






















Drupal 7 and JavaScript

What / Where / How

JavaScript bundled with core

jQuery 1.4.4
jQuery UI 1.8.7

 ajax.js	23 KB
 authorize.js	1 KB
 autocomplete.js	8 KB
 batch.js	939 bytes
 collapse.js	3 KB
 drupal.js	13 KB
 form.js	2 KB
 jquery.ba-bbq.js	4 KB
 jquery.cookie.js	961 bytes
 jquery.form.js	10 KB
 jquery.js	79 KB
 jquery.once.js	3 KB
 machine-name.js	5 KB
 progress.js	3 KB
 states.js	12 KB
 tabledrag.js	42 KB
 tableheader.js	4 KB
 tableselect.js	4 KB
 textarea.js	920 bytes
 timezone.js	3 KB
 vertical-tabs.js	7 KB

Including files

`module.info, drupal_add_js(),
drupal_add_library()`

Including files

.info file

- every page gets these files

`drupal_add_js()`

- good to use in functions that build pages

`drupal_add_library()`

- used with `hook_library()`
- good for bundling multi-file libraries together

Including files

module.info

```
name = Drupal 7 JavaScript Walkthrough
```

```
description = "Gives an example different ways to use javascript in drupal."
```

```
core = 7.x
```

```
; our test file
```

```
files[] = d7js.test
```

```
; some styles for our demo stuff
```

```
stylesheets[all][] = css/d7jsmod.css
```

```
; some scripts for the pages
```

```
scripts[] = js/everypage.js
```

```
; See http://drupal.org/node/542202 for more info on .info files
```

Including files

`drupal_add_js()`

- type file, inline, external, setting
- scope: header, footer
- group: JS_LIBRARY, JS_DEFAULT, JS_THEME
- weight: lighter items first, heavy items last

http://api.drupal.org/api/drupal/includes--common.inc/function/drupal_add_js

Including files

```
// include files from our module-  
drupal_add_js(drupal_get_path('module', 'd7jsmod') . '/js/page-2.js');  
  
drupal_add_js(array('somevalue' => '5555555555'), 'setting');  
  
// do some inline scripting-  
drupal_add_js('jQuery(document).ready(function () { console.log("page 2 -  
drupal_add_js('jQuery(document).ready(function () { console.log("page 2 -  
    array('type' => 'inline', 'scope' => 'footer', 'weight' => 5)  
);
```

Including files

`drupal_add_library()`

- add multiple files (js and css) in one call

`hook_library()`

- define your own libraries

`hook_library_alter()`

- hook into other libraries and alter them

Including files

drupal_add_library and hook_library

```
drupal_add_library('d7jsmod', 'library-2');  
  
drupal_add_library('system', 'jquery.cookie');
```

- hook_library: define dependencies, reference version / site, bundle css

```
$libraries['library-1'] = array(  
  'title'    => 'Library One',  
  'website' => 'http://example.com/library-1',  
  'version'  => '1.2',  
  'js'       => array( drupal_get_path('module', 'd7jsmod') . '/js/library-1.js' => array() ),  
  'css'      => array( drupal_get_path('module', 'd7jsmod') . '/js/library-1.css' => array('type' => 'file', 'media' => 'screen') ),  
);
```

Drupal and AJAX

Easier than it used to be - however
still lots of PHP

Drupal and AJAX

Forms in Drupal (at a glance)

- Custom Forms = Arrays
- Array items = Elements
- Elements have properties

Drupal reads arrays and does stuff

Drupal and AJAX

Element array's have '#ajax'

- callback: the function to call
- path: to override the core system/ajax path
- wrapper: ID element on the page to target
- effect
- speed
- event: defaults to auto-sense the type of element
- prevent: other events to prevent on this element while waiting
- method: replaceWith, before, after, append...
- progress

Drupal and AJAX

```
$form['howmany_select'] = array(-  
  '#title'      => t('How many checkboxes do you want?'),-  
  '#type'       => 'select',-  
  '#options'    => array(1 => 1, 2 => 2, 3 => 3, 4 => 4),-  
  '#default_value' => $default,-  
  '#ajax'       => array(-  
    'callback' => 'demo_form_autocheckboxes_callback',-  
    'wrapper' => 'checkboxes-div',-  
  ),-  
);
```

Drupal behaviors

centralized bundles of fun

Drupal behaviors

- element-centric
- attach / detach
- page load and ajax loads

```
// You could add additional behaviors here.
Drupal.behaviors.d7jsmodAttachDetach = {
  attach: function (context, settings) { console.log('attach - Drupal.behaviors.d7jsmodAttachDetach') },
  detach: function (context, settings) { console.log('detach - Drupal.behaviors.d7jsmodAttachDetach') }
};
```

Drupal behaviors

context - limit the scope of the selectors

settings - just the Drupal.settings

```
// Store our function as a property of Drupal.behaviors.-  
Drupal.behaviors.d7jsmodCheckboxes = {  
  attach: function (context, settings) {  
  
    console.log('attach - Drupal.behaviors.d7jsmodCheckboxes');  
  
    // Find all the secure links inside context that do not have our processed class.-  
    $('input[type=checkbox]', context)-  
      // Only process elements once.-  
      .once('d7js-processed')-  
      // Then stick some text into the link denoting it as secure.-  
      .click( indicateChecked )-  
  
      .each( indicateChecked )-  
    }-  
  }-  
}
```


Drupal behaviors

```
Drupal.attachBehaviors = function (context, settings) {  
  context = context || document;  
  settings = settings || Drupal.settings;  
  // Execute all of them.  
  $.each(Drupal.behaviors, function () {  
    if ($.isFunction(this.attach)) {  
      this.attach(context, settings);  
    }  
  });  
};
```

```
//Attach all behaviors.  
$(function () {  
  Drupal.attachBehaviors(document, Drupal.settings);  
});
```

Drupal Settings

useful

Drupal Settings

Pass data from PHP to JavaScript

```
// PHP-  
drupal_add_js(array('somevalue' => '5555555555'), 'setting');  
-  
// JavaScript-  
console.log('Drupal.settings.somevalue', Drupal.settings.somevalue);
```

Drupal Settings

Drupal: globally name-spaced

Core Settings:

- `Drupal.settings.basePath`
- `Drupal.settings.ajax`

Drupal Aggregation

Built in optimization

Drupal Aggregation

Combines files

- site-wide files (.info) get aggregated
- each page gets it's own aggregation file
 - drupal_add_js / drupal_add_library
- no compression, just combines the files