

# Using Ajax in Your Plugins and Themes



**GET THE SLIDES**

**[HTTP://WWW.ITSANANDERSON.COM/WCSEA.PDF](http://www.itsananderson.com/wcsea.pdf)**

# About Me



**ROSE-HULMAN**  
Institute of Technology

The nation's top undergraduate engineering, science, and mathematics college.

Institutional Research, Planning and Assessment

Home **Visitors** **Students** **Faculty** **Staff**

### IRPA Mission

Our mission is to plan, design, and implement strategies for the collection and analysis of data and report information to support the evolving needs of faculty and administrators in the education of Rose-Hulman students and to be recognized as a national leader in institutional research and the assessment of student learning.

### Current Services

The IRPA Office provides services to faculty, staff, students and other external organizations to support our institution's vision and mission. Currently, we offer the following information services to our constituents:

<b>Institutional Research</b> Planning Assessment CASO 360° Spotlight About IRPA	<b>Portfolio Assessment</b> Surveys/Tests Assessment Plans Scanning Tests/Exams Accreditation Support/Strategic Planning	<b>Educational Research</b> Institutional Data Planning Conferences/Special Events Institutional Student Learning Outcomes Evaluation
---	--	---

### What's New

- Visit Rose Evaluation Conference 2011 to learn more about conference registration and other news about this assessment conference to be held in April 2011.
- The Spring 2011 IRPA newsletter, *The Compass*, is now available.
- 2010-2011 Academic Alumni Survey (Preliminary Report Completion Begins)

### Popular Search Items

- Student Information Questionnaire Did you know (and you want to know) something about our class of 2014? Find out the facts collected through our IQG.
- 2010-11 Commencement Data Sheet (CCDS) General information in PDF format about Rose-Hulman on a variety of topics can be found in our CCDS.
- State brochure on accreditation available from the Council for Higher Education Accreditation (CHEA)—information about accreditation and its value.

### Contact Us

Julia Williams  
Executive Director of IRPA  
(812) 877-9150

Timothy Chivo  
Director of Institutional Research  
(812) 877-8910

Shannon Sexton  
Director of Assessment  
(812) 877-8020

Scott Fisher  
Director of Data Management & Reporting  
(812) 877-8830

Juliana Bradford  
Administrative Assistant  
(812) 877-8916

Margie Neuman  
Office Assistant/Conference Coordinator  
(812) 877-8101

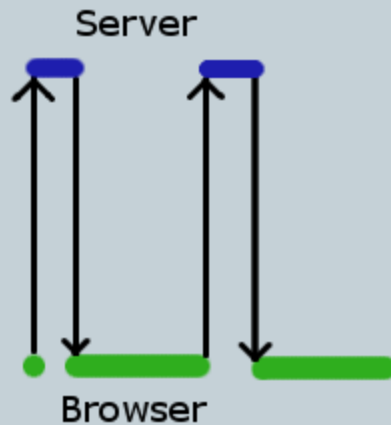
Office Phone  
(812) 877-8916

# Ajax – “What Does it Mean?”

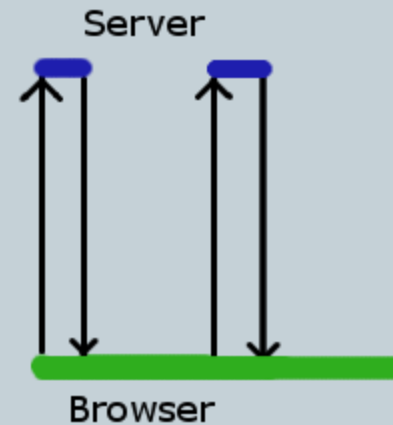


“Asynchronous JavaScript and XML”

Synchronous Requests



Asynchronous Requests



# Simplify With jQuery



- Send POST or GET requests
- Send data as an associative array (POST requests)
- Handle response with anonymous function
- Receive response as json, text, html (and more)

```
jQuery.get(  
    'ajax.php', {  
        name : 'Charlie Brown'  
    }, function(response) {  
        alert( response );  
    }, 'text' );
```

```
jQuery.post(  
    'ajax.php', {  
        name : 'Charlie Brown'  
    }, function(response) {  
        alert( response );  
    }, 'text' );
```

# jQuery Basics – No Conflict



- “\$” is not defined by default
- Use “jQuery” instead, or...

```
(function($) {  
    // Use $ to access jQuery here  
})(jQuery)
```

```
// To wait until page has loaded (useful!)  
jQuery(document).ready(function($) {  
    // Use $ to access jQuery here  
})
```

# jQuery Basics – Ajax Response Types



- json – Great for responses that need complex evaluation
- html – For html responses that will be inserted into the page
- text – For basic plain text responses
- others – jsonp, xml, script, multiple... less common

# Ajax in WordPress – Queue Your Script



- Use `wp_enqueue_script()` to add your script to a page
- `handle` – Name of script, e.g. “my-awesome-script”
- `src` – Where your script can be found
  - Use `get_template_directory_uri()` in parent themes
  - Use `get_stylesheet_directory_uri()` in child themes
  - Use `plugins_url( 'script.js', __FILE__ )` in plugins
- `deps` – An array of scripts that this script relies on, by handle. If you need jQuery, use something like “array( 'jquery' )”. Go to [wp\\_enqueue\\_script Codex](#) page for a list of available dependencies.
- `ver` – The version of the script. Useful for “cachebusting” when your plugin is updated
- `in_footer` – Whether to place the script in the footer (rather than the header). Defaults to false.

# Ajax in WordPress – Queue Your Script (cont'd)



## Examples

```
// Inside a theme
wp_enqueue_script( 'my-awesome-script', get_template_directory_uri() . '/js/my-awesome-script.js',
    array('jquery'), 1.0 );

// Inside a child theme
wp_enqueue_script( 'my-awesome-script', get_stylesheet_directory_uri() . '/js/my-awesome-script.js',
    array('jquery'), 1.0 );

// Inside a plugin
wp_enqueue_script( 'my-awesome-script', plugins_url( '/js/my-awesome-script.js', __FILE__ ),
    array('jquery'), 1.0 );
```

# Ajax in WordPress – Queue Your Script (cont'd)



- Queue scripts intelligently
- Use hooks and conditional tags

```
add_action( 'admin_init', 'queue_my_script' );
function queue_my_script() {
    // Queue your admin script here
}

add_action( 'load-edit.php', 'queue_my_script' );
add_action( 'load-post-new.php', 'queue_my_script' );
function queue_my_script() {
    if ( isset( $_GET['post_type'] ) ) {
        $post_type = $_GET['post_type'];
    } elseif ( isset( $_GET['post'] ) ) {
        $post_type = get_post_type( $_GET['post'] );
    }
    if ( ( isset( $_GET['post_type'] ) && 'my_post_type' == $post_type ) ) {
        // Queue your post type script here
    }
}
```

```
$menu_page = add_menu_page( /* plugin page params */ );
add_action( "load-$menu_page", 'queue_my_script' );
function queue_my_script() {
    // Queue your menu page script here
}

add_action( 'template_redirect', 'queue_my_script' );
function queue_my_script() {
    if ( is_single() ) {
        // Queue script only on single pages
    }

    if ( is_page( 'Contact' ) ) {
        // Queue script only on the "Contact" page
    }
}
```

# Ajax in WordPress – Making Requests



- Admin Ajax
  - ✦ Use 'ajaxurl' variable
  
- Frontend Ajax
  - ✦ Use 'wp\_localize\_script' function
  - ✦ Use 'init' hook

# Ajax in WordPress – Making Requests



## Admin Ajax

```
<?php  
  
add_action( 'wp_ajax_my_action', 'my_ajax_handler' );  
function my_ajax_handler() {  
    $name = $_POST['name'];  
    echo "hello $name";  
    die(); // Make sure you do this!  
}
```

```
jQuery.post(ajaxurl,  
    {  
        action : 'my_action',  
        name : 'Charlie Brown'  
    }, function(response) {  
        alert(response);  
    });
```

Make sure you check user permissions. Any logged in user can make ajax requests.

# Ajax in WordPress – Making Requests



## Frontend Ajax – wp\_localize\_script

```
wp_localize_script( 'my-awesome-script', 'my_awesome_script',  
    array( 'ajaxurl' => admin_url( 'admin-ajax.php' ) ) );
```

```
jQuery.post(my_awesome_script.ajaxurl,  
    {  
        action : 'my_action',  
        name : 'Charlie Brown'  
    }, function(response) {  
        alert(response);  
    });
```

```
add_action( 'wp_ajax_my_action', 'my_ajax_handler' );  
// Make sure you include the _nopriv_ action for unauthenticated users  
add_action( 'wp_ajax_nopriv_my_action', 'my_ajax_handler' );  
function my_ajax_handler() {  
    $name = $_POST['name'];  
    echo "hello $name";  
    die(); // Make sure you do this!  
}
```

# Ajax in WordPress – Making Requests



## Frontend Ajax – init

```
add_action( 'init', 'my_ajax_handler' );
function my_ajax_handler() {
    if ( !isset( $_POST['action'] ) ) {
        return;
    }
    $action = $_POST['action'];
    if ( 'my_action' == $action ) {
        $name = $_POST['name'];
        echo "hello $name";
        die();
    }
    // handle any other actions here
}
```

```
jQuery.post(location.href,
    {
        action : 'my_action',
        name : 'Charlie Brown'
    }, function(response) {
        alert(response);
    });
```

# Ajax in WordPress – Another Response Type



## Another Response Type – json

```
<?php

add_action( 'init', 'my_ajax_handler' );
function my_ajax_handler() {
    if ( !isset( $_POST['action'] ) ) {
        return;
    }
    $action = $_POST['action'];
    if ( 'my_action' == $action ) {
        $date = date('F j, Y');
        $name = $_POST['name'];
        $message = "hello $name";
        $response = array(
            'date' => $date,
            'message' => $message
        );
        die( json_encode( $response ) );
    }
    // handle any other actions here
}
```

```
jQuery.post(location.href,
{
    action : 'my_action',
    name : 'Charlie Brown'
}, function(response) {
    alert( 'The date is: ' + response.date );
    alert( 'The message is: ' + response.message );
}, 'json' );
```

# Simplify Your Ajax Handlers



```
add_action( 'wp_ajax_my_action1', 'my_action1' );
function my_action1() {
    // do stuff
}

add_action( 'wp_ajax_my_action2', 'my_action2' );
function my_action2() {
    // do stuff
}

add_action( 'wp_ajax_my_action3', 'my_action3' );
add_action( 'wp_ajax_nopriv_myaction3', 'my_action3' );
function my_action3() {
    // do stuff
}

add_action( 'wp_ajax_my_action4', 'my_action4' );
add_action( 'wp_ajax_nopriv_myaction4', 'my_action4' );
function my_action4() {
    // do stuff
}
```

# Simplify Your Ajax Handlers



```
<?php

class WP_Ajax {
    function __construct( $ajax_prefix = 'a', $nopriv_prefix = 'n' ) {
        $regex = "/^($ajax_prefix)?($nopriv_prefix)?_|^($nopriv_prefix)?($ajax_prefix)?/";
        $methods = get_class_methods( $this );
        foreach ( $methods as $method ) {
            if ( preg_match( $regex, $method, $matches ) ) {
                if ( count( $matches ) > 1 ) {
                    $action = preg_replace( $regex, '', $method );
                    if ( count( $matches ) == 3 ) {
                        add_action( "wp_ajax_$action", array( $this, $method ) );
                        add_action( "wp_ajax_nopriv_$action", array( $this, $method ) );
                    } else {
                        if ( $matches[1] == $ajax_prefix ) {
                            add_action( "wp_ajax_$action", array( $this, $method ) );
                        } else {
                            add_action( "wp_ajax_nopriv_$action", array( $this, $method ) );
                        }
                    }
                }
            }
        }
    }
}
```

# Simplify Your Ajax Handlers



```
<?php

class My_WP_Ajax extends WP_Ajax {
    function __construct() {
        parent::__construct();
    }

    function a_my_action1() {
        // do stuff
    }

    function a_my_action2() {
        // do stuff
    }

    function an_my_action3() {
        // do stuff
    }

    function an_my_action4() {
        // do stuff
    }
}

new My_WP_Ajax();
```

# Example Plugin



## Simple Post Ratings

# Questions?



Will Anderson

<http://www.itsananderson.com/>

@itsananderson

will@itsananderson.com