Bruce Lill
Kalechi Designs

Domino and Dojo

How to live with Dojo and 8.5

2009-3-3

Agenda

- Intro
- Overview
- Javascript Libraries
- Why Dojo?
- What is it
- How to use it
- Question & Answers

Intro

- Bruce Lill
  - Long time Notes developer / administrator – V2.0
  - Member of the Redbook Best Practices for Building Web Applications for Domino wiki
  - Presented here last year
    - Also presented at Lotusphere
  - Redbook Community Ambassador
    - We promote the Redbook and wiki content creation
  - Spend my time with Domino web development and DB2

Overview

- JavaScript Libraries are no longer just for leading edge web development.
- They were hard to learn
  - Easier to code yourself
- Now they provide benefit to all web developers
  - Cross-browser support – you don't have to do it!
  - Fairly stable – fewer bugs
  - Well documented
  - Well supported
  - Much easier to design with and use
  - Provide a base for others to build on.
  - Large number of features

JavaScript Libraries

- jQuery v1.2.6 – Nokia and MS support
- Sarrisa v0.9.9.4
- Prototype v1.6
- Scriptaculous v1.8.2
  - Provides visual effects for Prototype
- Spry v1.6.1 - Adobe
- YUI v2.6 - Yahoo
- EXT v2.2
- MooTools v1.2.1
- Dojo v1.2.3

5

jQery: DOM and CSS manipulation, Event handling, Ajax and Some visual effects

jQuery UI for fancy visual effects, used by Microsoft and Nokia as a core framework and will ship with Visual Studio and Nokia web runtime

EXT was based on YUI now you can use jQuery or Prototype

Why Dojo

- Because IBM supports it.
- Because it's on the 8.5 server
  - Version 1.1.1
- It's used in xPages widgets
  - DateTime, typeahead, views, rich text editor
- The new web rich text control is built with it
- iNotes
- Supports localization
- More will be coming in 8.5.1, 8.5.2.....

Dojo

- Dojo has 3 parts
- Core
  - Event handlers, DOM manipulation, graphical effects
- Dijit
  - Widgets – the really fun part!
- DojoX
  - complex widgets, graphing and charting

Dojo Install Locations

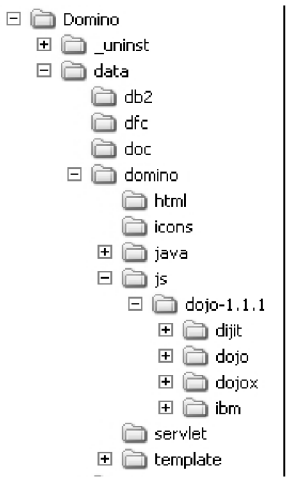
- To use the Domino 8.5 Version the url is
  - /domjs/dojo-1.1.1/dojo
- You can install Dojo under the HTML folder
  - /dojo-1.2.3/dojo
- You can use CDN like AOL
  - http://o.aolcdn.com/dojo/1.0.0
- You can import the Dojo files into a NSF as file resources.
  - /dbname.nsf/dojo
  - Use WebDav or download someone else copy.
  - Use Dojomino

8

Use the version number in the top folder name. You will be having different version and they are not all compatible If you can test all your applications then go for the simple name.

Dojo in Domino

- Domino 8.5 ships with version 1.1.1 of Dojo
- Located in data\domino\js\dojo-1.1.1



Web site documents

- domjs is defined on the web document

What to do with it

- You have Dojo on your server
- What do you do with it now?
- Add controls to web forms
  - Use the rich text editor control
  - Add Date or Time pickers
  - Field validations
  - Tool tips
  - Lots of web 2.0 features
- First thing is to test the install!

```
/domjs/dojo-1.1.1/dijit/themes/themeTester.html
```

11

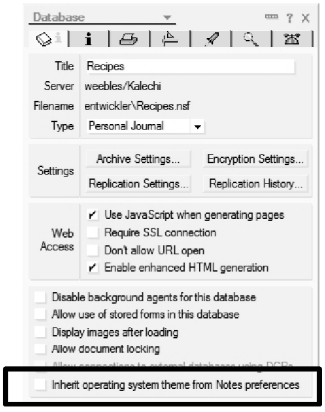
IBM uses themes to tie all the apps together.
So you can specify the theme for server or db

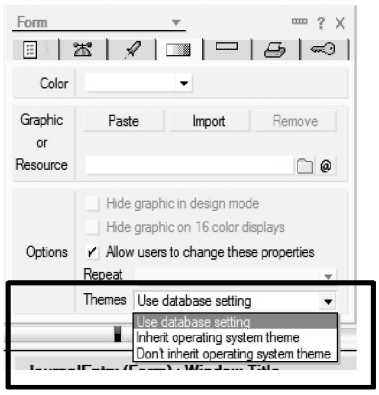
This means that non-xPage apps will need the css defined!

The Domino dojo is a modified version!

Domino Themes

- The test pages looks ugly!
- Domino 8.5 provides Themes for xPages
- These are set on the Database and form
- Use the standard Dojo CSS – tundra and soria





Start with the Header

- First you need to include the CSS files
- In the HTML Head Content

```
"<style type=\"text/css\">" + @NewLine +
"@import \"/domjs/dojo-1.1.1/dijit/themes/tundra/tundra.css\";"+
"@import \"/domjs/dojo-1.1.1/dijit/themes/soria/soria.css\";"+
"@import \"/domjs/dojo-1.1.1/dojo/resources/dojo.css\";" +
"</style>"
```

HTML Header

- This line will cause Dojo to load and parse the other controls

```
"<script type=\"text/javascript\"
   src=\"/domjs/dojo-1.1.1/dojo/dojo.js\"
      djConfig=\"parseOnLoad: true\">
</script>"
```

- "parseOnLoad:true" – Enables the page-load parsing of the widgets and in-markup code.
- You can add "isDebug:true" to enable debugging messages

```
djConfig="parseOnLoad:true, isDebug:true"></script>
```

addOnLoad Event

- The Dojo addOnLoad is used to insure all functions are loaded and ready.

```
//common method of loading code onLoad
var init = function(){
    console.log("This runs when page is ready." +
        "It will appear in the console");
};
dojo.addOnLoad(init);
```

- You can use the Form JS onLoad event if your careful - VERY careful
- It's NOT recommended

If you include the Domino rich text control, then the form's onLoad and JS header are ignored.

Rich Text Editor - Choices

- Using HTML
  - The default behavior renders the control as HTML.
- Using Java Applet
  - renders the control as an applet. This allows web users to see mouse-over distinctions for items in the outline, including change of color to indicate a selected item, and see background items.
- Using Best Fit for OS
  - Domino Web server renders the control as it sees fit depending on the browser.
- Using JavaScript Control
  - The control is built with the Dojo toolkit. The benefits are similar to using an applet, but uses standard HTML and CSS.

16

These are set on the field's property dialog.

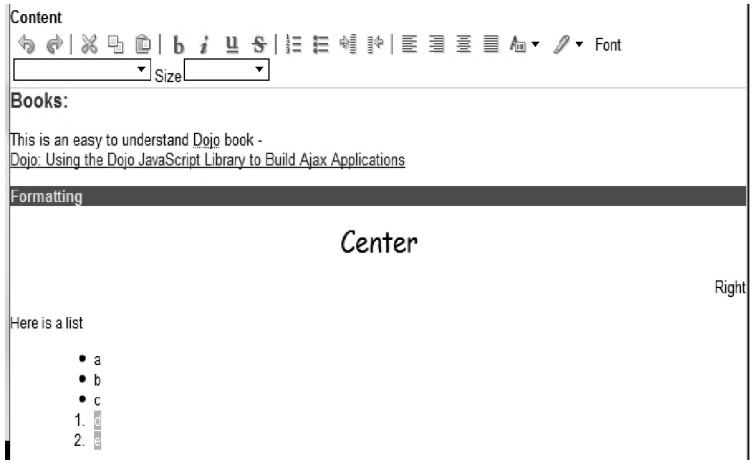
Working with Widgets

- Each control is downloaded when requested.

```
<script type="text/javascript">
      dojo.require("dojo.parser");
      dojo.require("dijit.form.TextBox");
      dojo.require("dijit.form.CheckBox");
      dojo.require("dijit.form.DateTextBox");
  </script>
```

As a JavaScript Control

- The rich text editor is now a Dojo widget



Rich Tech Editor Caveats

- The content is saved in MIME format
- It's converted to Notes CD if edited in Notes.
- Not everything converts correctly
  - Fonts
    - Web fonts are x-small, small, medium, large, etc.
    - Notes fonts are point sizes.
  - Table borders
  - Embedded images – visible but lost when saved
  - Tab tables – only the visible row is saved
  - “Hide when” - if hidden from web are lost during the save
- Security concern – malicious JavaScript
  - user can enter code that will be rendered when viewed

Some More Caveats

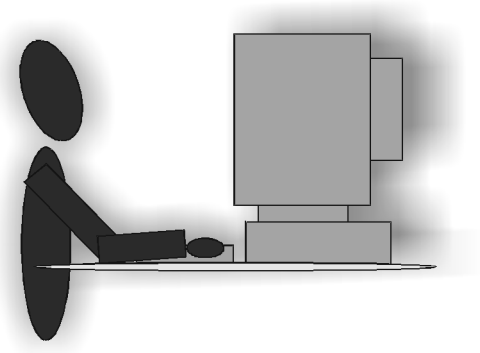
- The Domino Rich Text Control is set to prevent Dojo from parsing controls.
- Your choice is to load the control yourself
- Or change the behavior for the server
  - Edit this file on the server:

data\domino\template\dojo-1.1.1\dom_common.htm

```
<script type="text/javascript"
  src="&DOMINO_CTRL_URI_JSLIB;/dojo/dojo.js"
  djConfig="locale:
  '&DOMINO_CTRL_ACCEPT_LANGUAGE;', isDebug: true,
  parseOnLoad: true">
  </script>
```

  - Restart http

Demo Rich Text Control



Dojo Widgets

- dijit.Editor
- dijit.form.Textarea
- dijit.form.Button
- dijit.form.ValidationTextBox
- dijit.form.ComboBox
- dijit.Tooltip
- dojox.validate.regexp
- dijit.form.CheckBox
- dijit.form.TimeTextBox
- dijit.form.DateTextBox
- dijit.Menu
- dijit.form.Slider

Using the widgets

- You have to add the controls you need

```
"<script type=\"text/javascript\">" +
   "dojo.require(\"dojo.parser\");"  +
   "dojo.require(\"dijit.Editor\");" +
   "dojo.require(\"dijit.form.Textarea\");" +
   "dojo.require(\"dijit.form.Button\");" +
   "dojo.require(\"dijit.form.ValidationTextBox\");"+
   "dojo.require(\"dijit.form.ComboBox\");" +
   "dojo.require(\"dijit.Tooltip\");" +
   "dojo.require(\"dojox.validate.regexp\");" +
   "dojo.require(\"dijit.form.CheckBox\");" +
   "dojo.require(\"dijit.form.TimeTextBox\");" +
   "dojo.require(\"dijit.form.DateTextBox\");" +
   "dojo.require(\"dijit.form.TextBox\");" +
"</script>"
```

TextBox

- The base widget dijit.form.Textbox by itself can trim, change case, and require input.
- Add to the Field Properties other:

```
dojoType="dijit.form.TextBox" trim="true"
  propercase="true"
```

- Available Attributes
  - Lowercase - Converts all characters to lowercase
  - Maxlength - HTML INPUT tag maxlength declaration.
  - Propercase - Set the first character of each word to uppercase
  - Trim - Removes leading and trailing whitespace
  - uppercase

24

The dojo attributes can be set in the field properties, HTML tab – Other entry or the HTML Attributes in the Objects navigator

The only difference is the add delimiters needed for the HTML Attribute. It expects the value to be the result of a @formula. This will let you use a computed for display field that contains the dojo type and just reference it. Great if you have multiple fields with the same type.

ValidationTextBox

- Validating the input when the box loses focus
- InvalidMessage – the error message
  - PromptMessage – display when user enters field
  - Required – Field is required to have a value
  - regExp - Uses the standard Regular expression syntax
    - The start and ending qualifiers ^ and $ are implicit
- You can use the HTML attributes for the field
- Or the Field Properties HTML other:

```
dojoType="dijit.form.ValidationTextBox"
regExp="\d{5}"
required="true"
invalidMessage="Zip code needs 5 digits."
```

25

Fields HTML attributes expect formula results, so you have to add delimiters

The Field Properties – other allows you to just past the values in.
But it's harder to view

NumberTextBox

- Similar to the textBox, lets you control the values entered
- Attributes
  - currency: the 3 letter currency code - “USD"
  - fractional: (currency only) to include the fractional portion.
  - locale: override the locale on this widget only
  - pattern: override localized convention with this pattern.
  - places: number of decimal places to accept.
  - strict: strict parsing
  - symbol: (currency only) override currency symbol.
  - type: Format type: decimal, percent, currency

DateTextBox

- Returns the Selected date in text format
- Easy to convert to Notes date/time
  - Needs text format to display – not Notes date/time!
- Attibutes:
  - datePattern: override localized convention using this pattern regardless of locale.
  - formatLength: choose from full, long, medium or the default format of short
  - locale: override the locale on this widget only, choosing from djConfig.extraLocale
  - Strict (default is false):
    - True - parsing matches exactly by regular expression.
    - False - tolerant matching for abbreviations & white space.

ComboBox

- Works like a Select list but lets user enter a value or select a value.

```
dojoType="dijit.form.ComboBox" autocomplete="false"
value="Breakfast" onChange="setVal1"
```

- Can use dojo.data to pull selection from server
- Displayed value is passed back
  - The option tags do not have hidden submit values
- To use a hidden value use FilteringSelects.

FilteringSelect

- Filter the combo box as the user enters text.
- It will display an error if the user enters invalid selection.
- Good for long list that normally need scrolling.
- In the US it great for selection of states.

```
"dojo.require(\"dijit.form.FilteringSelect\");" \
```

- In other properties add this:

```
dojoType="dijit.form.FilteringSelect"
  autocomplete="true""
```

Checkbox and RadioButtons

- Both are included in the CheckBox.js so use
  - dojo.require("dijit.form.CheckBox")
- Work like the normal Domino equivalents
- Use the Dojo version more for consistency then for additional functionality

NumberSpiner

- Used to let the user enter integer values.
- The down and up arrow buttons "spin" the number up and down.
- Holding down a button makes the spinning accelerates.
- Attrributes
  - LargeDelta – amount adjusted using the PgUp/Dn keys
  - SmallDelta - amount adjusted using the arrow keys
  - DefaultTimeout - milliseconds before a held button becomes typematic
  - TimeoutChangeRate - fraction of time used to change the typematic timer between events

Form Widget

- Provide control over all dijits used in a form.
- Methods work only on dijits in the form
- Methods
  - GetValues - generate a JSON structure from all widgets on the form
  - IsValid - Return true if every widget's isValid method returns true.
  - SetValues - fill in form values from a JSON structure
  - Submit - programmatically submit form
  - Sample form:
    - archive.dojotoolkit.org/nightly/dojotoolkit/dijit/tests/form/Form.html

Can be used to process data from an agent. The agent returns data in JSON format. When the data is updated and submitted, it is passed to an agent as JSON data and the agent can update the documents.

Tooltip

- Display pop-up text to help or provide information

```
<span id="Help"> </span>
<div dojoType="dijit.Tooltip" connectId="Help"
label="Select one or more categories to group this
  recipe">
</div>
```

33

Easy way to add help functionality to a web page or site.

I have used a help view/ help docs to let users maintain what is displayed. A dblookup in a computedtext field pull the needed vergace for the tip to display.

Slider

Slider can be horizontal or Vertical

```
dojo.require("dijit.form.slider");
dojoType=dijit.form.VerticalSlider"

<div dojotype="dijit.form.HorizontalSlider" name="Rate1"
  value="<Computed Value>"  maximum="10" minimum="0"
  discretevalues="11" style="margin-left: 5px; height:
  50px;" id="Rate1" onchange="setRating(arguments[0]);" >
```

- Add labels

```
<ol dojoType="dijit.form.HorizontalRuleLabels"
  container="topDecoration" style="height:1em;
  font-size:75%; color:gray;">
  <li>0</li><li>2</li><li>4</li>
  <li>6</li><li>8</li><li>10</li>
</ol>
```

- Use the onchange event to update other fields
- If you want to save the value to Domino, you will have to code for it.

34

The slider need the onchange event to be able to update a field that Domino can use. This then can be used as an way for users to select data.

Tree

- Displays a graphical tree structure
- The TreeStoreModel has a single root
  - dijit.tree.TreeStoreModel

```
div dojoType="dijit.tree.TreeStoreModel" jsId="model"
childrenAttrs="kids" store="store" query="{id:'root'}">
  </div>
```

- The ForestStoreModel has multiple roots
  - dijit.tree.ForestStoreModel
- Data is a JSON structure
  - Can be from view, page, file or an agent

35

This can be used as a web site navigator by pulling the JSON from a view or profile document
dijit.tree.ForestStoreModel

The ForestStoreModel connects a data store with multiple "root" items to a dijit.Tree.

An example might be a geographical database; there are multiple continents but no "single" top level item called "world".

Debugging

- Companion.js for IE
- Firebug for Firefox
  - Let's you step through Javascript
  - See what is downloading
  - Watch HTTP requests
- Yslow
  - Give you a list of potential bottlenecks
  - Ignore the CDN or update it
    - In Firefox type about:config in the url bar
    - Do a right click, select New -> String
    - Type extensions.firebug.yslow.cdnHostnames as preference name
    - Set the value to your domain eg kalechi.com

If you are going to use AJAX or httpRequest then the HTTPheader add-on for Firefox is really nice to have. It lets you modify the request on the fly.

Performance

- Don't include everything on one page!
- Set the expiration on files and forms
  - Forms can be set in header
  - Files can be set using web rules
    - Create a separate rule for css, js, jpg, png, ect.
- Do a custom Build of Dojo
  - Then you load a smaller file at one time.
- Don't Parse, load programmatically Dojo controls

```
new dijit.form.Button({caption:"Name ",
   onclick:'logMessage("clicked simple")'});
```

Performance - continued

- To see the difference between parse and loading a control
- Dojo button test:

`turtle.dojotoolkit.org/~owen/bench/dojo/dijit/bench/test_Button-programmatic.html?count=100`

- On your server – Test page:

`yourserver/domjs/dojo-1.1.1/dijit/bench/benchTool.html`

- On your server - Results:

`yourserver/domjs/dojo-1.1.1/dijit/bench/test_button-results.html`

Resources

- The home of Dojo
  - dojotools.com
- Developerworks
  - www.ibm.com/developerworks/lotus/community/wikis.html
- Dojomino.com
  - Dojo tools for all Domino releases
- Sitepen
  - www.sitepen.com/
- Of course my site
  - www.kalechi.com

Questions?

Don't forget to the Evals!

Remember to share your knowledge!

As Caleb says
"Sharing is Good!"