

Odczarowujemy WP_Rewrite

Krzysiek Dróżdź
krzysiek@wpmagus.pl

Wpmagus.pl

Wrocław, 27 czerwca 2013

Wstęp

Częste problemy i błędy
Co można z tym w ogóle zrobić?

Dowiedzmy się, co wiemy

Czego dowiemy się z Codexu
Czego dowiemy się z kodu WP
Przydatne przy debugowaniu

Kto z Was...?

- ...słyszał?



Kto z Was...?

- ...słyszał?
- ...korzystał?

Kto z Was...?

- ...słyszał?
- ...korzystał?
- ...miał problemy?

Kto z Was...?

- ...słyszał?
- ...korzystał?
- ...miał problemy?
- ...używa często?

Kto z Was...?

- ...słyszał?
- ...korzystał?
- ...miał problemy?
- ...używa często?

Czym jest WP_Rewrite

WP_Rewrite:

- to klasa WordPressa
- odpowiada za obsługę przyjaznych linków
- przechowuje ustawienia w bazie

Funkcje związane z WP_Rewrite

- `add_rewrite_rule()`
- `flush_rewrite_rules()`
- `add_rewrite_tag()`
- `add_permastruct()`
- `add_rewrite_endpoint()`
- `add_feed()`

add_rewrite_rule()

Pozwala dodać własne reguły.

Za pomocą wyrażeń regularnych definiujemy, jak sparsować przyjazny link i na jego podstawie *wyliczyć* parametry requestu.

Przykład

```
add_rewrite_rule(
    '^nutrition/([^/]*)/([^\s]*)/?',
    'index.php?page_id=12&food=$matches[1]&variety=$matches[2]',
    'top');
```

Sprawi, że obsługiwane zaczną być adresy
strona.pl/nutrition/milkshakes/vanilla. Takie requesty
będą przekierowane na stronę o ID=12.

flush_rewrite_rules()

Służy do odświeżania rewrite rulesów

Jak to się układa w czasie?

- `do_action(plugins_loaded)`
- **utworzenie globalnej zmiennej `wp_rewrite`**
- `do_action(setup_theme)`
- `do_action(after_setup_theme)`
- `do_action(init)`
- `do_action(wp_loaded)`
- **parsowanie requestu**
- wysłanie nagłówek
- wykonanie głównego `WP_Query`
- `do_action_ref_array(wp)`
- `apply_filters(template_include)`

Kilka faktów o rewrite rulesach

- są generowane na podstawie permastructów i endpointów
- są cache'owane i zapamiętywane w bazie (opcja `rewrite_rules`)
- są wyrażeniami regularnymi
- odpowiadają za parsowanie requestu i *wyliczenie* rewrite tagów

Wyświetl wszystkie reguły

```
function wpm_debug_rewrite_rules() {
    global $wp_rewrite;

    echo '<div>';
    if (!empty($wp_rewrite->rules)) {
        echo '<table><thead><tr><td>Rule</td><td>Rewrite</td></tr></thead><tbody>';
        foreach ($wp_rewrite->rules as $name => $value) {
            echo '<tr><td>'. esc_html($name) .'</td><td>'. esc_html($value) .'</td></tr>';
        }
        echo '</tbody></table>';
    } else {
        echo 'NO RULES';
    }
    echo '</div>';
}
```

Wyświetl użytą/dopasowaną regułę

```
function wpm_debug_page_request() {  
    global $wp, $template;  
    echo '<!-- Request: '. esc_html($wp->request) .' -->'.PHP_EOL;  
    echo '<!-- Matched Rewrite Rule: '. esc_html($wp->matched_rule) .' -->'.PHP_EOL;  
    echo '<!-- Matched Rewrite Query: '. esc_html($wp->matched_query) .' -->'.PHP_EOL;  
    echo '<!-- Loaded Template: '. esc_html(basename($template)) .' -->'.PHP_EOL;  
}
```

Zdarzyło się każdemu... ?

- Tworzymy własny typ wpisu za pomocą zgodnie z przykładem z Codex
- Dodajemy wpis, publikujemy i chcemy go zobaczyć (lub odwiedzamy stronę archiwum tego typu wpisu)
- Widzimy 404

Co się wtedy zazwyczaj robi?

- Szuka w Google
- Trafia na informację, że trzeba *sflushować rewrite rulesy*
- ... i że wystarczy w tym celu zapisać ustawienia bezpośrednich odnośników

Co się wtedy zazwyczaj robi?

- Szuka w Google
- Trafia na informację, że trzeba *sflushować rewrite rulesy*
- ... i że wystarczy w tym celu zapisać ustawienia bezpośrednich odnośników

Ale to niezbyt dobre rozwiązanie!

Z czego wynika problem?

- Rejestracja CPT powoduje utworzenie nowego permastructa
- ...Ale nie generuje rewrite rulesów

Z czego wynika problem?

- Rejestracja CPT powoduje utworzenie nowego permastructa
- ...Ale nie generuje rewrite rulesów
- Zatem WordPress nie rozpoznaje tego formatu adresu URL

Co zatem zrobić?

- *Sflushować rewrite rulesy ;)*
- czyli wywołać funkcję `flush_rewrite_rules`

Co zatem zrobić?

- *Sflushować rewrite rulesy ;)*
- czyli wywołać funkcję `flush_rewrite_rules`

Zatem dodajemy wywołanie tej funkcji zaraz pod kodem rejestrującym CPT i wszystko zaczyna działać.

Co zatem zrobić?

- *Sflushować rewrite rulesy ;)*
- czyli wywołać funkcję `flush_rewrite_rules`

Zatem dodajemy wywołanie tej funkcji zaraz pod kodem rejestrującym CPT i wszystko zaczyna działać.

I tak się zazwyczaj kończy. . .

Jak zatem zrobić to poprawnie?

WordPress nie robi tego automatycznie po dodaniu permastructa, bo...

Jak zatem zrobić to poprawnie?

WordPress nie robi tego automatycznie po dodaniu permastructa, bo...

- wygenerowanie rewrite rulesów jest czasochłonne
- i powoduje zapis do bazy danych

Jak zatem zrobić to poprawnie?

WordPress nie robi tego automatycznie po dodaniu permastructa, bo...

- wygenerowanie rewrite rulesów jest czasochłonne
- i powoduje zapis do bazy danych

Czyli nie chcemy tego robić podczas każdej odsłony strony.

To co teraz?

Musimy wywołać `flush_rewrite_rules` w odpowiednim momencie i tylko raz (wtedy, gdy coś zmieniamy).

To co teraz?

Musimy wywołać `flush_rewrite_rules` w odpowiednim momencie i tylko raz (wtedy, gdy coś zmieniamy).

- w motywie wykorzystajmy `after_switch_theme`
- we wtyczce - `activation_hook` i `deactivation_hook`

To wystarczy w większości wypadków.

Kto pierwszy ten lepszy

Jeśli Twoja reguła powinna zostać dopasowana do adresu strony, ale tak nie jest, to pamiętaj, że reguły dopasowywane są po kolei. Możliwe, że jakaś reguła, która jest wyżej od Twojej, została już dopasowana.

Kto pierwszy ten lepszy

Jeśli Twoja reguła powinna zostać dopasowana do adresu strony, ale tak nie jest, to pamiętaj, że reguły dopasowywane są po kolei. Możliwe, że jakaś reguła, która jest wyżej od Twojej, została już dopasowana.

Tutaj przydają się funkcje, które wcześniej pokazałem.

Trochę nieszkodliwego zamieszania

Dlaczego działają:

`strona.pl/testowa-strona` i `strona.pl/TeStoWa-StrONa`
oraz

`strona.pl/category/testowa-kategoria` i

`strona.pl/category/tEsTOWA-kAtEgORiA` a nie działają:

`strona.pl/Feed` i `strona.pl/Category?`

Trochę nieszkodliwego zamieszania

Dlaczego działają:

`strona.pl/testowa-strona` i `strona.pl/TeStoWa-StrONa`
oraz

`strona.pl/category/testowa-kategoria` i

`strona.pl/category/tEsTOWA-kAtEgORiA` a nie działają:

`strona.pl/Feed` i `strona.pl/Category?`

Bo:

Rewrite rulesy są dopasowywane za pomocą wyrażeń regularnych, które zwracają uwagę na rozmiar znaków. Jednak bardzo często rewrite tagi nie są specjalnie walidowane i przyjmują także wielkie litery.

Na przykład zmieniamy strukturę URLi kategorii

Czyli mamy:

`strona.pl/category/ciekawe`,

a chcemy mieć:

`strona.pl/blog/ciekawe`

Na przykład zmieniamy strukturę URLi kategorii

Czyli mamy:

`strona.pl/category/ciekawe`,

a chcemy mieć:

`strona.pl/blog/ciekawe`

- Pytamy się na forum lub FB jak to zrobić
- i otrzymujemy odpowiedź, że wystarczy *dodać regułę* `mod_rewrite` do pliku `.htaccess`

Na przykład zmieniamy strukturę URLi kategorii

Czyli mamy:

`strona.pl/category/ciekawe`,

a chcemy mieć:

`strona.pl/blog/ciekawe`

- Pytamy się na forum lub FB jak to zrobić
- i otrzymujemy odpowiedź, że wystarczy *dodać regułę `mod_rewrite` do pliku `.htaccess`*
- Dodajemy. Sprawdzamy, że nowy format URL działa i zadowoleni zamykamy temat

Na przykład zmieniamy strukturę URLi kategorii

Czyli mamy:

`strona.pl/category/ciekawe`,

a chcemy mieć:

`strona.pl/blog/ciekawe`

- Pytamy się na forum lub FB jak to zrobić
- i otrzymujemy odpowiedź, że wystarczy *dodać regułę `mod_rewrite` do pliku `.htaccess`*
- Dodajemy. Sprawdzamy, że nowy format URL działa i zadowoleni zamykamy temat

Ale to nie jest rozwiązanie

Na przykład zmieniamy strukturę URLi kategorii

Czyli mamy:

`strona.pl/category/ciekawe`,

a chcemy mieć:

`strona.pl/blog/ciekawe`

- Pytamy się na forum lub FB jak to zrobić
- i otrzymujemy odpowiedź, że wystarczy *dodać regułę `mod_rewrite` do pliku `.htaccess`*
- Dodajemy. Sprawdzamy, że nowy format URL działa i zadowoleni zamykamy temat

Ale to nie jest rozwiązanie, bo np. funkcja `get_category_link` nadal będzie generować standardowe URLe.



Zamieniamy *page* na *strona*

Problem

Chcemy sprawić, aby paginacja używała słowa *strona* zamiast *page*.

Zamieniamy *page* na *strona*

Problem

Chcemy sprawić, aby paginacja używała słowa *strona* zamiast *page*.

```
function set_my_pagination_base() {
    global $wp_rewrite;
    $wp_rewrite->pagination_base = 'strona';
} add_action( 'wp_loaded', 'set_my_pagination_base' );

function set_default_pagination_base() {
    global $wp_rewrite;
    $wp_rewrite->pagination_base = 'page';
}

function myplugin_activate() {
    set_my_pagination_base();
    flush_rewrite_rules();
} register_activation_hook( __FILE__, 'myplugin_activate' );

function myplugin_deactivate() {
    set_default_pagination_base();
    flush_rewrite_rules();
} register_deactivation_hook( __FILE__, 'myplugin_deactivate' );
```



Koszyk, logowanie, itp.

Problem

Chcemy dodać dynamicznie generowaną stronę, która będzie udawała zwykłą stronę, ale nie będzie można jej edytować.

Koszyk, logowanie, itp.

Problem

Chcemy dodać dynamicznie generowaną stronę, która będzie udawała zwykłą stronę, ale nie będzie można jej edytować.

```
function wpm_insert_rules() {
    global $wp, $wp_rewrite;
    $wp->add_query_var( 'template' );

    $wp_rewrite->add_rule('cart/?$', 'index.php?template=cart', 'top' );
} add_filter('wp_loaded', 'wpm_insert_rules');

function wpm_select_template($template) {
    global $wp_query;

    if ( 'cart' == get_query_var('template') ) {
        $wp_query->set( 'is_404', false );
        return get_stylesheet_directory() . '/cart.php';
    }
    return $template;
} add_action( 'template_include', 'wpm_select_template' );
```


A jeśli chcemy podpiąć coś większego?

Problem

Łączymy własny kod PHP (ale nie tylko) z WordPressem.

A jeśli chcemy podpiąć coś większego?

Problem

Łączymy własny kod PHP (ale nie tylko) z WordPressem.

```
function my_add_external_rewrites() {  
    global $wp_rewrite;  
    $new_non_wp_rules = array(  
        'ogloszenia' => 'core/ogloszenia.php',  
    );  
    $wp_rewrite->non_wp_rules = $new_non_wp_rules + $wp_rewrite->non_wp_rules;  
}
```

A jeśli chcemy podpiąć coś większego?

Problem

Łączymy własny kod PHP (ale nie tylko) z WordPressem.

```
function my_add_external_rewrites() {  
    global $wp_rewrite;  
    $new_non_wp_rules = array(  
        'ogloszenia' => 'core/ogloszenia.php',  
    );  
    $wp_rewrite->non_wp_rules = $new_non_wp_rules + $wp_rewrite->non_wp_rules;  
}
```

A w pliku /core/ogloszenia.php

```
define('COREPATH', dirname(__FILE__));  
define('WP_USE_THEMES', true);  
$wp_did_header = true;  
require_once( '../wp-load.php' );
```

Dziękuję za uwagę!

Krzysiek Dróżdź
krzysiek@wpmagus.pl