Introduction to customizing WordPress using functions.php

Budapest WordPress meetup

05/02/2015

Pro developers

Amater developers

Tweakers

Plug and Players
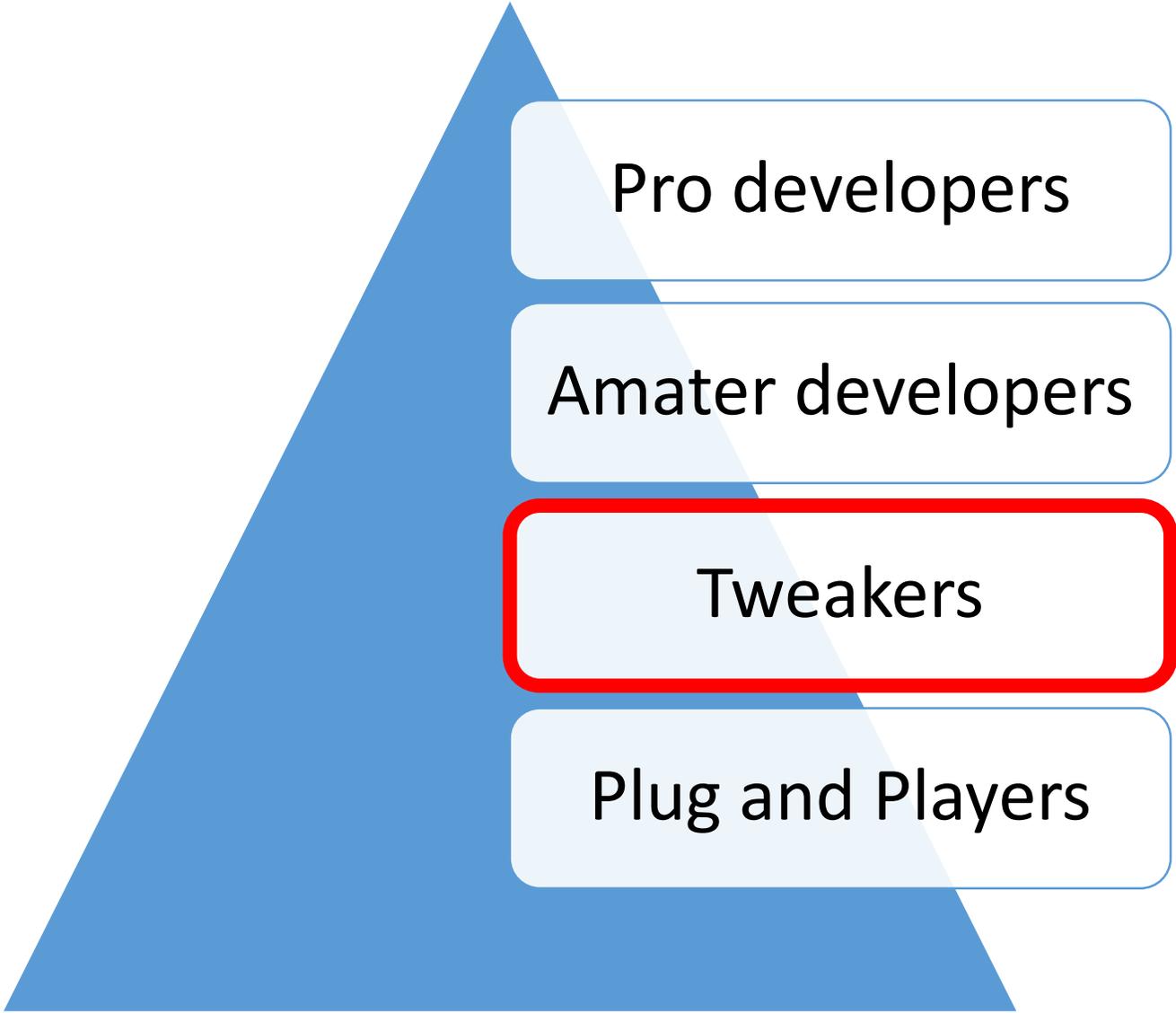
WordPress is a tweaker's paradise

How many times have you read on the forums?

Just paste this code
into your theme's **functions.php**

functions.php

"You can use it to call functions, both PHP and built-in WordPress, and to define your own functions"

(WordPress Codex)

+ automatically loaded in both admin and front-end pages

+ can add features/extend functionality of both theme and WP installation

- If theme is changed, features/functionality is lost

- messy if you plan on inserting more than a handful of code

- white screen of death

```
/root-folder
    /wp-admin
    /wp-content
        /plugins
        /themes
            /themename
                /functions.php
```

Remember: it is best NOT to touch the functions.php file in an existing theme, but rather create a «child theme» that only contains the modifications.

See: http://codex.wordpress.org/Child_Themes

Practical example

Add custom CSS class to first paragraph in blog posts

```php
function highlight_first_paragraph($content){

        global $post;

        if ($post->post_type == "post"){

                return preg_replace('/<p([^>]+)?>/', '<p$1 class="first">', $content, 1); }

        else {

                return $content;

        }

}


add_filter('the_content', 'highlight_first_paragraph');
```

BUT...

design and functionality should be clearly separated

design -> themes
functionality -> plugins

plugin

- Requires specific, unique Header text

- Is stored in wp-content/plugins, usually in a subdirectory

- Executes only when individually activated, via the Plugins panel

- Applies to all themes

- Best if meant for a single purpose

Example: Define header for custom plugin

```php
<?php
/**
 * Plugin Name: Custom Functions
 * Plugin URI: http://wpbudapest.com
 * Description: Custom functions for WP site (no UI).
 * Author: LS
 * Author URI: http://wpbudapest.com
 * Version: 1.0
 */
```

Finding a middle ground

1. create a folder in your child theme and recursively include the snippets
2. Include php snippets in that folder programatically from functions.php

+ easier to maintain

+ less mess

1. Create folder and php files

```
/root-folder
    /wp-admin
    /wp-content
        /plugins
        /themes
            /themename
                /snippets
                    /login_custom_logo.php
                    /post_first_paragraph.php
                    /etc.
```

2. Add code to functions.php

```php
/* include all PHP files in ./snippets/ folder */
foreach ( glob( dirname( __FILE__ ) . '/snippets/*.php' ) as $file )
        include $file;
```

Alternatives

So-called «functionality» plugins

Functionality

https://wordpress.org/plugins/functionality/

My Custom Functions

https://wordpress.org/plugins/my-custom-functions/