



Emerging Technologies

Introduction to Dojo

An Open Source, Accessible Web Development Toolkit

Becky Gibson
Web Accessibility Architect
Dojo Committer

Agenda

- What is Dojo
- Dojo Core Widget Set - Dijit
- Web 2.0 Accessibility Issues
- Accessibility Support in Dojo
 - Keyboard Support
 - Low Vision Support
 - Assistive Technology Support via ARIA
- ARIA Live Regions
- Building Accessible Applications with Dojo
- Questions

Dojo - What is it?

- Open Source JavaScript Toolkit
- “Easy” Ajax
- Data Binding
- Full event system
- Browser abstraction layer
- User Interface Widgets
- Dual Licensed
 - Academic Free License v2.1
 - BSD License
- Dojo 1.0 available in November, 2007
 - 1.1 in March, 2008
- Dojo 1.2 released October, 2008
 - 1.2.3 in December, 2008
- Dojo 1.3 planned for Q1 2009



Core Widget Set - dijit

- Accessible
- Internationalized
- Customizable Look and Feel
- Developer Documentation
 - API
 - User Manual
- Supported Browsers
 - Firefox 2, Firefox 3
 - IE 6, IE 7, IE 8 support in progress
 - Safari 3
- Data Binding
 - Tree, Grid*, Select

*Grid is in dojox (dojo extensions)

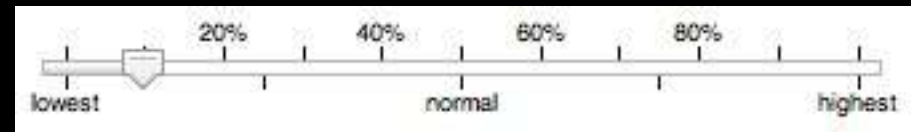
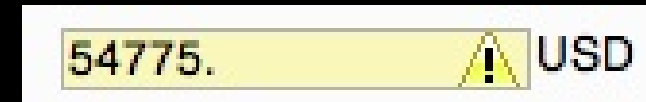
Dojo Core Widgets

- Form Widgets
- Layout Widgets
- Advanced Widgets

Form Widgets

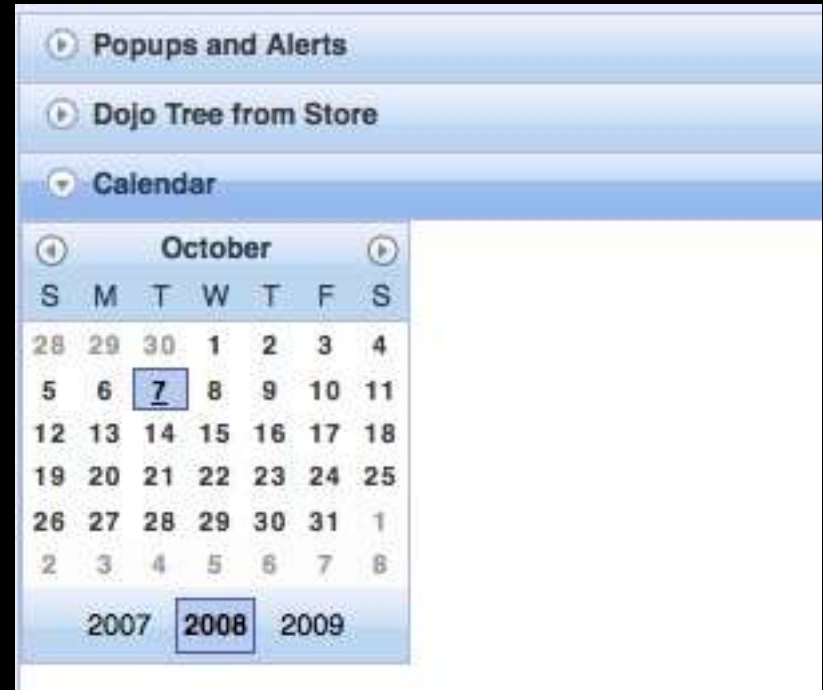


- Button, Dropdown Button, Combo Button
- Checkbox, Radio
- ComboBox, Filtering Select, Multi Select
- Textbox
- Currency & Integer Validation Textboxes
- Resizable Textarea
- Slider
- Integer Spinner
- Inline Editbox
- Dropdown Calendar



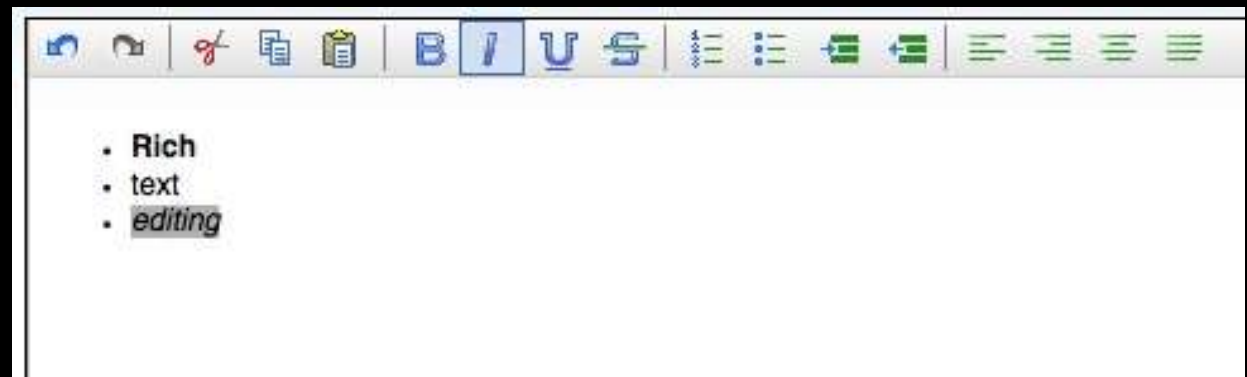
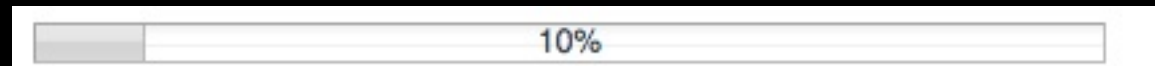
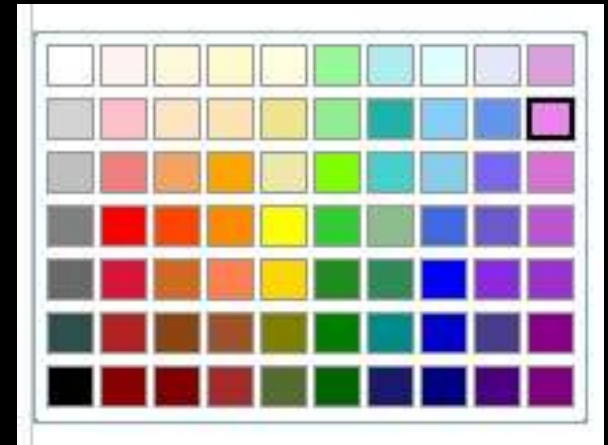
Layout Widgets

- Accordion Container
- Content Pane
- Dialog
- Border Container
- Tab Container
- Title Pane



Advanced Widgets

- Color Palette
- Context Menu
- Rich Text Editor
- Progress Bar
- Toolbar
- Tooltip, Tooltip Dialog
- Tree



Web 2.0 Accessibility Concerns

- Rich Interface controls
 - Reliance on Mouse
 - Lack of Semantics
- Incremental Updates via Ajax
- Changes in focus
- Excessive Navigation via Tab key
- Content aggregation from various resources (Mashups)
- Multimedia
 - Lack of captioning
 - Interaction issues
- Paradigm Shift

Solving Web 2.0 Accessibility Issues

- Provide Accommodations for
 - Vision
 - Hearing
 - Cognitive
- Semantics & identification
- Navigation & keyboard support
- Drag and Drop
- Incremental Updates via Ajax



Dojo Core Widgets Accessibility Support

- Provide Accommodations for
 - **Vision**
 - Hearing
 - Cognitive
- **Semantics & identification**
- **Navigation & keyboard support**
- Drag and Drop
- Incremental Updates via Ajax

Dojo core widgets support the issues presented in bold, underline, orange

Accessibility Support in Dojo Core Widgets - 1.0.x+

- Keyboard support in IE 6, IE 7, Firefox 2, Firefox 3
- Screen reader support in FF 2 & FF 3 with JAWS 9 or later
- Low Vision Support
 - Windows high contrast support in IE 6, IE 7, FF2, FF3
 - No fixed font sizes
 - Support Images off in FF
- Low Vision support allows claim of “equivalent facilitation” for US Section 508 requirement to run with CSS turned off.

Dojo Keyboard Support

- Implement focus handling via `tabindex` attribute
- Add key event handlers to implement navigation
 - Use `tab` key to navigate between components
 - Arrow and other key navigation within a component
- Keep keyboard and mouse interaction in-sync
- Set focus to elements, do NOT simulate focus via CSS
- DHTML Style Guide group is defining widget behaviors

Keyboard Demo

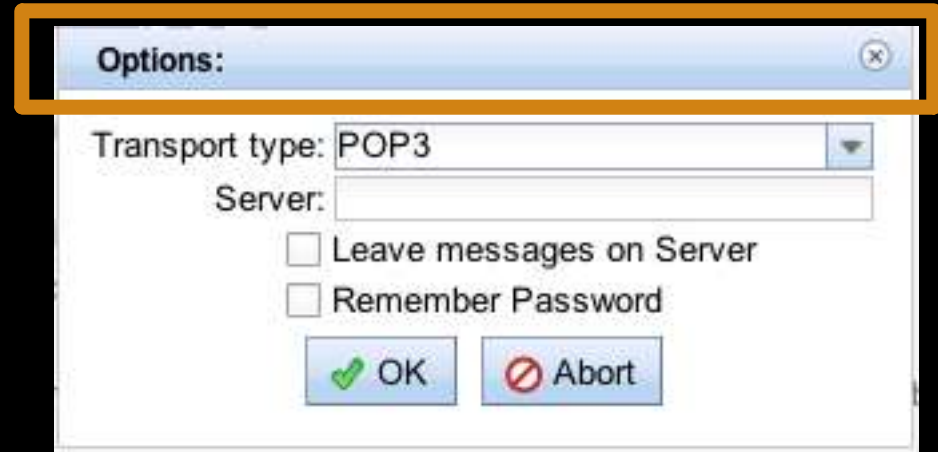
Some older but still valid pre-recorded demos available at
<http://www.weba11y.com/demos/index.html>

Dojo Low Vision Support

- Detect high contrast mode
 - Create and insert a div into the document
 - Assign different colors to left and right border of div
 - Examine rendered div - if left color==right color high contrast is on
- Detect Images off mode (FF only)
- dijit_a11y class to trigger style changes
- Provide text alternatives for CSS Background images
- No hard coded font sizes

High Contrast with CSS

Excerpt from Dialog template:
close icon in the dialog title



```
<div class="dijitDialog" waiRole="dialog">
```

...additional template code

```
  <div dojoAttachPoint="titleBar">
    <span id="{id}_title">{title}</span>
    <span class="dijitDialogCloseIcon"
      dojoAttachEvent="onclick: onCancel">
      <span dojoAttachPoint="closeText"
        class="closeText">x</span>
    </span>
  </div>
```


High Contrast with CSS

```
.tundra .dijitDialogCloseIcon {  
  background : url("images/tabClose.png");  
}
```

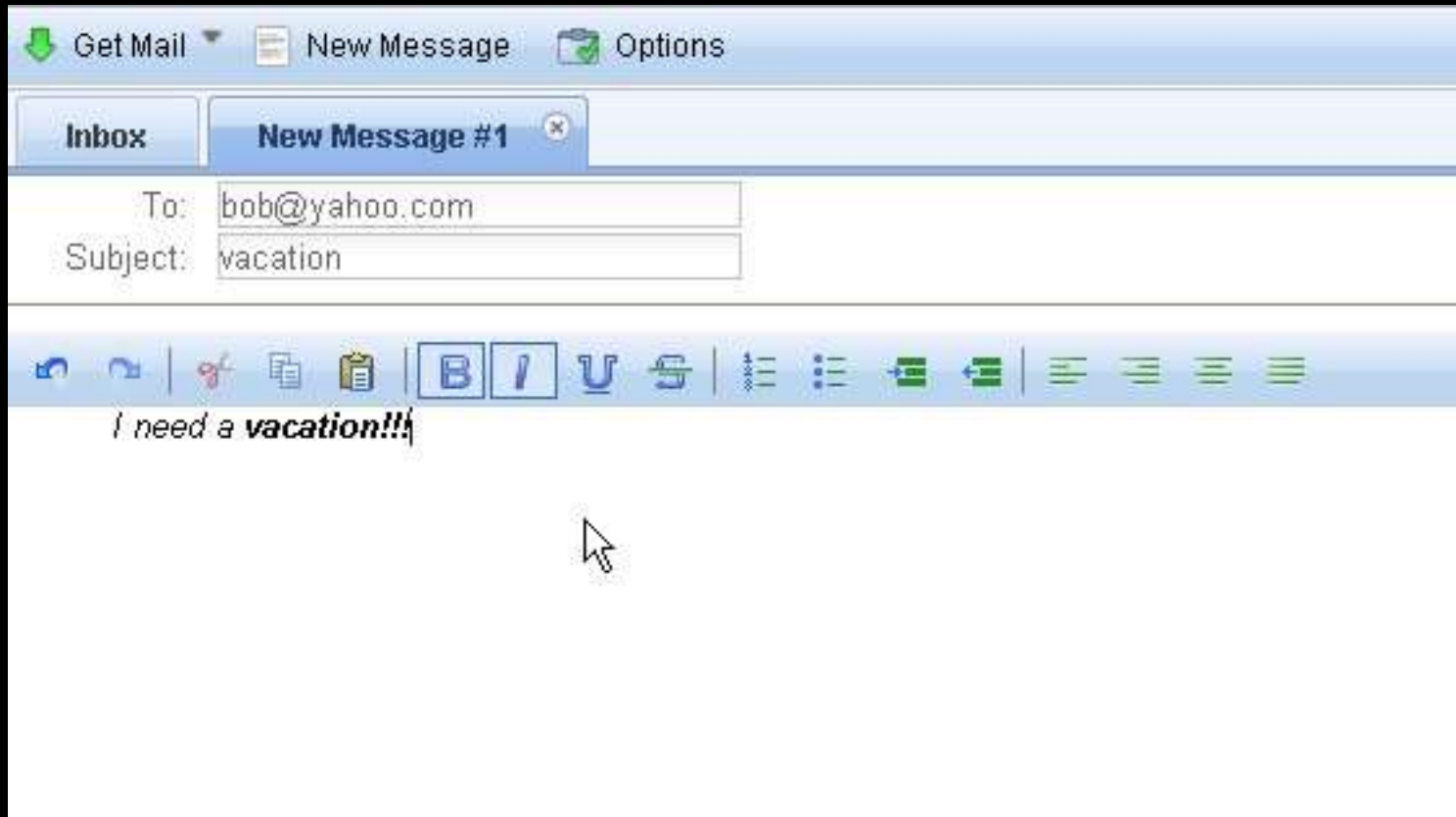
```
.dijitDialog .closeText {  
  display:none;  
}
```

```
.dijit_a11y .dijitDialog .closeText {  
  display:inline;  
}
```

Dialog in High Contrast Mode



Dojo Sample Compose Mail Page



Dojo Sample Compose Mail - High Contrast



Dojo Screen Reader Support

- Provide keyboard support
- Identify each widget type
- Indicate current state and properties of the widgets

Accessible Rich Internet Applications to the Rescue!



Accessible Rich Internet Applications (ARIA)

- Addresses Web 2.0 Accessibility Issues
- Developed at IBM and donated to W3C
 - Web Accessibility Initiative (WAI) Protocols & Formats Group
 - Nearing last call draft
- Implemented in Firefox browser
 - Being implemented in IE 8
 - Implemented Opera 9.5
- Supported by Window-Eyes & JAWS screen readers
- Used by IBM, Dojo, AOL, Yahoo, Google, SAP,

ARIA Support (current or in-progress)



ARIA Details

- Add role semantics to scripted user interface (UI) elements
- Update state information dynamically
- Make items focusable via tabindex attribute
- Add keyboard event handling
 - Mimic the keyboard behavior of the rich client UI
 - Minimize tab key navigation
- Add live region information and notification* to support Ajax

*Live region support initial implementation in Firefox 3 and JAWS 10

ARIA Example - Tree



Role = tree
(on outer container)

Role = treeitem
expanded=true
(on open Africa node)

Role = treeitem
selected=true
(on highlighted Egypt child node with no children)

Role = treeitem
expanded=false
(on closed Australia node)

ARIA Live Regions

- Perceivable sections are identified with region role
- Live indicates region is updated
 - Values of: Off, Polite, Assertive, Rude
- Atomic identifies the extent of updates
 - True – entire region is updated and relevant
 - False – only changed element needs to be presented to user
- Controls identifies the element which triggers an update
- Implemented in Firefox 3 with some support in JAWS 10

ARIA Examples - Live Region

- Editing auto save notification
 - live=polite; atomic=true;
- Server maintenance notification
 - Live=assertive; atomic=true;
- New mail - speak new entries
 - Live=polite; atomic=false;

Current Live Region/Ajax Best Practices

- Make Navigation Easy
- Place updated information after the trigger that causes the update
- Update existing elements with new content
- Be very careful about changing focus
- Avoid special notification
- Use Semantic HTML

Screen Reader Demo

Some older but still valid pre-recorded demos available at
<http://www.weba11y.com/demos/index.html>

Building Accessible Applications with Dojo

- Basic HTML accessibility
- Overall application navigation
- Create content in appropriate source code order
- Widget specific accessibility “features”

Application is More than the Sum of Parts


Basic HTML Accessibility

- Use semantic markup
- Label form elements - Dojo will respect any label elements
- Use standard radio grouping
- Be aware of sizing issues - don't use px

Label Form Elements

```
<label class="firstLabel" for="fname">Name *</label>  
<input type="text" id="fname" name="fname" class="medium"  
dojoType="dijit.form.ValidationTextBox" required="true"/>
```

Name And Address As you type in the text below, notice how your input is auto corrected and also the auto corr

Name *	<input type="text" value=""/>	
Address *	<input type="text" value="5 Technology Park Drive"/>	
City *	<input type="text" value="Westford"/>	State <input type="text" value="Massachusetts"/>

Use Standard Radio Grouping

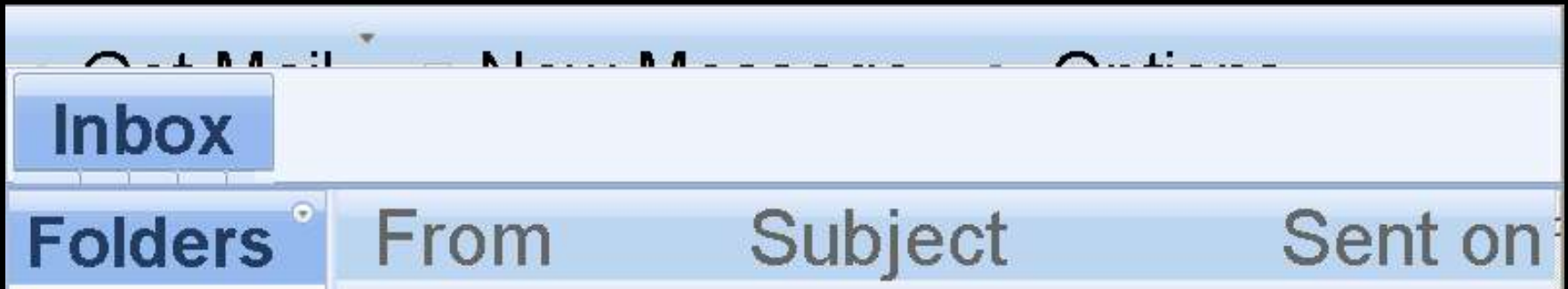
Dojo uses standard input type="radio" element with overlay image

```
<fieldset id="hours" class="dijitInline">
  <legend>Hours </legend>
  <input type="radio" name="hours" id="full"
    value="full" dojoType="dijit.form.RadioButton" />
  <label for="full">Full time</label>
  <input type="radio" name="hours" id="part"
    value="part" dojoType="dijit.form.RadioButton" />
  <label for="part">Part time</label>
</fieldset>
```



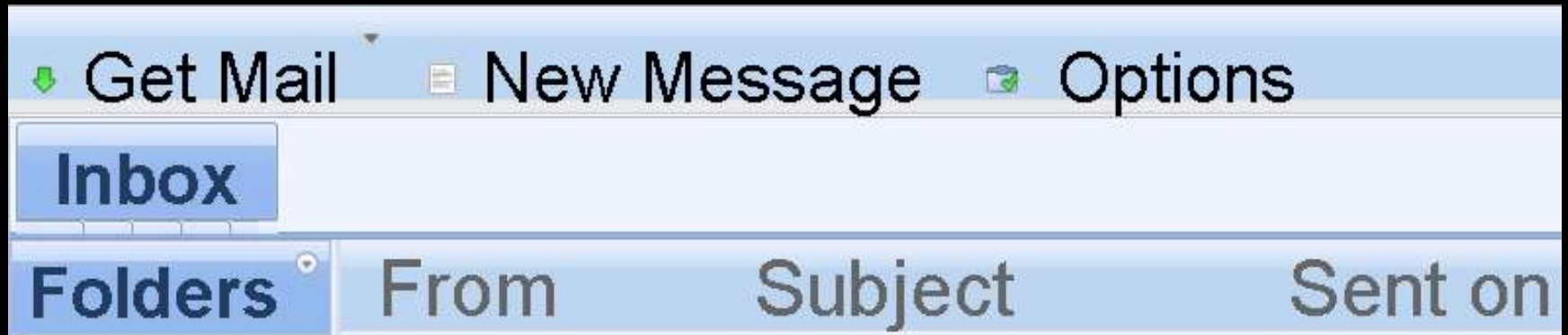
Be Aware of Sizing Issues and Large Fonts Incorrect

```
<div dojoType="dijit.Toolbar" region="top" style="height:25px;">
```



Correct

```
<div dojoType="dijit.Toolbar" region="top" style="height:1.25em;">
```



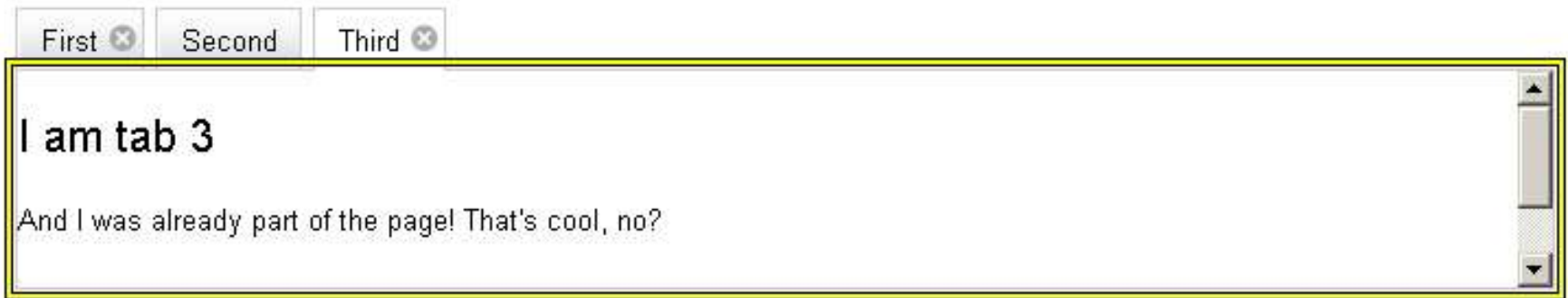
Overall Navigation

- Tab from component to component, key navigation within a component
- Be wary of assigning `tabindex>0`
- Dijit ContentPanels not added into tab order by default
 - Add `tabindex=0` on contentPanels with no focusable elements
 - Add `role=region` and `aria-live` and `aria-atomic` for panes with information that is updated and should be spoken by the screen reader
- Note that Firefox adds containers with overflow into the tab order
- Firefox 3 adds containers into the accessibility hierarchy
 - Mark containers as presentation or grouping as necessary

```
<div aria-role="presentation" class="formBackground">  
  <form>  
    .....form elements.....  
  </form>  
</div>
```

Dijit ContentPane Example

Inspect32 used to highlight focus (yellow outline) on tab panel container (implemented using dijit ContentPane) in Firefox.



Firefox puts the tab panel into the tab order - IE does not. If you want the panel in the tab order in all browsers, add `tabindex=0` in the `ContentPane` source.

```
<div dojoType="dijit.layout.ContentPane" title="Third"
onClose="testClose" closable="true" tabindex="0">
```

Dijit Live ContentPane Example

When the information in a ContentPane is updated via Ajax and should be spoken by the screen reader – add aria information to the source code.



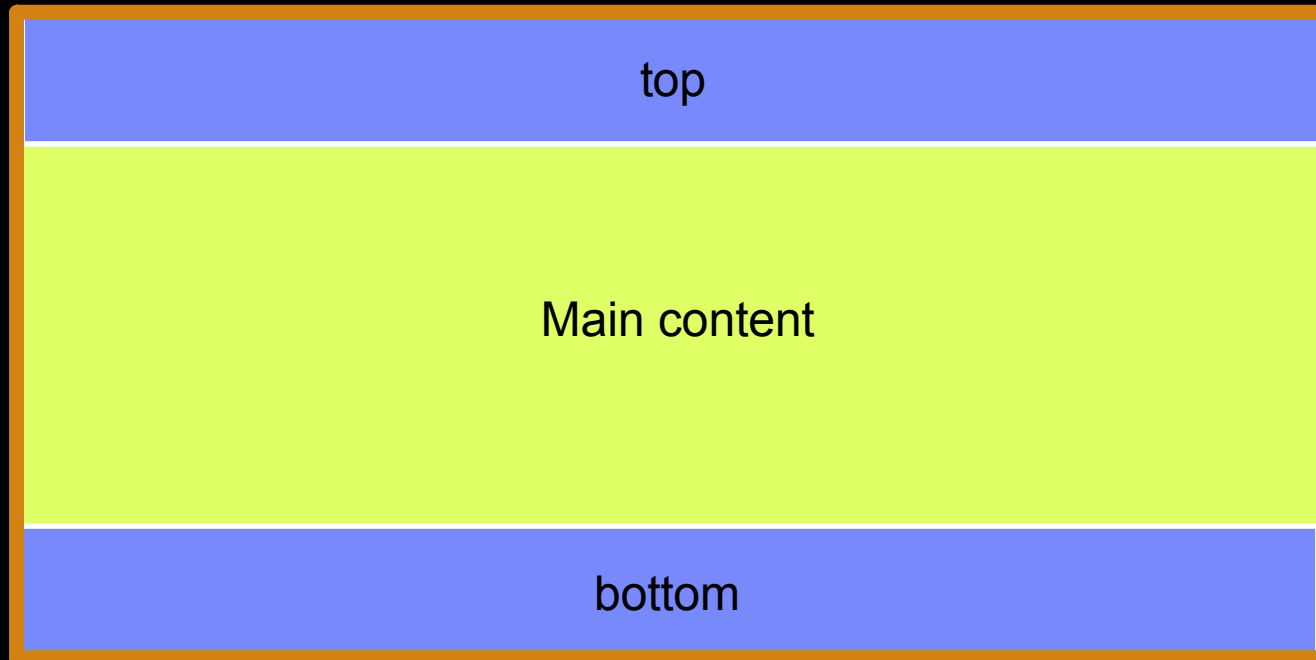
```
<div id="content" dojoType="dijit.layout.ContentPane" title="Content"
    role="region" aria-live="assertive" aria-atomic="true">
    initial text displayed in the pane
</div>
```

Source Code Order - Incorrect BorderContainer

```
<div dojoType="dijit.layout.BorderContainer" >  
  <div dojoType="dijit.layout.ContentPane" region="top" splitter="true">  
    top bar  
  </div>  
  <div dojoType="dijit.layout.ContentPane" region="bottom" splitter="true">  
    bottom bar  
  </div>  
  <div dojoType="dijit.layout.ContentPane" region="center">  
    main content  
  </div>  
</div>
```

INCORRECT source code order, bottom region should be below main content!

Rendered BorderContainer



Correct tab order is top, main content, bottom.

Code from previous page would create the incorrect reading order of top, bottom, main content.

Source Code Order - Correct BorderLayout

```
<div dojoType="dijit.layout.BorderContainer" >  
  <div dojoType="dijit.layout.ContentPane" region="top" splitter="true">  
    top bar  
  </div>  
  <div dojoType="dijit.layout.ContentPane" region="center">  
    main content  
  </div>  
  <div dojoType="dijit.layout.ContentPane" region="bottom" splitter="true">  
    bottom bar  
  </div>  
</div>
```

Specific Dijit Accessibility Issues

- Dojo Book page for each widget contains an accessibility section
 - Provides key mapping
 - Describes any known AT issues
- Use optionsTitle parameter to label arrow portion of ComboButton
 - Spoken by Window-Eyes but not JAWS
 - In Firefox 3 ComboButton arrow and DropDownButton identified as “menu button”
- No dialog information spoken by JAWS 9, 10 in FF 2, works in FF3
 - Make sure user is aware that a dialog box will open

Specific Dijit Accessibility Issues - continued

- Slider text values are not spoken (will be addressed in future Dojo release)
- BorderContainer splitters are in the tab order to allow changing pane size via the keyboard
 - This may change in future releases
 - JAWS 9,10 identifies splitter as “separator”
- Tooltips
 - Display on mouseover or focus of trigger element
 - Removed on mouseout or blur of trigger element
 - Spoken by the screen reader
- Rich Text Editor

Rich Text Editor (RTE) Accessibility

- Waiting for AT to implement IAccessible2 API for full support
- When edit area gets focus, screen reader speaks “edit area”
- Select all text for JAWS to speak edit area contents
- In Firefox an extra press of tab key is required to tab out of RTE
 - Overrides FF2 implementation of tab key press working as indent within the editor and thus no keyboard mechanism to leave editor
 - With focus in editor first tab key sets focus to the editor frame, screen readers speaks “edit area frame”
 - Next key press takes user to next focusable element in the tab order

Summary of Web 2.0 Accessibility with Dojo

- ARIA spec not yet a standard – planned for 2009
- No ARIA support in Internet Explorer (yet) - Coming in IE8!
- Tabindex support being added to Safari - keyboard support coming soon
- Still a few accessibility quirks in Dijit Widgets
- Assistive Technology support needs to evolve
 - Issues with virtual browsing and interactive mode in screen readers
 - Additional support for live regions
 - Additional support for drag and drop

“The journey of a thousand miles starts with a single step.”

Resources

- Becky's Accessibility Presentations and Papers
 - <http://www.weba11y.com/Presentations/presentations.html>
- ARIA Roadmap, Best Practices, Primer, Specifications
 - <http://www.w3.org/wai/pf>
- Mozilla Developer Center - Firefox ARIA Information
 - http://developer.mozilla.org/en/docs/Accessible_DHTML
- Mailing list for ARIA issues
 - <http://lists.w3.org/Archives/Public/wai-xtech/>
- Dojo
 - Home - <http://www.dojotoolkit.org/>
 - Dojo Campus - <http://dojocampus.org>
 - Book - <http://www.dojotoolkit.org/book/dojo-book-1-0>
 - API Reference - <http://api.dojotoolkit.org>
 - 1.3 Book (in progress) - <http://docs.dojocampus.org>
- DHTML Style Guide Draft
 - http://dev.aol.com/dhtml_style_guide

Dojo Books in Print

- Dojo The Definitive Guide by Matthew Russel, O'Reilly Press
- Dojo Using the Dojo JavaScript Library to Build Ajax Applications by James E. Harmon, Addison-Wesley Developer's Library
- Mastering Dojo by Rawld Gill, Craig Riecke, Alex Russel, Pragmatic Press

