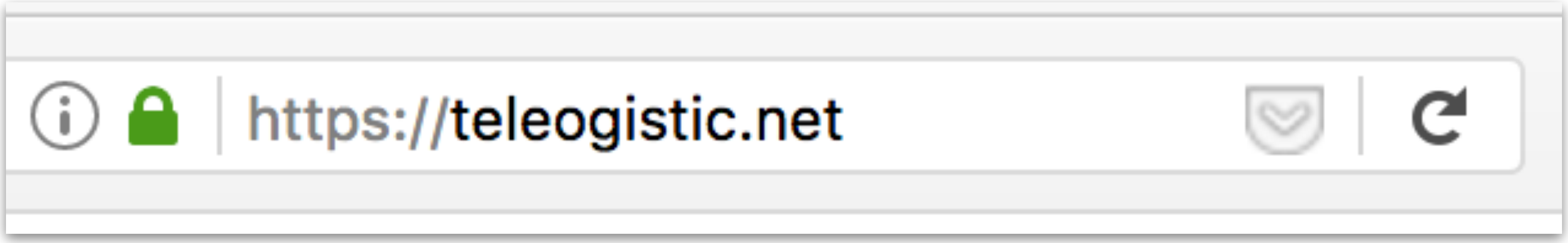
Interrupting WordPress

Boone Gorges
WordCamp Chicago 2017
April 29, 2017

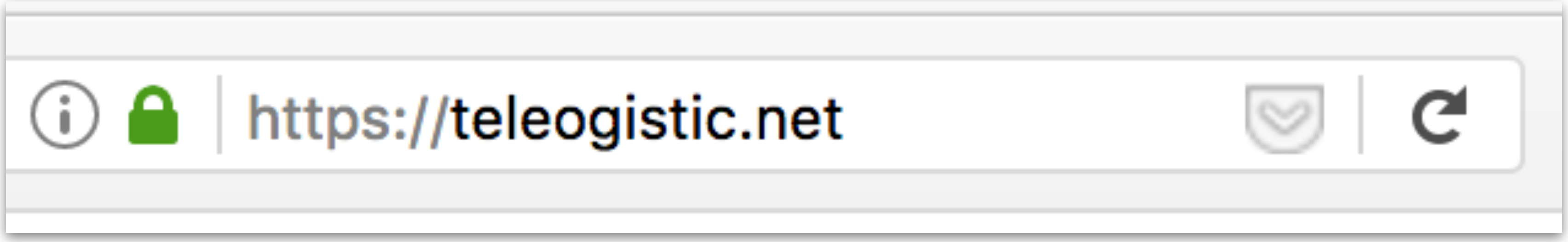
Boone

- WordPress core committer
- BuddyPress lead developer
- Developer and consultant, Hard G
- Rocks extremely hard

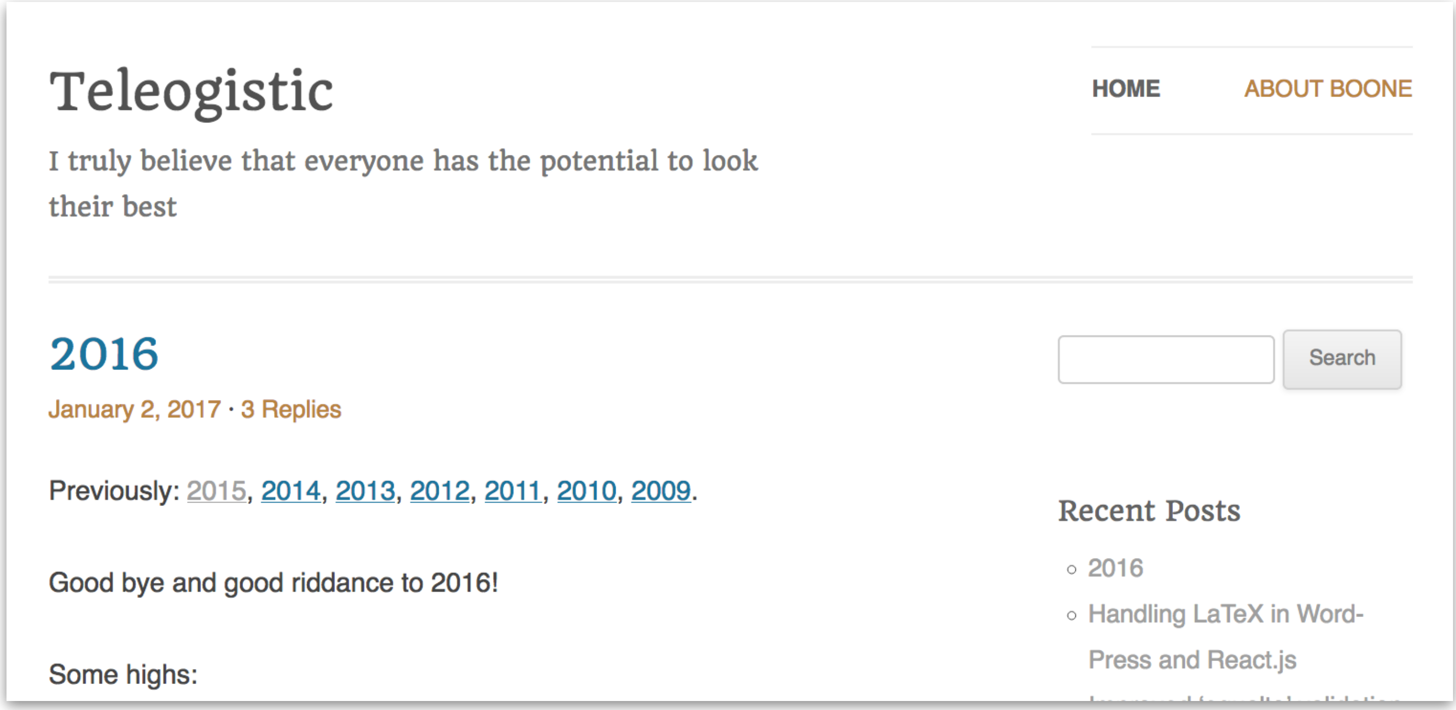
[warm up crowd
with hilarious joke]



https://teleogistic.net



https://teleogistic.net



Teleogistic

I truly believe that everyone has the potential to look their best

2016

January 2, 2017 · 3 Replies

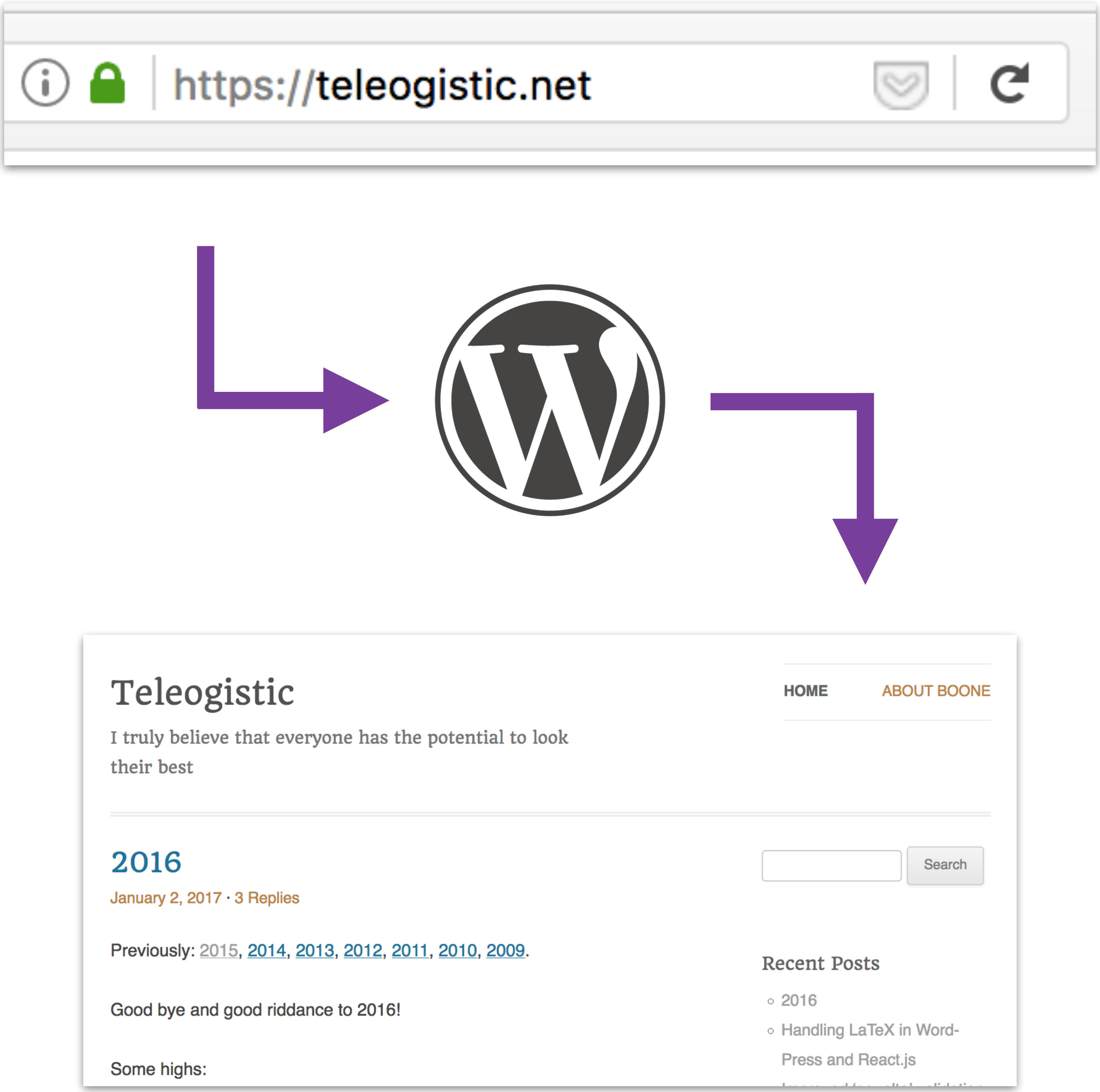
Previously: 2015, 2014, 2013, 2012, 2011, 2010, 2009.

Good bye and good riddance to 2016!

Some highs:

Recent Posts

- 2016
- Handling LaTeX in WordPress and React.js



https://teleogistic.net

Teleogistic

I truly believe that everyone has the potential to look their best

HOME ABOUT BOONE

2016

January 2, 2017 · 3 Replies

Previously: 2015, 2014, 2013, 2012, 2011, 2010, 2009.

Good bye and good riddance to 2016!

Some highs:

Recent Posts

- 2016
- Handling LaTeX in WordPress and React.js

The Web Page

theme templates > HTML
plugins that modify output
CSS
JavaScript

Magic

URL parsing > rewrites > query vars >
database query > template loader

The Web Page

theme templates > HTML
plugins that modify output
CSS
JavaScript

Scary

URL parsing > rewrites > query vars >
database query > template loader

The Web Page

theme templates > HTML
plugins that modify output
CSS
JavaScript

Black Box

URL parsing > rewrites > query vars >
database query > template loader

The Web Page

theme templates > HTML
plugins that modify output
CSS
JavaScript

Auuuuuuuuugh

URL parsing > rewrites > query vars >
database query > template loader

The Web Page

theme templates > HTML
plugins that modify output
CSS
JavaScript

Copy and Paste from Stack Overflow

URL parsing > rewrites > query vars >
database query > template loader

The Web Page

theme templates > HTML
plugins that modify output
CSS
JavaScript

Terminology

bootstrap

Terminology

action

```
do_action( ‘init’ );

add_action( ‘init’, ‘my_callback’ );
```

Terminology

filter

```
$template = apply_filters(
   ‘template_include’,
   $template
);

add_filter( ‘template_include’, ‘my_callback’ );
```

Where

What

Why

Where

`plugins_loaded`

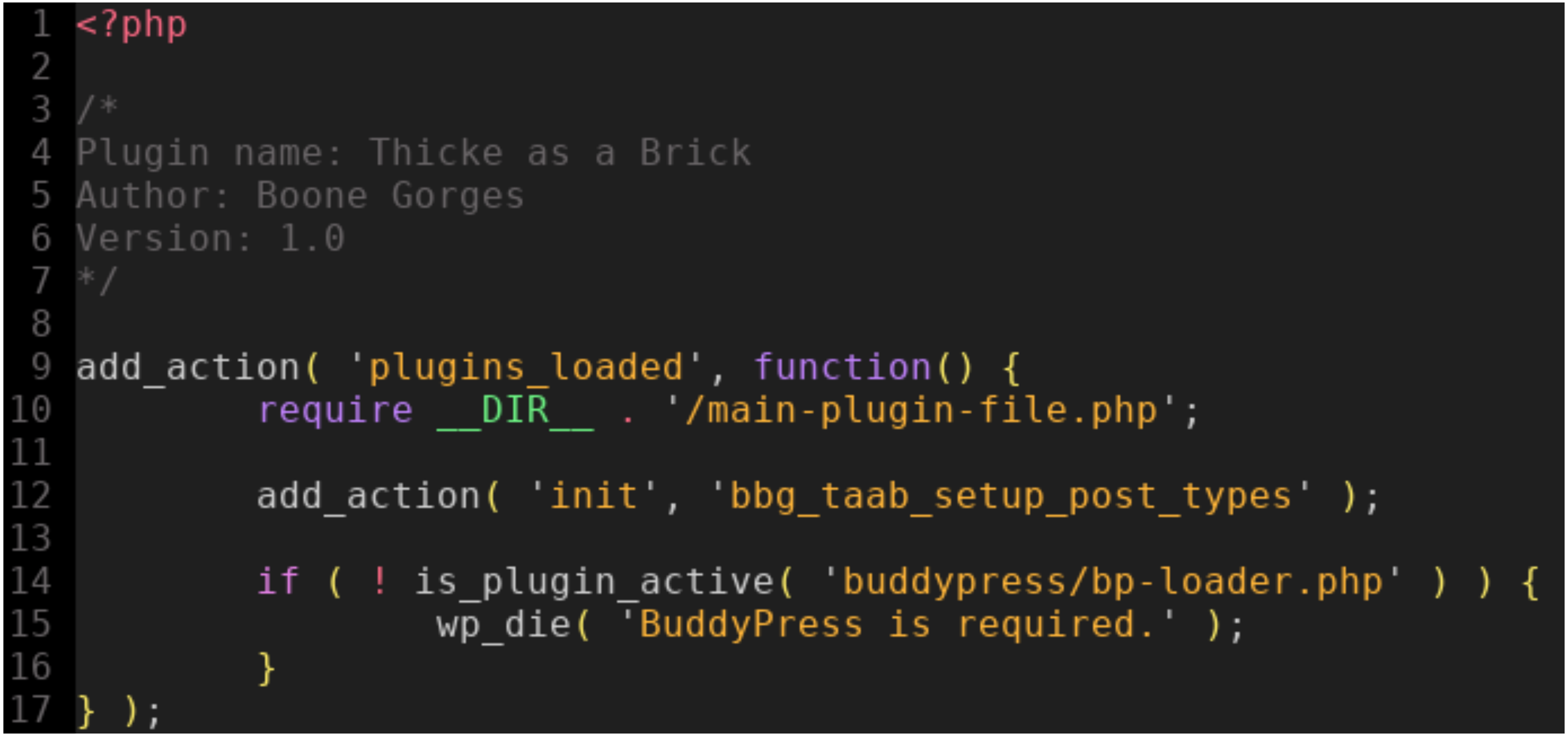
What

Core libraries; plugin loader files

Why

One of the earliest places you can hook into WordPress.

- Plugin dependencies and libraries
- Main plugin bootstrap
- Pluggable functions



```php
<?php

/*
Plugin name: Thicke as a Brick
Author: Boone Gorges
Version: 1.0
*/

add_action( 'plugins_loaded', function() {
        require __DIR__ . '/main-plugin-file.php';

        add_action( 'init', 'bbg_taab_setup_post_types' );

        if ( ! is_plugin_active( 'buddypress/bp-loader.php' ) ) {
                wp_die( 'BuddyPress is required.' );
        }
} );
```

wp-content/plugins/my-plugin/my-plugin.php

Where

`after_setup_theme`

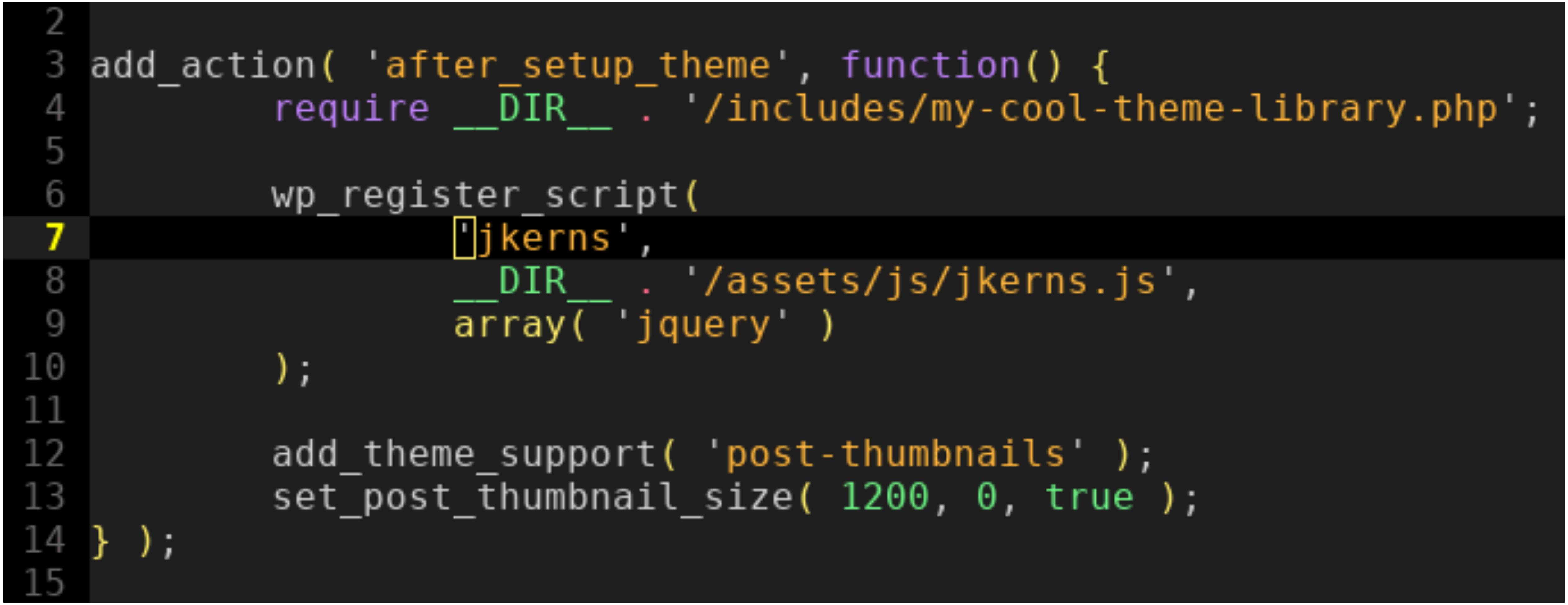
What

Language files, `functions.php` from theme

Why

Earliest place you can hook into WP
from a theme.

- Register CSS/JS assets
- Declare theme support
- Load theme libraries



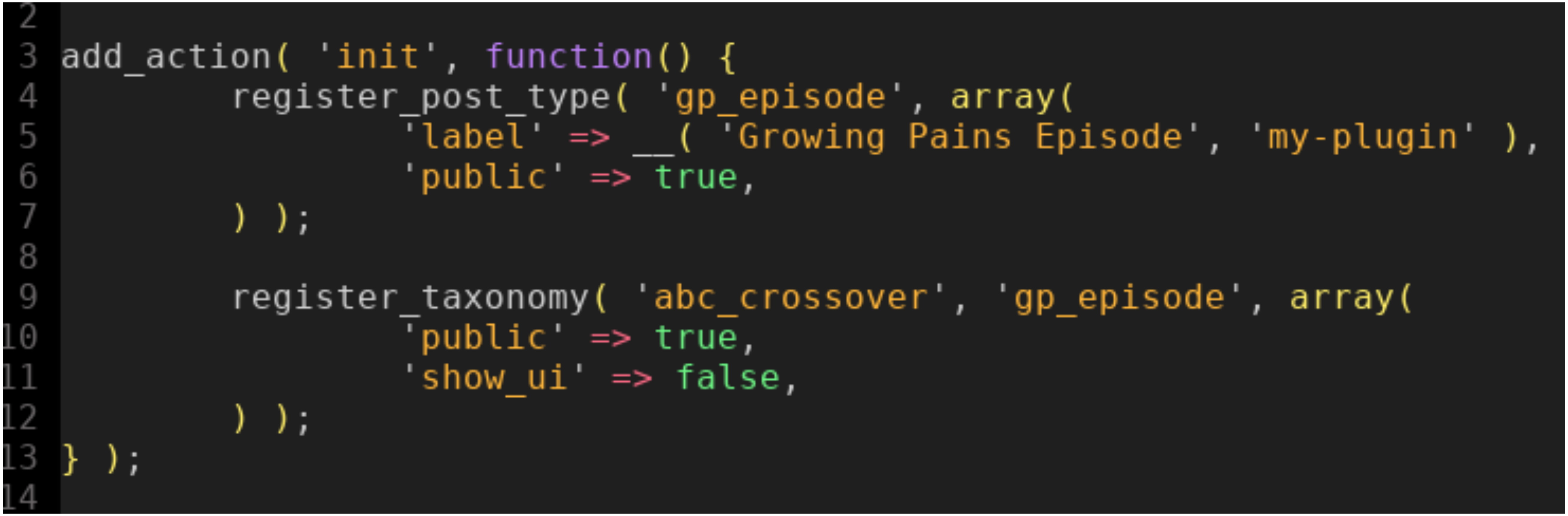
```php

add_action( 'after_setup_theme', function() {
        require __DIR__ . '/includes/my-cool-theme-library.php';

        wp_register_script(
                'jkerns',
                __DIR__ . '/assets/js/jkerns.js',
                array( 'jquery' )
        );

        add_theme_support( 'post-thumbnails' );
        set_post_thumbnail_size( 1200, 0, true );
} );

```

wp-content/themes/my-theme/functions.php

Where `init`

What Much of WP is set up on this hook, including core post types and taxonomies, REST API, and user authentication

Why Register your post types, taxonomies, post statuses, etc here - but don’t depend on other plugins’ having done so yet



```php

add_action( 'init', function() {
        register_post_type( 'gp_episode', array(
                'label' => __( 'Growing Pains Episode', 'my-plugin' ),
                'public' => true,
        ) );

        register_taxonomy( 'abc_crossover', 'gp_episode', array(
                'public' => true,
                'show_ui' => false,
        ) );
} );

```

wp-content/plugins/my-plugin/my-plugin.php

Where

`wp_loaded`

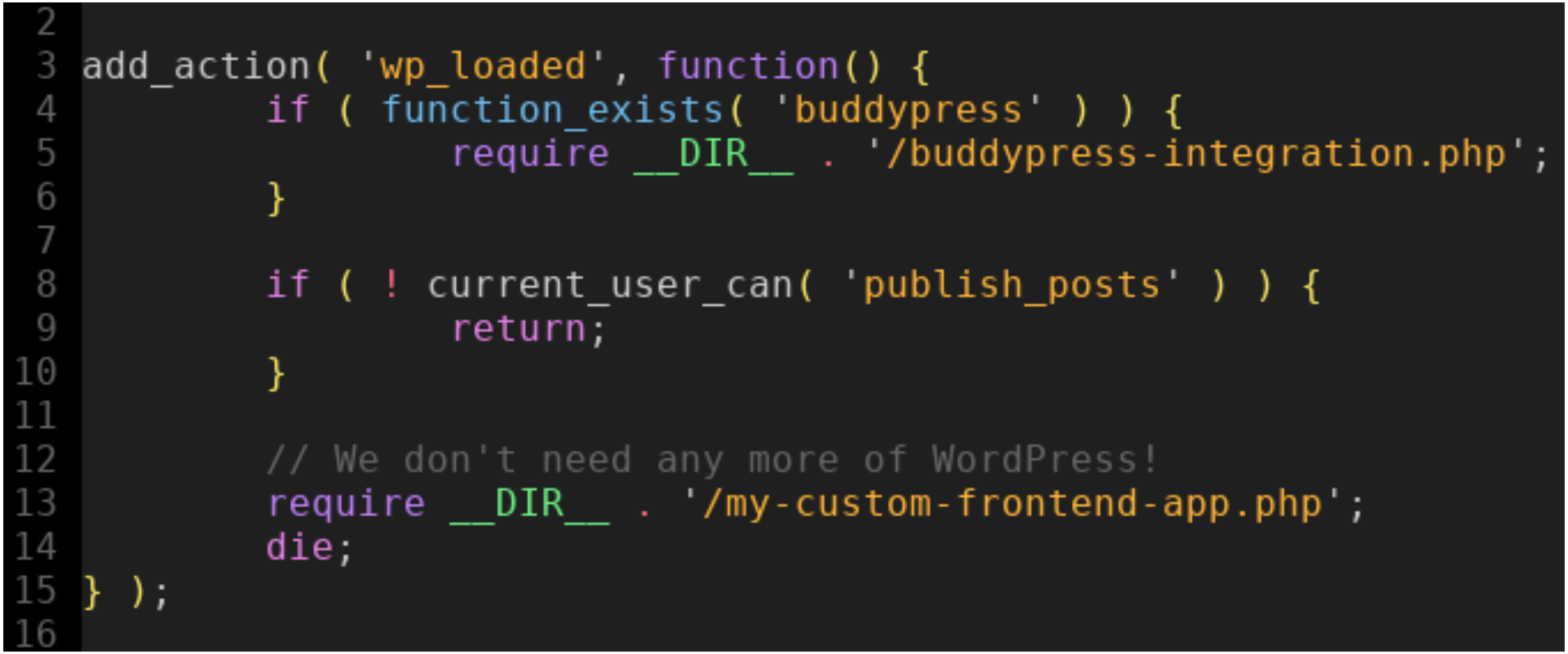
What

Core fully loaded, including user auth.
Plugins should have initialized fully by now.

Why

Operations that require access to plugins,
but don’t depend on request-specific data.

- Load plugin-interdependent libraries
- Kill default output (use REST API instead!)



```php
add_action( 'wp_loaded', function() {
        if ( function_exists( 'buddypress' ) ) {
                require __DIR__ . '/buddypress-integration.php';
        }

        if ( ! current_user_can( 'publish_posts' ) ) {
                return;
        }

        // We don't need any more of WordPress!
        require __DIR__ . '/my-custom-frontend-app.php';
        die;
} );
```

wp-content/plugins/my-plugin/my-plugin.php

Where

`parse_request`

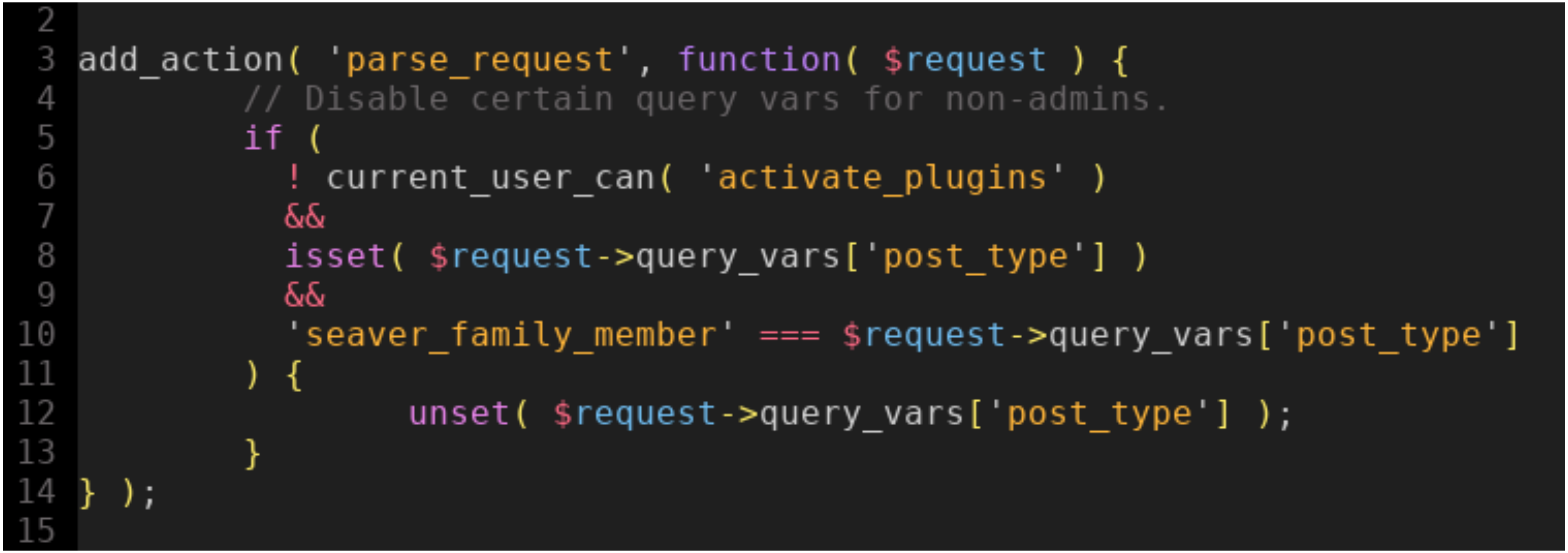
What

Request URL has been parsed according to rewrite rules/permalink structure.

Why

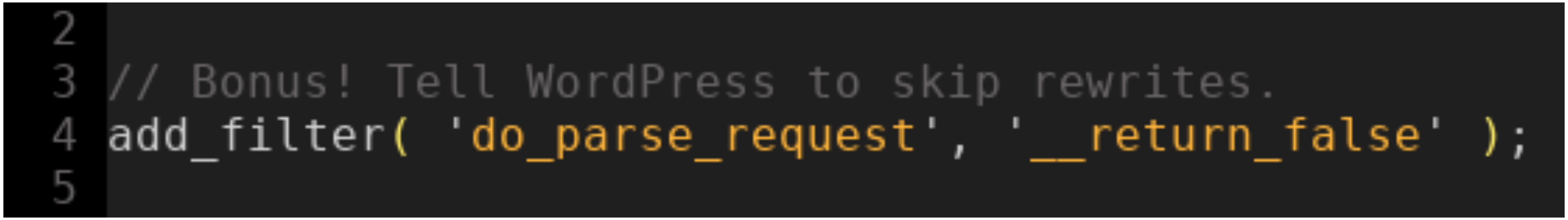
Modify or bypass the way that WP turns URL into query vars. Last chance to interrupt before headers begin to be sent to client.

- Custom/dynamic query vars
- Use rewrite API without `WP_Query`



```
add_action( 'parse_request', function( $request ) {
        // Disable certain query vars for non-admins.
        if (
          ! current_user_can( 'activate_plugins' )
          &&
          isset( $request->query_vars['post_type'] )
          &&
          'seaver_family_member' === $request->query_vars['post_type']
        ) {
                unset( $request->query_vars['post_type'] );
        }
} );
```

wp-content/plugins/my-plugin/my-plugin.php



```php

// Bonus! Tell WordPress to skip rewrites.
add_filter( 'do_parse_request', '__return_false' );

```

wp-content/plugins/my-plugin/my-plugin.php

Where

`pre_get_posts`

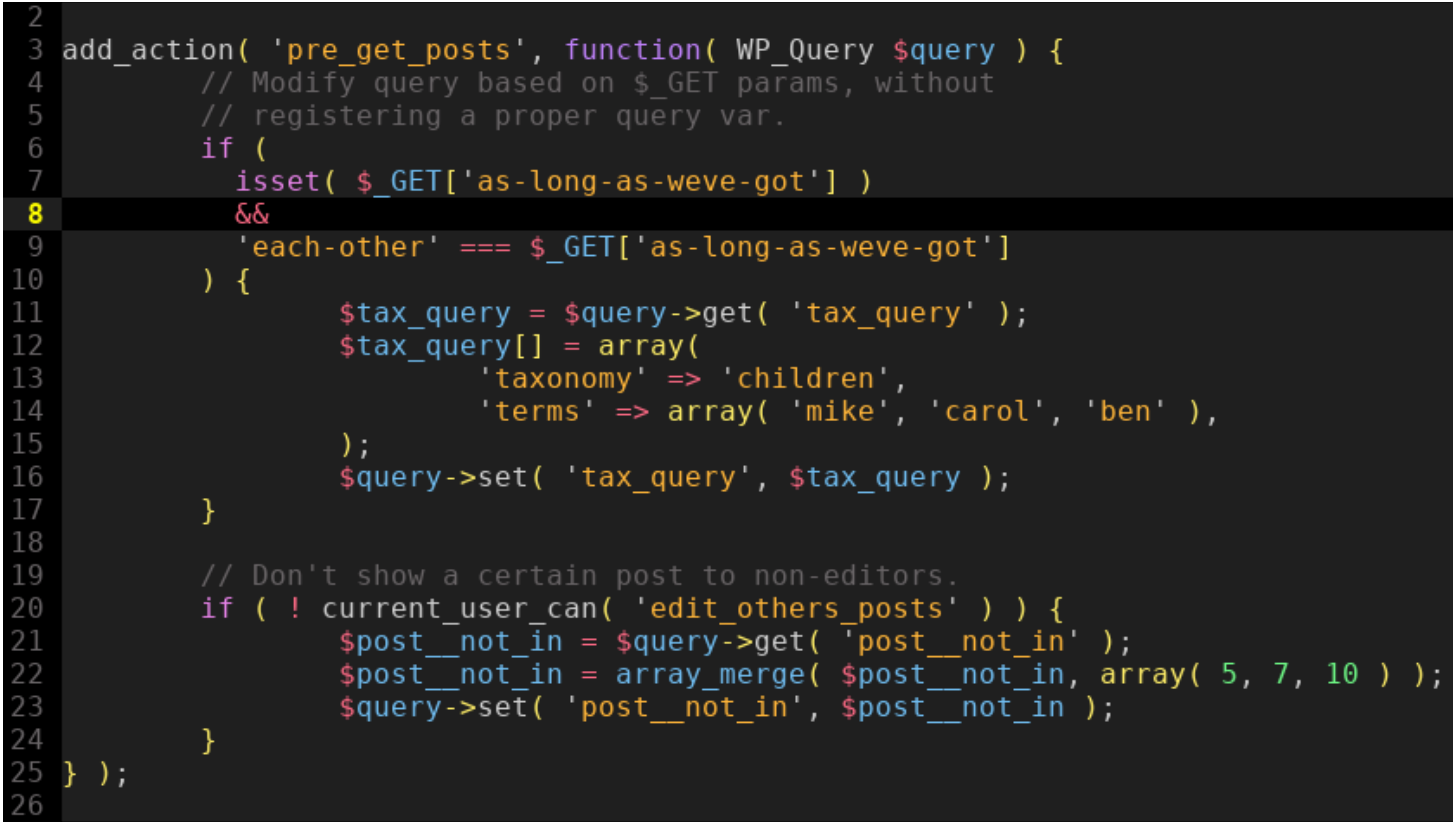
What

WP_Query has been built from query vars, but query has not yet been run.

Why

Last chance to modify the main database query.

- Poor-man’s query vars
- Exclusions based on role, etc



```php

add_action( 'pre_get_posts', function( WP_Query $query ) {
        // Modify query based on $_GET params, without
        // registering a proper query var.
        if (
          isset( $_GET['as-long-as-weve-got'] )
          &&
          'each-other' === $_GET['as-long-as-weve-got']
        ) {
                $tax_query = $query->get( 'tax_query' );
                $tax_query[] = array(
                        'taxonomy' => 'children',
                        'terms' => array( 'mike', 'carol', 'ben' ),
                );
                $query->set( 'tax_query', $tax_query );
        }

        // Don't show a certain post to non-editors.
        if ( ! current_user_can( 'edit_others_posts' ) ) {
                $post__not_in = $query->get( 'post__not_in' );
                $post__not_in = array_merge( $post__not_in, array( 5, 7, 10 ) );
                $query->set( 'post__not_in', $post__not_in );
        }
} );

```

wp-content/plugins/my-plugin/my-plugin.php

Where

`wp`

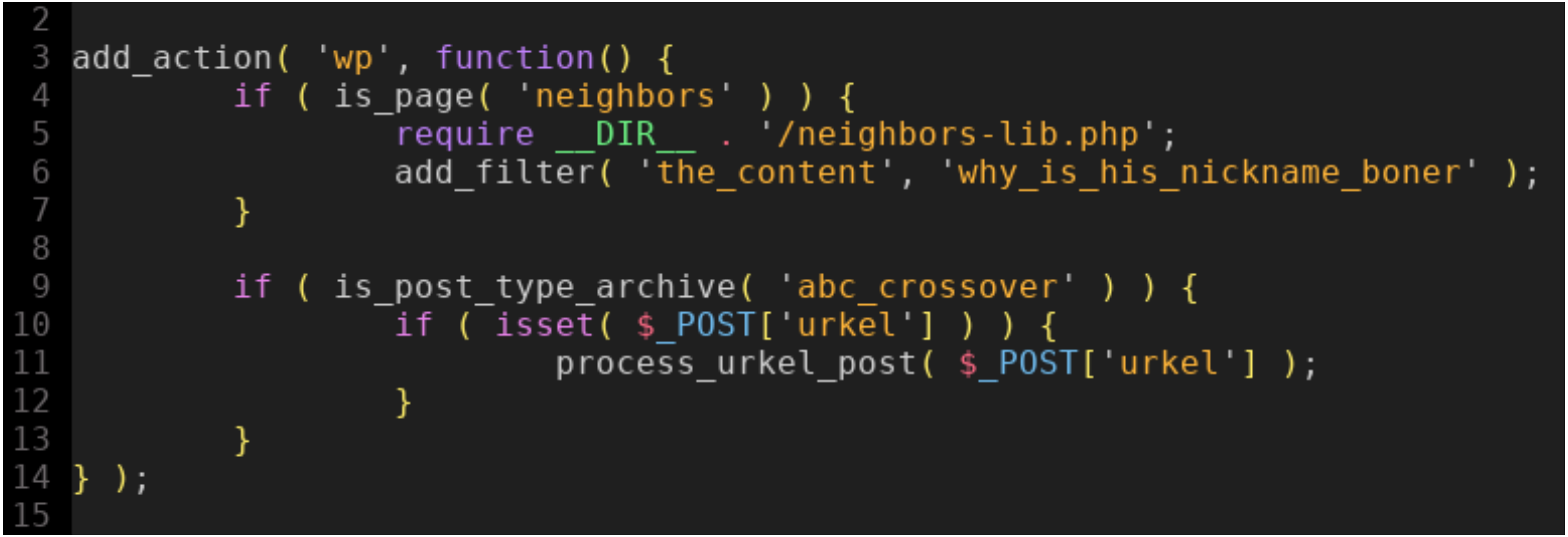
What

Main query has been run.

Why

By this point, you know which WP item (post, page, etc) has been requested.

- Custom template loaders
- POST controllers
- Post-specific libraries



```php

add_action( 'wp', function() {
        if ( is_page( 'neighbors' ) ) {
                require __DIR__ . '/neighbors-lib.php';
                add_filter( 'the_content', 'why_is_his_nickname_boner' );
        }

        if ( is_post_type_archive( 'abc_crossover' ) ) {
                if ( isset( $_POST['urkel'] ) ) {
                        process_urkel_post( $_POST['urkel'] );
                }
        }
} );

```

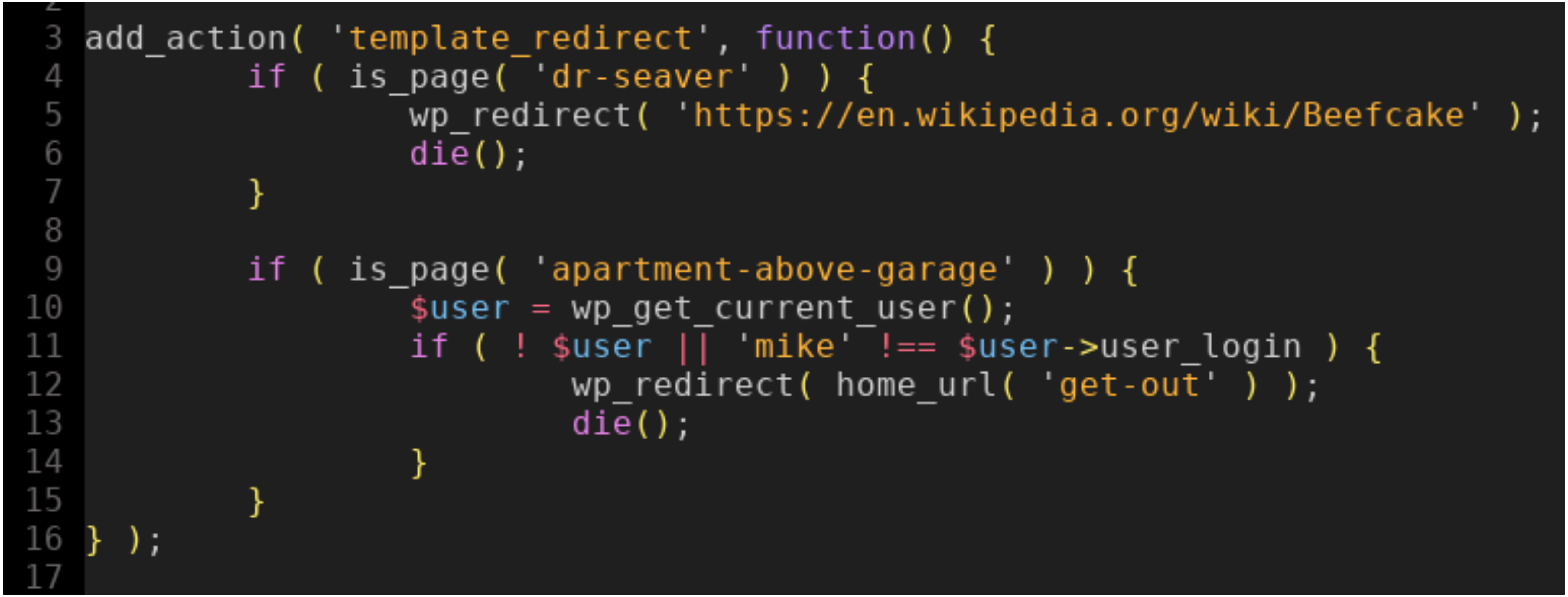
wp-content/plugins/my-plugin/my-plugin.php

Where `template_redirect`

What Canonical redirect, including “old slug” redirects, happens here.

Why Last chance to safely perform a redirect before template HTML is generated and sent to client.

- Custom redirects
- Authorization checks



```php
add_action( 'template_redirect', function() {
        if ( is_page( 'dr-seaver' ) ) {
                wp_redirect( 'https://en.wikipedia.org/wiki/Beefcake' );
                die();
        }

        if ( is_page( 'apartment-above-garage' ) ) {
                $user = wp_get_current_user();
                if ( ! $user || 'mike' !== $user->user_login ) {
                        wp_redirect( home_url( 'get-out' ) );
                        die();
                }
        }
} );

```

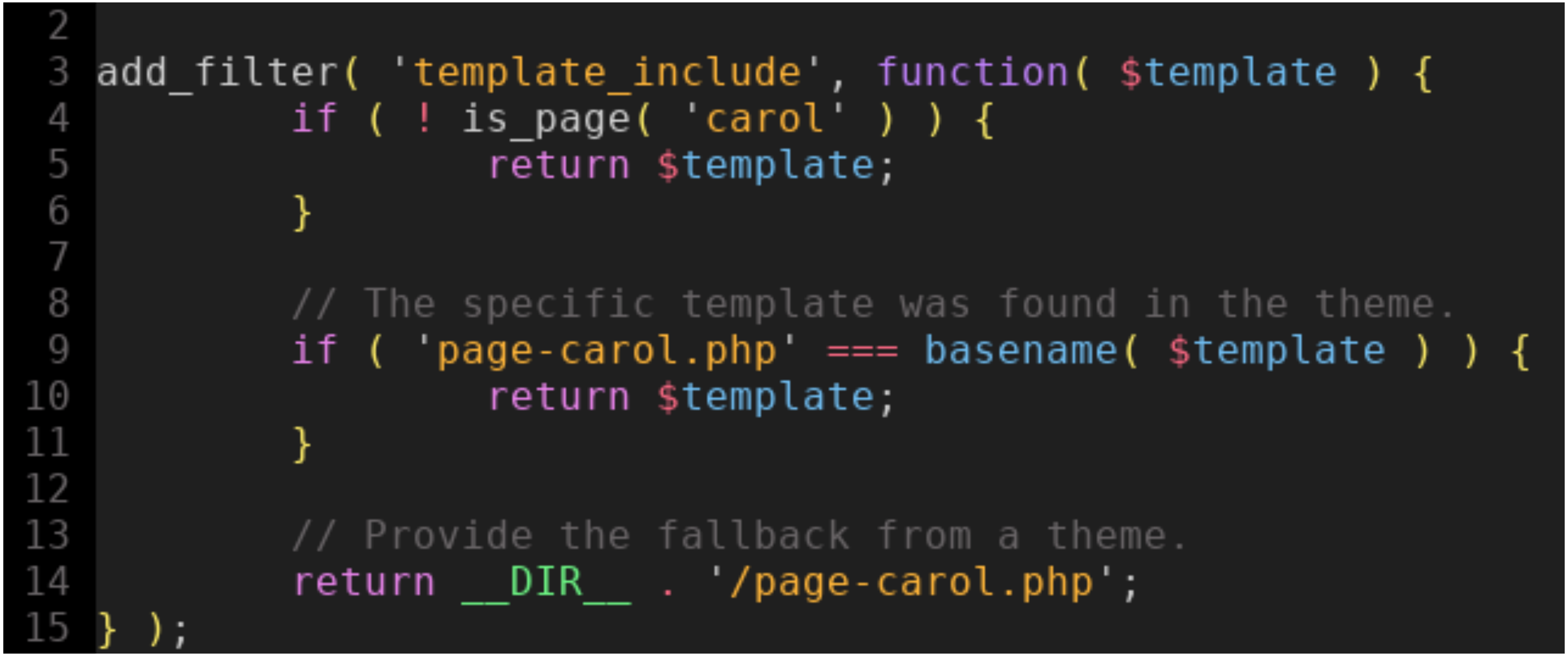
wp-content/plugins/my-plugin/my-plugin.php

Where `template_include`

What Filters the template located by WP’s template hierarchy just before including.

Why Modify the selected template on the fly.

- Provide missing theme templates from a plugin
- Miscellaneous custom template logic



```php

add_filter( 'template_include', function( $template ) {
        if ( ! is_page( 'carol' ) ) {
                return $template;
        }

        // The specific template was found in the theme.
        if ( 'page-carol.php' === basename( $template ) ) {
                return $template;
        }

        // Provide the fallback from a theme.
        return __DIR__ . '/page-carol.php';
} );
```

wp-content/plugins/my-plugin/my-plugin.php

Magic

URL parsing > rewrites > query vars >
database query > template loader

The Web Page

theme templates > HTML
plugins that modify output
CSS
JavaScript

Thanks