

Building Amazing Interfaces With Drupal and JQuery UI

Nick Lewis

<http://nicklewis.org>

<http://entermedianow.com>

jQuery UI Facts

- **Part of Drupal 7 core**
- All plugins are tested for compatibility in IE 6.0+, FF 2+, Safari 3.1+, Opera 9.0+, and Google Chrome.
- Great Documentation!
- A modular architecture with a theme layer (sound familiar?)

Annoying Steps

1. [Upgrade to jQuery 1.32 via jquery_update 6.x-2.x-dev](#)
2. Unzip jquery_ui into sites/libraries
3. Rename the directory to "jquery-ui" instead of "jquery-ui-release-172-custom-download4dennis-etc..."
4. Make a helper function

Here's my helper function

```
function jui_add($files = array(), $type = 'ui') {  
  $sui_path = 'sites/all/libraries/jquery-ui/development-bundle/ui';  
  static $called;  
  if (!$called) {  
    $called = true;  
    drupal_add_js($sui_path.'/ui.core.js', 'module');  
  }  
  foreach($files as $file) {  
    drupal_add_js($sui_path.'/'.$type.'/'.$file.'.js', 'module');  
  }  
}
```

Putting this function to use.

```
<?php
function meh_page() {

    jui_add(array('sortable'));

    jui_add(array('highlight'),'effects');

    $output = 'MY PAGE';
    return $output;
}
?>
```

jQuery UI from a Themer's Perspective

Adding jQuery UI styles to a theme

Use your theme's .info file

name = My UI enabled theme

core = 6.x

engine = phptemplate

stylesheets[all][] = my-amazing-styles.css

stylesheets[screen,projection][] = ui/ui.all.css

UI Container Theme Classes

.ui-widget-header, .ui-widget-content

Zanger St. Hubbins



Vicerok Bifflebun



Zipplefon Canterkuk



Dr. Bam Shizzlebon



Zanger St. Hubbins



Vicerok Bifflebun



Zipplefon Canterkuk



Dr. Bam Shizzlebon



UI State Classes

Council On A New Wombat Century

Board of Wombats

UIStates

jQuery flick theme classes

`.ui-widget-header`

`.ui-widget-content`

`ui-state-default` (Default)

`ui-state-error` (Error)

`ui-state-highlight` (Highlighted)

`ui-state-active` (Active)

`ui-state-hover` (Hover)

`ui-state-disabled` (Disabled)

No, you don't have to handcode these styles in your theme.

The screenshot displays the ThemeRoller interface, which allows users to customize themes without handcoding. The interface is divided into several sections:

- ThemeRoller Header:** Includes the logo, navigation buttons for "Roll Your Own", "Gallery", and "Help", and a "Download theme" button.
- Settings Panel (Left):** Contains various styling options such as "Font Settings", "Corner Radius", "Header/Toolbar", "Background color & texture" (with a color picker and percentage input), "Border", "Text", "Icon", "Content", "Clickable: default state", "Clickable: hover state", "Clickable: active state", "Highlight", "Error", "Modal Screen for Overlays", and "Drop Shadows".
- Accordion:** Shows a sample accordion with three sections. Section 1 is expanded, displaying placeholder text.
- Slider:** Displays a horizontal slider control.
- Datepicker:** Shows a calendar for November 2009, with the 13th highlighted.
- Progressbar:** Shows a horizontal progress bar.
- Highlight / Error:** Shows two sample buttons: one with an information icon and the text "Hey! Sample ui-state-highlight style." and another with a warning icon and the text "Alert: Sample ui-state-error style."
- Tabs:** Shows a set of three tabs labeled "First", "Second", and "Third".
- Dialog:** Shows a button labeled "Open Dialog".
- Overlay and Shadow Classes:** Shows a sample dialog box with placeholder text.

A banner at the top right of the interface reads: "★ New! Bring ThemeRoller into any page: [Get the ThemeRoller Firefox Bookmarklet!](#)"

UI Plugins

(where the action begins)

The Accordion

Council On A New Wombat Century

Board of Wombats

Wombats Accordion

UIStates

Wombats Accordion

▼ Dr. Bam Shizzlebon



▶ Zipplefon Canterkuk

▶ Vicerok Bifflebung

▶ Zanger St. Hubbins

Accordion (The Making Of)

Part 1: Read the docs

The markup of your accordion container needs pairs of headers and content panels:

```
<div id="accordion">  
  <h3><a href="#">First header</a></h3>  
  <div>First content</div>  
  <h3><a href="#">Second header</a></h3>  
  <div>Second content</div>  
</div>
```

Accordion (The Making Of)

Part 2: covert to the drupal

```
function wombat_accordion() {
  $wombats = wombat_load_wombats();
  jui_add(array('accordion'));
  drupal_add_js(WOMBAT_PATH.'/wombat.behaviors.js', 'module');
  $output = '<div id="wombat-accordion">';
  foreach($wombats as $node) {
    $output .= '<h3><a href="#">'.$node->title.'</a></h3>';
    $output .= '<div>'.node_view($node, FALSE, TRUE).'
```

Accordion (The Making Of)

Part 3: set yer drupal behavior in the js file

```
Drupal.behaviors.wombatAccordion = function(context) {  
  $("#wombat-accordion:not('.wombat-accordion-processed')")  
    .addClass('wombat-accordion-processed')  
    .each(function() {  
      $(this).accordion();  
    });  
}
```


Yes, we are having a good time.



Tabs: The good times never end

Board of Wombats Wombats Accordion **Wombats Tabs** UIStates

Wombats Tabs

Dr. Bam Shizzlebon Zipplefon Canterkuk Vicerok Bifflebung Zanger St. Hubbins



[Home](#)

Tabs (The Making Of)

Part 1: Read the all powerful docs

Note: This is a sample of markup generated by the tabs plugin, not markup you should use to create a tabs. The only markup needed for that is

```
<div id="tabs">
  <ul>
    <li><a href="#tabs-1">Nunc tincidunt</a></li>
    <li><a href="#tabs-2">Proin dolor</a></li>
    <li><a href="#tabs-3">Aenean lacinia</a></li>
  </ul>
  <div id="tabs-1">
    <p>Tab 1 content</p>
  </div>
  <div id="tabs-2">
    <p>Tab 2 content</p>
  </div>
  <div id="tabs-3">
    <p>Tab 3 content</p>
  </div>
</div>.
```

Tabs (The Making Of)

Part 2: "Make it Drupal"

```
function wombat_tabs() {
  $wombats = wombat_load_wombats();
  jui_add(array('tabs'));
  drupal_add_js(WOMBAT_PATH.'/wombat.behaviors.js', 'module');
  $output = '<div id="wombat-tabs">';
  foreach($wombats as $key => $node) {
    $tabs[] .= '<li><a href="#tab-' . $key . '">' . $node->title . '</a></li>';
    $tabs_content .= '<div id="tab-' . $key . '">' . node_view($node, FALSE, TRUE) . '</div>';
  }
  $output .= '<ul>';
  $output .= implode($tabs);
  $output .= '</ul>';
  $output .= $tabs_content;
  $output .= '</div>';
  return $output;
}
```

Tabs (The Making Of)

Part 3: Drupal Behaviorization

```
Drupal.behaviors.wombatTabs = function(context) {  
  $("#wombat-tabs:not('.wombat-tabs-processed')")  
    .addClass('wombat-accordion-tabs')  
    .each(function() {  
      $(this).tabs();  
    });  
}
```

Dialogs & Effects

Dialogs: One Giant Behavior

```
Drupal.behaviors.wombatDialog = function(context) {
  $("#wombat-dialog .node:not('.wombat-dialog-processed')")
    .addClass('wombat-accordion-processed')
    .each(function() {
      var node = $(this);
      node.click(function(){
        $("#wombat-dialog").append('<div id="dialog"></div>');
        var wombatPhoto = $(this)
          .find('img');

        var name = $(this)
          .find('h3')
          .text();

        wombatPhoto
          .wombatShowForce();

        var wombatDialog = $('#dialog')
          .empty()
          .prepend(name + " is making threats and wild gestures")
          .dialog({
            'title': 'Danger!',
            'buttons': {
              'Smack him!': function() {
                wombatPhoto
                  .wombatSmack()
                wombatDialog
                  .dialog('close');
              },
              'Run away!': function() {
                wombatPhoto
                  .wombatGrow()
                wombatDialog
                  .dialog('close');
              }
            }
          });
    });
});
```

Dialogs: Breaking it Up

Standard Behavior ->

Click event on node ->

Shake effect (more later) ->

Actual Dialog definition->

Button Callbacks ->

```
Drupal.behaviors.wombatDialog = function(context) {
  $("#wombat-dialog .node:not('.wombat-dialog-processed')")
  .addClass('wombat-accordion-processed')
  .each( function() {
    var node = $(this);
    node.click( function(){
      $("#wombat-dialog").append('<div id="dialog"></div>');
      var wombatPhoto = $(this)
        .find('img');

      var name = $(this)
        .find('h3')
        .text();

      wombatPhoto
        .wombatShowForce();

      var wombatDialog = $('#dialog')
        .empty()
        .prepend(name + " is making threats and wild gesture")
        .dialog({
          'title': 'Danger!',
          'buttons': {
            'Smack him!': function() {
              wombatPhoto
                .wombatSmack()
              wombatDialog
                .dialog('close');
            },
            'Run away!': function() {
              wombatPhoto
                .wombatGrow()
              wombatDialog
                .dialog('close');
            }
          }
        });
    });
  });
};
```


jQuery UI FX

```
$.fn.wombatShowForce = function() {  
  return this  
    .each( function() {  
      $(this)  
        .effect('shake', {}, 100);  
    });  
}  
  
$.fn.wombatSmack = function() {  
  return this  
    .each( function() {  
      $(this)  
        .effect('explode', {}, 1000);  
    });  
}  
  
$.fn.wombatGrow = function() {  
  return this  
    .each( function() {  
      $(this)  
        .effect('scale', {percent:500}, 1000);  
    });  
}
```

Sortable Lists

Add a new item to this day:

Instructional strategy

Student Work

In the Classroom

Outside the Classroom

To do list

Student Work: Class work

[rwar](#)

gdzg



To do list

[test 1 2](#)

test



Instructional strategy

[Social Studies review](#)

sfgwsdfd



Sortable Lists: the markup

```
▼ <ul id="day_2" class="sortable sortable-processed ui-sortable" unitplan_nid="5212" day="2">  
  ▶ <li class="qui-active-form unitplan-form-wrapper ui-helper-hidden unitplan-item clearfix" style="">  
  ▶ <li id="item_5216" class="unitplan-item-state-default unitplan-item clearfix" style="">  
  ▶ <li id="item_5218" class="unitplan-item-state-default unitplan-item clearfix">  
  ▶ <li id="item_5213" class="unitplan-item-state-default unitplan-item clearfix" style="">  
</ul>
```

Sortable Lists: the jquery

```
$.fn.sortableContainerList = function() {  
  return this.each( function() {  
    var list = $(this);  
    var settings = {  
      forcePlaceholderSize: true,  
      connectWith: 'ul.sortable',  
      placeholder: 'ui-state-highlight',  
      change: function(event, ui) {  
        var $previous = $(this).prev();  
        if (0 === $previous.size()) {  
          var $next = $(this).next();  
          $next.insertBefore(this);  
        }  
      },  
      update: function(event, ui) {  
        var result = list.sortable('serialize');  
        var day = list.attr('day');  
        var container_nid = list.attr('unitplan_nid');  
        $.get(Drupal.settings.console.sortableSave+'/'+container_nid+'/'+day, result);  
      }  
    };  
    $(this)  
      .sortable(settings);  
  });  
};
```

Sortable Lists: The request

GET http://localhost/quilltree/unitplan-ajax/5212/sort...le-save/5212/2?item[]=5213&item[]=5216&item[]=5218 200 OK 381ms

Params Headers Response

```
item[] 5213  
item[] 5216  
item[] 5218
```

Sortable Lists: the callback

```
function unitplan_ajax_sortable_save($unitplan_nid) {
  if (arg(3) && arg(4)) {
    unitplan_cache_reset($unitplan_nid);
    $day = arg(4);
    $container_nid = arg(3);
    if ($_GET['item']) {
      foreach($_GET['item'] as $weight => $item_nid) {
        db_query("UPDATE {unitplan_item} SET day = %d,weight= %d, unitplan_nid = %d WHERE item_nid = %d", $day,$weight,$container_nid, $item_nid);
      }
    }
  }
}
```

Questions?