Create a Custom MetaSlider Theme

Slideshow: Theme Integrations · https://www.metaslider.com/docs/custom-metaslider-theme/

This guide will show you how to create custom MetaSlider slideshow designs using code.

This will help you if you're a theme developer and want to provide a custom MetaSlider theme. This will also help designers who want to customize a slideshow design for their clients.

We do also offer a no-code approach to creating custom theme via the Theme Editor. Please note that themes created through the method in this tutorial will not be available in the Theme Editor.

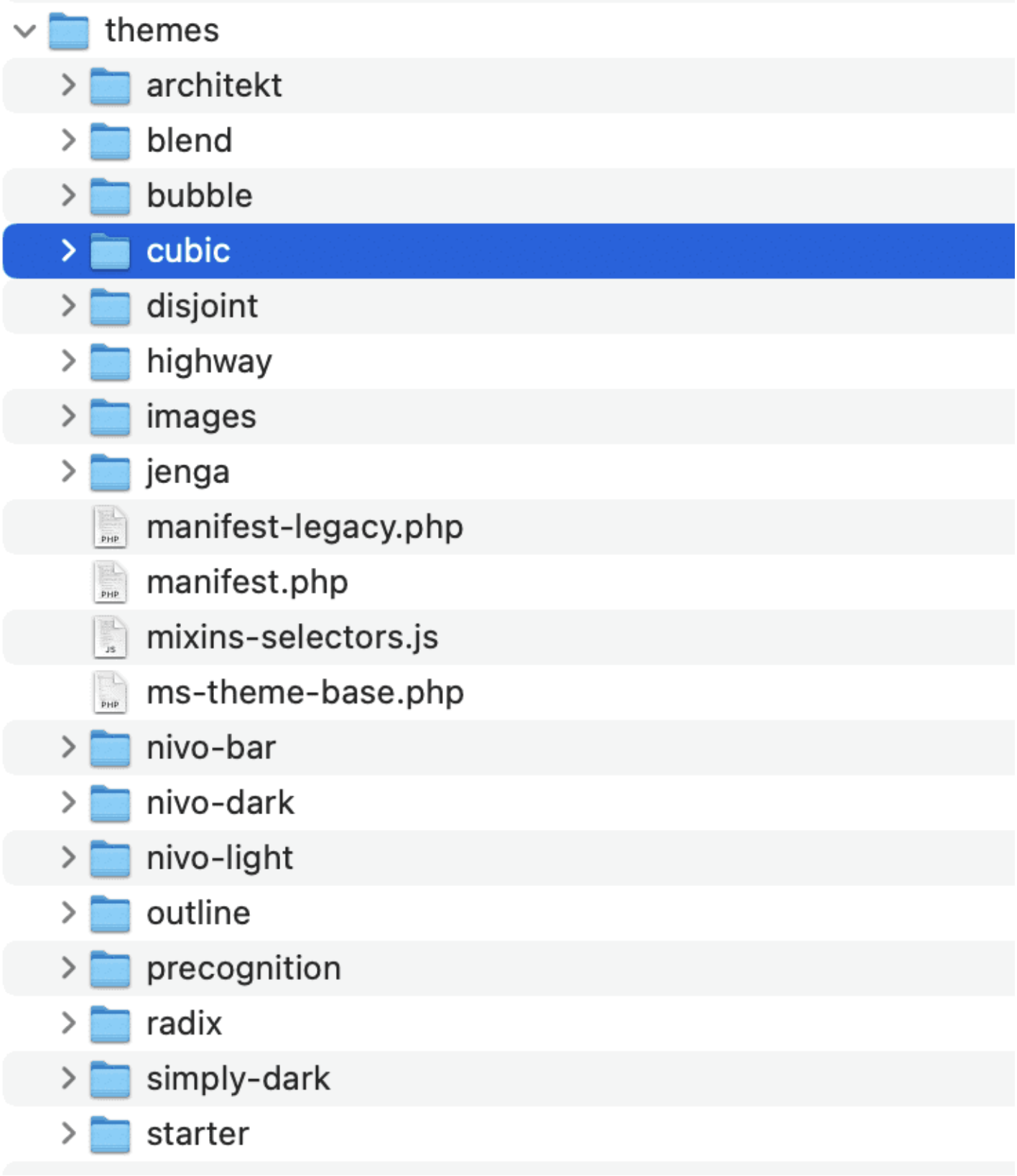
A Boilerplate MetaSlider Theme

We provide a boilerplate example of a custom MetaSlider theme. You can download this zip and decompress it to use as an example: Click to download the boilerplate.

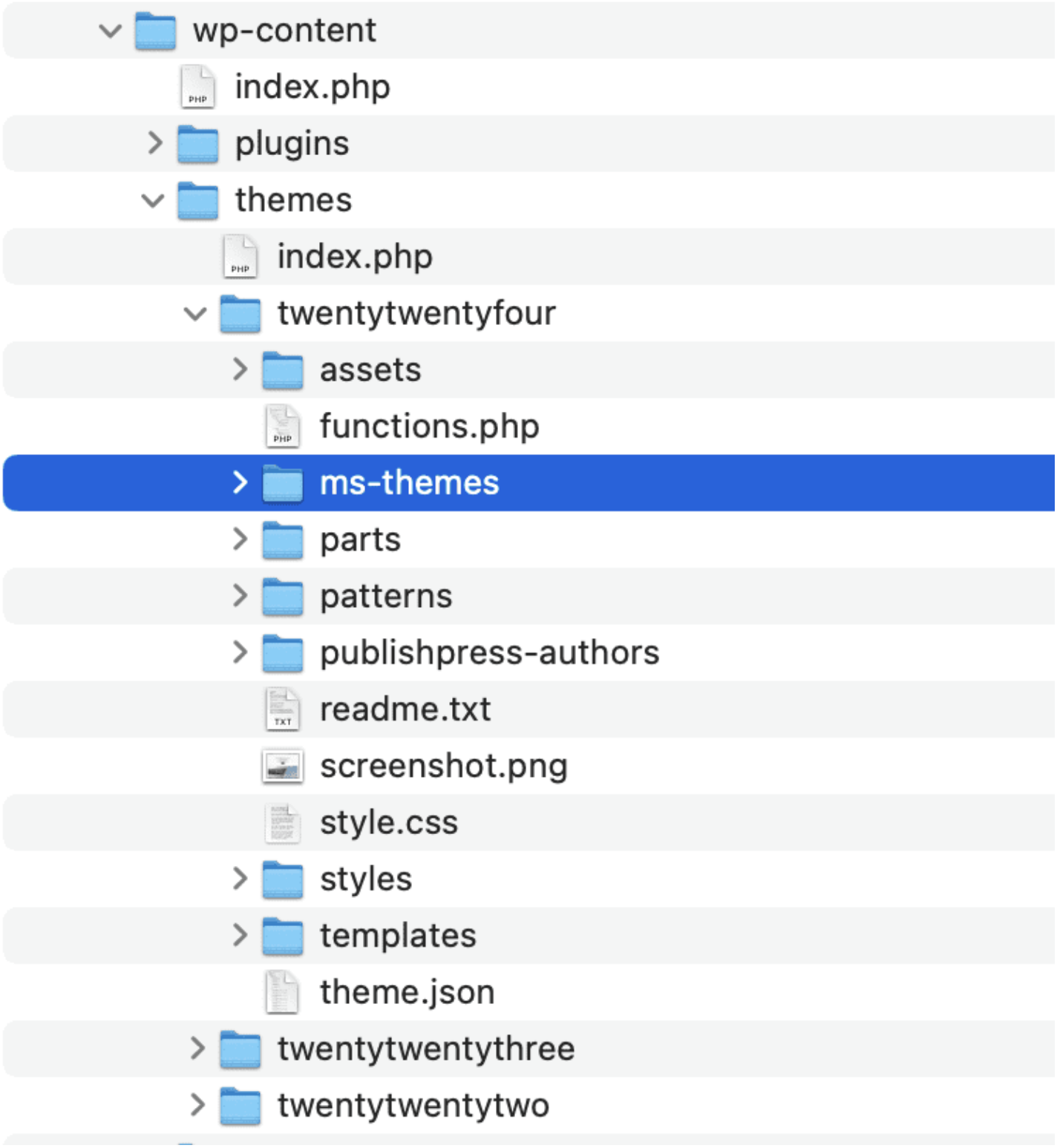
How to Customize an Existing Theme

To customize one of the existing themes provided with MetaSlider.

- Copy the folder of the theme from /wp-content/plugins/ml-slider/themes/. For example, we are going to take the "cubic"folder.



- Create a folder called "ms-themes" inside your WordPress theme.



- Paste the folder for your theme copy inside "ms-themes".

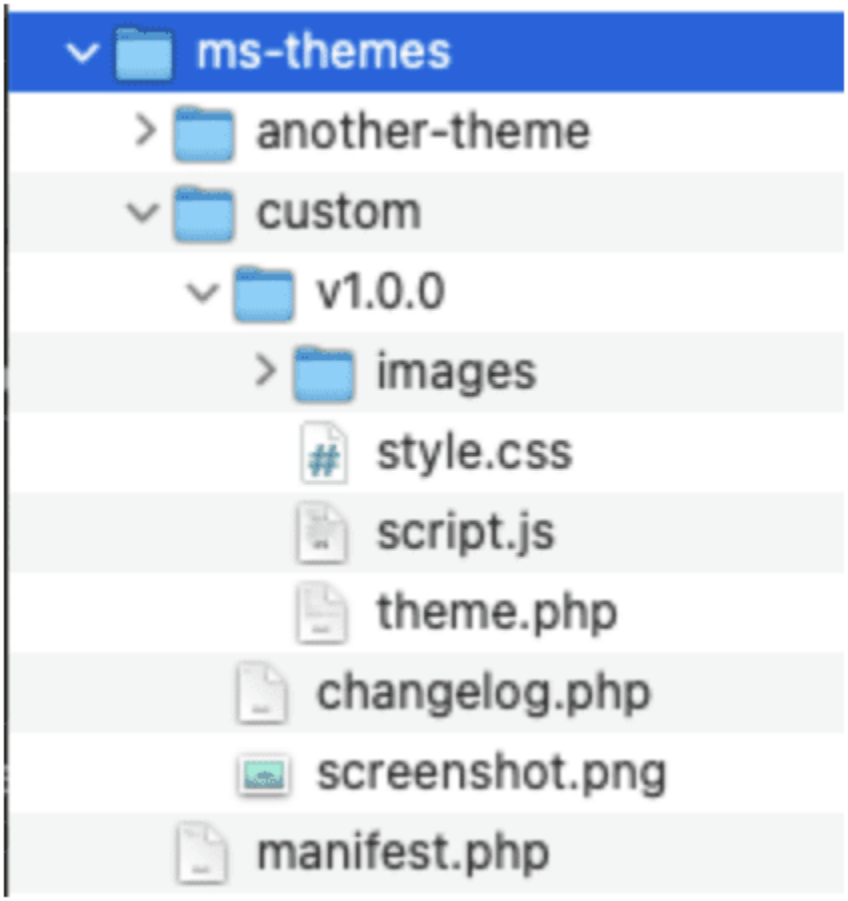
There are two things you will need to modify after moving the theme:

- MetaSlider does require that your theme has a unique name. So to modify a custom theme, you will need to replace the words 'cubic' and 'Cubic' words (case sensitive) with a custom name by checking the code from the duplicated files.
- You will need declare the custom theme through a manifest.php file inside /ms-themes/. This details for this file are available in the boilerplate above, and the text guide below.

How to Build Your Own MetaSlider Theme

This part of the guide will show you how to build your own MetaSlider theme from scratch.

- Add a "ms-themes" folder inside your active WordPress theme. In this example "custom" and "another-theme" are the names of our themes.
- Replace 'custom' and 'Custom' (case sensitive) with a unique name of your theme.



Theme Part #1. The manifest.php file

This file must be located in the "ms-themes" folder. The file will define all your custom themes. In this code example below we have declared "custom" and "another-theme". Please replace 'custom' and 'Custom' (case sensitive) with your own theme's name.

```
<?php
if ( !defined( 'ABSPATH' ) ) exit;

return array(
    'custom' => array(
        'folder' => 'custom',
        'title' => 'Custom Theme',
        'type' => 'external',
        'supports' => array( 'flex' ),
        'tags' => array('minimalist', 'lorem' ),
        'description' => 'Theme description goes here...',
        'screenshot_dir' => get_stylesheet_directory_uri() . '/ms-themes/custom'
    ),
    'another-theme' => array(
        'folder' => 'another-theme,
        'title' => 'Another Theme',
        'type' => 'external',
        'supports' => array( 'flex' ),
        'tags' => array('minimalist', 'lorem' ),
        'description' => 'Theme description goes here...',
        'screenshot_dir' => get_stylesheet_directory_uri() . '/ms-themes/another-theme'
    )
);
```

Theme Part #2. The theme.php file

This file is located in each theme folder. This file adds hooks to modify your custom theme and enqueues the CSS and JS. Please replace "custom" and "Custom" words (case sensitive) with the name of your own custom theme. Here are some replacement examples:

- Replace `Custom` (capitalize) word from `MetaSlider_Theme_Custom` with your own theme name in capitalize. e.g. `MetaSlider_Theme_Lorem`

- Replace `custom` (lowercase) from `public $id = 'custom'` with your own theme name in lowercase. e.g. `public $id = 'lorem'`

```
<?php

if (!defined('ABSPATH')) {
    die('No direct access.');
}

/**
 * Main theme file
 */
class MetaSlider_Theme_Custom extends MetaSlider_Theme_Base
{
    /**
     * Theme ID
     *
     * @var string
     */
    public $id = 'custom';

    /**
     * Theme Version
     *
     * @var string
     */
    public $version = '1.0.0';

    public function __construct()
    {
        parent::__construct($this->id, $this->version);
    }

    /**
     * Parameters
     *
     * @var string
     */
    public $slider_parameters = array();
```

```
    /**
     * Enqueues theme specific styles and scripts
     */
    public function enqueue_assets()
    {
        wp_enqueue_style(
            'metaslider_' . $this->id . '_theme_styles',
            get_stylesheet_directory_uri() . '/ms-themes/' . $this->id . '/v1.0.0/style.css',
            array('metaslider-public'),
            '1.0.0'
        );
        wp_enqueue_script(
            'metaslider_' . $this->id . '_theme_script',
            get_stylesheet_directory_uri() . '/ms-themes/' . $this->id . '/v1.0.0/script.js',
            array('jquery'),
            '1.0.0',
            true
        );
    }
}

if (!isset(MetaSlider_Theme_Base::$themes['custom'])) {
    new MetaSlider_Theme_Custom();
}
```

For consistency reasons we suggest to match `public $version` and `public $id` values with your theme folders:

Theme Part #3. The changelog.php file

This file is located in each theme folder.

```
<?php

return array(
    'v1.0.0' => array(
        'First version'
    )
);
```

Theme Part #4. The script.js file

We recommend to use the 'metaslider/initialized' event to modify the slideshow DOM and/or modify your custom theme. identifier is the slideshow ID, as example #metaslider_123. So to access each slideshow object using your custom theme check through `if( $(identifier).closest('.metaslider.ms-theme-custom').length ) { ... }` where 'custom' from '.metaslider.ms-theme-custom' is the name of your theme.

```
(function($) {
// metaslider has been initilalised
 $(document).on('metaslider/initialized', function(e, identifier) {
            if ($(identifier).closest('.metaslider.ms-theme-custom').length) {
                  // Add your custom JS here to apply to your theme with class 'metaslider.ms-theme-custom'
            }
 });

})(jQuery);
```

Theme Part #5. The style.css file

Here are some sample CSS selectors to use. Replace the 'custom' word with your own theme name in lowercase.

```
.metaslider.ms-theme-custom {
     /* Some CSS goes here */
}

.metaslider.ms-theme-custom .slides {
     /* Some CSS goes here */
}

.metaslider.ms-theme-custom .slides > li {
     /* Some CSS goes here */
}
```

Theme Part #6. The images folder

Use this folder for any images you make references to in the style.css folder.

Theme Part #7. screenshot.png file

This is the image that is displayed when selecting your theme. The recommended size is 400x267.

Theme Part #8. functions.php file

To make your custom MetaSlider theme available, in your current WordPress theme’s functions.php file add the code below:

```
add_filter( 'metaslider_extra_themes', 'custom_ms_themes', 10, 1 );
function custom_ms_themes( $themes )
{
    $themes[] = trailingslashit( get_stylesheet_directory() . '/ms-themes' );

    return $themes;
}
```