Image Manipulation in WordPress 3.5

WordCamp Phoenix 2013

Mike Schroder (DH-Shredder)
@GetSource - http://www.getsource.net
Who Am I?

- Mike Schroder, a.k.a DH-Shredder, a.k.a. @GetSource
- Third Culture Kid, enjoy Coffee & Sailing
- WordPress 3.5 Recent Rockstar and wp-cli Contributor
- Co-Author of WP_Image_Editor
- Happy DreamHost Employee
What was wrong?
GD was used directly.
This meant a mess of functions for Developers.
What changed?
GD was abstracted.
WP_Image_Editor is born.
What’s WP_Image_Editor?

Centralized way to read an image file directly, manipulate it, and output
Community Built.

Primarily Marko Heijnen and I, with awesome help from Japh Thomson, Kurt Payne, Andrew Nacin and Cristi Burcă
3.5 comes with Editors for GD and Imagick support
Imagick 2.2.0+ compiled with Imagemagick 6.2.9+ for full support
With **Imagick**, this means support for **Color Profiles**.
What’s the catch?

Deprecation filters from GD abstraction
Deprecated Filters

- `image_save_pre` is now `image_editor_save_pre`
- `wp_save_image_file` is now `wp_save_image_editor_file`
- `image_edit_before_change` is now `wp_image_editor_before_change`
What’s it missing?

Centralized way to read an image attachment from the database and manage its sizes and properties.
That’s WP_Image.
...Hopefully
Okay.

So, what *can* I do with it?
Resize, Batch Resize, Crop, Flip, Rotate, and Stream.
SHOW ME THE CODE
/**
 * Tests whether there is an editor that supports a given mime type or methods.
 * @since 3.5.0
 * @access public
 * @param string|array $args Array of requirements.
 * Accepts { 'mime_type'=>string, 'methods'=>[string, string, ...] } } 
 * @return boolean true if an eligible editor is found; false otherwise
 */
function wp_image_editor_supports( $args = array() ) {
    ...
}
/**
 * Returns a WP_Image_Editor instance and loads file into it.
 * @since 3.5.0
 * @access public
 * @param string $path Path to file to load
 * @param array $args Additional data.
 * Accepts {'mime_type'=>string, 'methods'=>{string, string, ...} }
 * @return WP_Image_Editor|WP_Error
 */

function wp_get_image_editor( $path, $args = array() ) {
    ...
}
/**
 * Resizes current image.
 *
 * @since 3.5.0
 * @access public
 * @abstract
 * @param int $max_w
 * @param int $max_h
 * @param boolean $crop
 * @return boolean|WP_Error
 */

abstract public function resize( $max_w, $max_h, $crop = false );
/**
 * Processes current image and saves to disk
 * multiple sizes from single source.
 *
 * @since 3.5.0
 * @access public
 * @abstract
 *
 * @param array $sizes { {'width'=>int, 'height'=>int, 'crop'=>bool}, ... }
 * @return array
 */
abstract public function multi_resize( $sizes );

Batch Resize.
abstract public function crop( $src_x, $src_y, $src_w, $src_h, $dst_w = null, $dst_h = null, $src_abs = false );
abstract public function rotate( $angle );
/**
 * Flips current image.
 * @since 3.5.0
 * @access public
 * @abstract
 *
 * @param boolean $horz Horizontal Flip
 * @param boolean $vert Vertical Flip
 *
 * @return boolean|WP_Error
 */

abstract public function flip( $horz, $vert );
/**
 * Streams current image to browser.
 *
 * @since 3.5.0
 * @access public
 * @abstract
 *
 * @param string $mime_type
 * @return boolean|WP_Error
 */

abstract public function stream( $mime_type = null );
Save **or** Convert.

```php
abstract public function save( $destfilename = null, $mime_type = null );
```
Time for an **Example**!

```php
// Get instance of WP_Image_Editor selected by WordPress
$image = wp_get_image_editor( '/path/to/cool_image.jpg' );

// Returns WP_Error on failure, so check.
if ( ! is_wp_error( $image ) ) {

    // Rotate in 90 degree increments, for now.
    $image->rotate( 90 );

    // Thumbnail, and crop.
    $image->resize( 300, 300, true );

    // Uses extension for type, unless optional mime parameter is used.
    $image->save( 'new_image.gif' );

    // Types only limited by Editor and what WordPress allows for uploads.
    if ( $image->supports_mime_type( 'application/pdf' ) )
        $image->stream( 'application/pdf' );
}
```
To load an attachment...

No current alternative to load_image_to_edit(), so:

wp_get_image_editor( _load_image_to_edit_path( $post_id ) );
Extend to create your own image-manip engines or functions.
Let’s extend **Imagick** for everyone’s favorite filter:
Sepia!
Sepia makes the world go brown.
Extend from `WP_Image_Editor`.

class GS_Imagick_Sepia_Editor extends WP_Image_Editor_Imagick {

    /**
     * Filters current in-memory image with Sepia
     * @since 1.0
     * @access public
     *
     * @param int $amount
     * @return bool|WP_Error
     */
    public function sephia( $amount = 80 ) {
        try {
            $this->image->sephiaToneImage( $amount );
            return true;
        }
        catch ( Exception $e ) {
            return new WP_Error( 'image_sephia_error', $e->getMessage() );
        }
    }
}
**Add Sepia Editor to beginning of editor search array.**

* The new editor doesn't need to be at beginning if specifically requesting an editor with `sepia()` method, but it's safer overall.

```php
function gs_add_imagick_sepia( $editors ) {
    if( ! class_exists( 'GS_Imagick_Sepia_Editor' ) )
        include( plugin_dir_path( __FILE__ ) . 'editors/imagick-sepia.php' );

    array_unshift( $editors, 'GS_Imagick_Sepia_Editor' );
    return $editors;
}
add_filter( 'wp_image_editors', 'gs_add_imagick_sepia' );
```
// Request an Editor with the `sepia()` method.
$sepia_editor = wp_get_image_editor( "/path/to/cool-image.jpg",
array( 'methods' => array( 'sepia' ) ) );

// Double-check that we have an editor, and the file is open.
if ( ! is_wp_error( $sepia_editor ) ) {

    // Filter with sepia using our new method.
    $sepia_editor->sepia();

    // Send the image to the browser without saving.
    $sepia_editor->stream();
}

Require and run the new method.
And, that’s it!
Image Editor Examples

• **GD**: `wp-includes/class-wp-image-editor-gd.php`
• **Imagick**: `wp-includes/class-wp-image-editor-imagick.php`
• **Gmagick**: [http://wordpress.org/extend/plugins/gmagick/](http://wordpress.org/extend/plugins/gmagick/)
I DON'T ALWAYS RESIZE

BUT WHEN I DO, I USE WP_IMAGE_EDITOR
More Resources!

- http://make.wordpress.org/core/2012/12/06/wp_image_editor-is-incoming/
- https://github.com/getsourc/imagick-sepia/
- https://github.com/humanmade/WPThumb/

Mike Schroder (DH-Shredder)
@GetSource - http://www.getsource.net