all open source projects
• Fluid is looking at common problems:
  – 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?
• Higher education has a unique opportunity:
  – Often co-located with our users
  – Gather ideas and feedback, literacy, user testing
  – We can build better user interfaces
Fluid's Approach

- Cross-project collaboration:
  - Share scarce UX resources across projects
  - Solve common challenges
  - Recognize recurring user interface idioms and needs
- A holistic approach: combine technology & design
- A two-fold path:
  - Social: build a community around UX
  - Technical: new UI development tools
Accessibility Vision

- Accessible software is better software for everyone
- Disability is an artifact of the environment:
  - A mismatch between the user and the system
- 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
Participating Projects

• Sakai
  – Collaboration and learning environment
  – Teaching, research, and group collaboration

• uPortal
  – Enterprise portal system
  – Aggregates personalized student information

• Moodle
  – Learning management system
  – Strong focus on pedagogy

• Kuali Student
  – Upcoming, next generation student system
  – Viable alternative to high-cost commercial products
Who Are We?

• Fluid is Sakai
• We’re members of this community and others
• All solving UX issues together
• Collaborate with tool and project teams
• One strand of activity among many
What are we Building?

• Rich, flexible DHTML user interface components
  – Components that work across applications
  – More than just widgets: activities and workflows
  – Easy to embed into tools and customize

• Lightweight JavaScript development framework
  – Wiring up component dependencies
  – JavaScript conveniences for DHTML accessibility
  – A simple set of supports: builds on existing toolkits

• User Experience Toolkit
• Interaction Designs
• Help integrating into Sakai and uPortal
Introducing Fluid 0.1...
What's in Fluid 0.1?

• A component: the Lightbox
• A slice of the framework: the Reorderer
• The UX Toolkit:
  – How to do a UX Walkthrough
  – Three drag and drop design patterns
• Lots of sample code and documentation
Why is 0.1 Important?

• Opens up our work to wider audience:
  – Download it, try it out, share your thoughts, get involved
• Established a community process:
  – Release manager, status updates, code freezes
  – Designers can stop a release if there are UX problems
  – Transparent voting for committers
• This is the first of six more release milestones on a roughly quarterly schedule
UX Toolkit
UX Toolkit

• All the stuff you need to design great interfaces
• Components and markup examples
• UI Design Patterns
  – Good advice when designing UIs
  – Take material from Tidwell, Yahoo!, and other patterns
• UX Walkthroughs & Distributed User Testing
  – Driven by the communities, but collaborative
  – Reusable protocol and checklist
  – VULab
• User Persona Library
Components

Components are recurring interactions:

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

Choosing components is based on:

- Talking with users
- Analysis of existing applications across projects
- Solving the most frequent and severe problems
UX Walkthroughs: Goals

• Assess what we've got: identify user pain points
• Identify “componentizable” solutions
• Drive our development priorities
• Share simple, approachable techniques for usability and accessibility assessment
• Anyone can do a UX walkthrough
  – Try out our checklists and heuristics
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
Virtual Usability Lab

- Open source distributed usability testing
- Competition to expensive tools like Morae
- Before and after survey questions
- Remote screen recording
- No installation required
- Mouse and keyboard tracking
- Designed within a community that needs it!
Open Source Design Patterns Library
How Patterns Can Help You

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

• Patterns serve as tools to communicate ideas, solutions, and knowledge about commonly recurring design problems

• A *pattern language* is a collection of related patterns which provide a method for choosing design solutions in a field of expertise

• User interface design patterns can help you create more usable *and* innovative user interfaces!

• Sakai UI Design Pattern Library: [http://confluence.sakaiproject.org/confluence/display/DESPAT/Library](http://confluence.sakaiproject.org/confluence/display/DESPAT/Library)
Open Source Design Pattern Library Goals

• Provide a practical tool to help designers and developers create the proper user interface for a specific context

• Specifically,
  – Providing guidance on proper implementation of Fluid components
  – Documenting patterns found in Fluid applications
  – Space for Fluid communities to grow their own design pattern libraries
  – Bringing together patterns from other collections (e.g. Tidwell, Van Welie) which are helpful to Fluid applications
  – Potentially, collect patterns from other open-source communities, or even organizations which would like to open-source their patterns

• http://wiki.fluidproject.org/display/fluid/Open+Source+Design+Pattern+Library
Drag and Drop Pattern

Problem Summary
The user needs a way to visually change the order or manipulate the position of elements on the screen.

Solution
Provide drag and drop capability on the page, so that the user can use a mouse gesture or keyboard key-combination to re-order items directly on the screen.

Use When
- You are displaying a list of items on a single page, and you want to give the user the ability to change the order or position of the items.
- You want to allow the user to change the order or position of items in context, without having to go to another page.

Why
Get involved with Design Patterns!

" Talk to Allison during the conference
" Check out the varied info on design patterns on our wiki: http://wiki.fluidproject.org/display/fluid/Design+Patterns

• Join Fluid Design Patterns group & start creating patterns
  – Join fluid-work@fluidproject.org mailing list
  – Join sakai-dg-ui@collab.sakaiproject.org mailing list
  – Email me (abloodworth@berkeley.edu) with your contact info

• Check out and create more Sakai design patterns: http://confluence.sakaiproject.org/confluence/display/DESPAT/Library

• Sign up to help us with competitive analysis of other pattern libraries

• Help with development of (Drupal) content management framework for pattern library
Welcome to your personal workspace.

In Sakai each user has his or her own individual worksite called My Workspace. My Workspace is a place where you can keep personal documents, create new sites, maintain a schedule, store resources, and much more.

The default information displayed here for a new user can be modified by the Sakai Administrator by editing the file sakai.properties configuration value "myworkspace.info.uri" to point to the html file desired.
Sakai version 2.5 w/new More Sites design
Sakai’s ‘More Tabs’ re-design

• We’ve completed the re-skinning work the tabs
• Next steps:
  – Finish partial re-design of Customize Tabs to match new interface when 10448 is turned on (SAK-11460) - December
  – Update Help content to reflect changes (SAK-11963) - December
  – Finish adding accessibility semantics (SAK-11540) - 2.5.1
  – Add Customize Tabs link to tab row - post 2.5
  – Do full re-design of Customize Tabs to match interface when 10448 is turned on
  – Create a re-usable Fluid tab component?
• We’d love to have your help with coding our designs!

• More info: http://confluence.sakaiproject.org/confluence/display/UI/Sakai+%27More%27+Tab+Design
Component Architecture
Technical Goals

• Make it easier for developers to build better, more accessible user interfaces
• Support collaboration with designers
• Make it easier to share designs within a community
• Enable components to be adapted 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
What is a Fluid Component?

• On the client-side, a Fluid component consists of:
  – One or more HTML templates
  – One or more layers of CSS
  – JavaScript for behavioural logic
  – Accessibility metadata (control, presentation, etc)

• And on the server-side:
  – Binding conventions: markup with known, formal IDs
  – The ability to deliver the appropriate markup to the client and respond to its requests
Anatomy of a Component

Anatomy of a Fluid Component

- **CSS**
  - styles
  - colours
  - sizes
  - fonts
  - borders
  - focus
  - etc.

- **HTML**
  - structure
  - ARIA metadata
  - labels
  - controls

- **JavaScript**
  - UI logic
  - client-side validation

- **Metadata**
  - detailed roles
  - modality
  - control types

Designing software that works – for everyone
A Flexible Framework

• Solve the need for reuse and accessibility together
• Components need to adapt to different contexts:
  – Available screen real estate
  – Type of content
  – Amount of content
  – Method of control and navigation
• Leverage the web’s strength in separating structure from presentation:
  – Alternative HTML templates
  – Alternative stylesheets
• Augment with alternative behaviours
Huh?

- Flexible layouts
- Enhanced navigation options
- Alternative control strategies
Composition = Flexibility

• Fluid components are built out of smaller units
  – Keyboard handlers
  – Layout managers
  – Server callbacks

• Composition enables flexibility
  – At runtime, wire up alternative behaviour
  – Use web standards to change presentation (HTML/CSS)

• Easy to extend or modify component behaviour
Component Composition

Fluid Component

- HTML
- CSS
- Layout Behaviour
- Server Callbacks
- Keyboard Mappings
The Fluid Framework

• Common accessibility APIs:
  − Focus management
  − Keyboard handlers
  − Getting/setting ARIA properties

• Framework infrastructure:
  − Dependency injection
  − Server-side communication
  − Portal-friendly DOM conventions

• Adaptation:
  − The ability to wire up component behaviour at runtime
Fluid Framework Illustrated

The Fluid Framework

- Components
  - Dependency Management
  - Component Adaption
  - Accessibility Plugin
  - DOM Accessors
  - Server Callbacks (AJAX)
  - jQuery
Markup Contracts

• The DOM is a stew of potentially diverse elements
• Finding your own stuff is a challenge
• Solution: semantic, namespaced IDs
  – Unique placement prefixes: portal namespacing
  – Reflects containment relationships
  – Describes what the markup represents
• IDs become a stable way to communicate with server
  id="gallery::lightbox::thumbnail::orderable:0:"
• Framework will help build the necessary jQuery selectors
Accessibility API

- jQuery.a11y.keyboard
  .makeTabFocussable(elements)
  .makeSelectable(container, elements, direction)
  .makeActivatable(elements, function)
- jQuery.a11y.aria
  .getRole()
  .setRole(elements)
  .addState(elements, state)
  .removeState(elements, state)
Heuristics for Building a Fluid Component

• Favour markup over messaging
  – Use HTML as data format
  – Use standard form POSTs to help degrade gracefully
• Break layout logic into separate units
• Provide configurable keyboard mappings
• Pass all dependencies in on initialization, at least until we have a bit of IoC support
• Allow for template and markup variations
Project Roadmap
Release Plan

• Quarterly milestone releases
• Whole package:
  – Components, framework, UX Toolkit, Documentation
• Major Goals:
  – User research
  – Delve into file management components
  – Viable framework: everyone can build components
  – Launch the Open Source Design Patterns Library
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
Instructor organiser view (pops up as helper in any site)

<table>
<thead>
<tr>
<th>Recent</th>
<th>Uploaded by me</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>My Tags</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Actresses</td>
<td>my file 1.doc</td>
</tr>
<tr>
<td>Playhouses</td>
<td>my file 2.doc</td>
</tr>
<tr>
<td>Dramatists</td>
<td>my file 3.doc</td>
</tr>
<tr>
<td>Audiences</td>
<td>my file 1.doc</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Week 1</strong></th>
<th><strong>Week 2</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>My Sites</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Drama 101</td>
<td>my file 1.doc</td>
</tr>
<tr>
<td>Drama 102</td>
<td>my file 2.doc</td>
</tr>
<tr>
<td>Drama 103</td>
<td>my file 3.doc</td>
</tr>
<tr>
<td>Drama 104</td>
<td>my file 1.doc</td>
</tr>
</tbody>
</table>

student syllabus view (a tool in a site)

**FIRST IDEAS !**
Image Gallery Project Goals

• Deliver small scale local pilot to UCB campus starting no later than fall 2008
• Contribute to and leverage work in the Fluid project to develop a component for file upload
• Explore the relationship between file management and presentation within Sakai to help evolve UX in this area
Image Gallery Design Goals

- Easy upload from within Image Gallery
  - from computer, current or other site
  - *Image Uploader* incubator for abstract *Uploader* component
- Increased dynamic interaction of image and collection organization using "playlist" metaphor
- Add own context to images & collections through tagging, “collection-lists”, meta-data
  - Social networking later phases
- General usability clean up
Multi-step Indicator Component

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.
1) Choose Images ➔ 2) Add Images to Image Gallery ➔ 3) Add Information to Images

Choose Images from:
- Your computer
- Resources in this site
- Resources in another site you own

Browse to upload images.

Add More Images

Continue  Back  Cancel
1) Choose Images  →  2) Add Images to Image Gallery  →  3) Add Information to Images

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

Choose images from resources

<table>
<thead>
<tr>
<th>Title</th>
<th>File Type</th>
<th>Created by</th>
<th>Created Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filename</td>
<td>jpg</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>Folder name</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Folder name</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Folder name</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Filename</td>
<td>png</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>Filename</td>
<td>jpg</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>Filename</td>
<td>gif</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>Filename</td>
<td>gif</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>Filename</td>
<td>jpg</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
</tbody>
</table>

Would be great if this was all my image files & it shows me where they’ve been used along with some of this other information.
1) Choose Images

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

Choose Images in:

Choose Site

From Resources

<table>
<thead>
<tr>
<th>Title</th>
<th>File Type</th>
<th>Created by</th>
<th>Created Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Folder name]</td>
<td>jpg</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>[Folder name]</td>
<td>png</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>[Folder name]</td>
<td>jpg</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>[Folder name]</td>
<td>png</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>[Folder name]</td>
<td>jpg</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
</tbody>
</table>

From Image Gallery

<table>
<thead>
<tr>
<th>Collection</th>
<th>Title</th>
<th>File Type</th>
<th>Created by</th>
<th>Created Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Images (228)</td>
<td>[Folder name]</td>
<td>png</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>Recently Added (5)</td>
<td>[Folder name]</td>
<td>jpg</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>Flagged (118)</td>
<td>[Folder name]</td>
<td>png</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
<tr>
<td>Collection 1 (22)</td>
<td>[Folder name]</td>
<td>jpg</td>
<td>[Creator]</td>
<td>[Created date]</td>
</tr>
</tbody>
</table>

Choose Files  Cancel
1) Choose Images → 2) Add Images to Image Gallery → 3) Add Information to Images

Tag All These Images

Tags:

For multi-word tags use quotes (i.e. "image tool") or underscore (i.e. image_tool)

Your previously used tags:

Add to Collection: Week 1

Add Images  Back  Cancel
What’s next

• Image Gallery, Fluid & Resources WG collaboration
  – Envision resources of the future - my files from anywhere
• Image Gallery work released sometime next summer
• Contextual inquiries inform problem space and component work
Fluid 0.2: January 2008

• Maintenance release
• Strengthens our existing code base, adding:
  – Production-friendly release packaging
  – Support for almost any markup you can throw at it
  – Nested Reorderers
  – Cleaner API for modifying drag & drop behaviour
  – Consolidation on jQuery

• How you can help:
  – bug fixes, write test markup, try out the Reorderer
Fluid 0.3: March 2008

- Contextual inquiry and design framework
- File Upload component
- Portlet Layout Manager (drag and drop portlets)
- Tag cloud component
- Several new design patterns
- How you can help:
  - Get involved in user research
  - Code, designs, testing for new components
  - Write a design pattern
Fluid 0.4: June 2008

• File Picker
• File Upload 2.0
• One other new component
• U-Camps
• New design patterns
• How you can help:
  – Join the U-Camp team
  – Code, designs, testing
  – Integrate components into your tools
Fluid 0.6: September 2008

- Focus on adaptation
- Landmark navigation component
- Extend existing components for adaptation
- User preferences editor
- Accessibility design patterns
- A new wave of UX walkthroughs, targeted at file management

How you can help:
- Accessibility testing
- Help with UX walkthroughs
- Component and framework development help
Fluid 0.8: December 2008

- U-Camps
- Updated design patterns CMS
- Two new components
- Lots of user testing
Fluid 1.0: March 2009

- Last wave of Sakai 3.0 integration and testing
- Content management for UX Toolkit
- Polished do-it-yourself UX Walkthrough kit
- Two new components
- Framework next steps
- How you can help:
  - Roll Fluid components into your tool
  - Coding, design, user testing
  - Help with QA effort
  - Contribute to vision for post-funding phase
Getting Involved
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
Wrapping Up

• Fluid Project Web Site:
  http://fluidproject.org

• Our wiki:
  http://wiki.fluidproject.org

• Our source code:
  https://source.fluidproject.org/svn

• Our mailing lists:
  fluid-work@ for community collaboration
  fluid-talk@ for anything you’re interested in