WordPress: Using Non-Default URIs

WordPress Meetup Kyiv

Oleg Butuzov, 2018

What this talk about?

Changing default WordPress behaviour
changing URI structure

Handling HTTP request in WordPress

Client making request

```
GET /tag/wp/page/2 HTTP/1.1
Host: example.com
```

DNS telling client where webserver located

```
$ dig a example.com
> ;; ANSWER SECTION:
> example.com.      6244    IN  A   93.184.216.34
```

webserver handles request to host

```
<VirtualHost *:80>
  DocumentRoot "/www/html/example"
  ServerName example.com
</VirtualHost>
```

Running virtual rules on host if file not found

```
<IfModule mod_rewrite.c>
    RewriteRule ^index\.php$ - [L]
    RewriteCond %{REQUEST_FILENAME} !-f
    RewriteCond %{REQUEST_FILENAME} !-d
    RewriteRule . /index.php [L]
</IfModule>
```

WordPress & rewrite_rules

```
$rewrite_rules = [
  '^wp-json/?$' => 'index.php?rest_route=/'
  '^wp-json/(.*)?' => 'index.php?rest_route=/$matches[1]'
  '^index.php/wp-json/?$' => 'index.php?rest_route=/'
  '^index.php/wp-json/(.*)?' => 'index.php?rest_route=/$matches[1]'
  'category/(.+?)/feed/(feed|rdf|rss|rss2|atom)/?$' => 'index.php?category_name=$matches[1]&feed=$matches[2]'
  'category/(.+?)/(feed|rdf|rss|rss2|atom)/?$' => 'index.php?category_name=$matches[1]&feed=$matches[2]'
  'category/(.+?)/embed/?$' => 'index.php?category_name=$matches[1]&embed=true'
  'category/(.+?)/page/?([0-9]{1,})/?$' => 'index.php?category_name=$matches[1]&paged=$matches[2]'
  'category/(.+?)/?$' => 'index.php?category_name=$matches[1]'
  'tag/([^/]+)/feed/(feed|rdf|rss|rss2|atom)/?$' => 'index.php?tag=$matches[1]&feed=$matches[2]'
  'tag/([^/]+)/(feed|rdf|rss|rss2|atom)/?$' => 'index.php?tag=$matches[1]&feed=$matches[2]'
  'tag/([^/]+)/embed/?$' => 'index.php?tag=$matches[1]&embed=true'
  'tag/([^/]+)/page/?([0-9]{1,})/?$' => 'index.php?tag=$matches[1]&paged=$matches[2]'
  'tag/([^/]+)/?$' => 'index.php?tag=$matches[1]'
   // Тут було ще 71 правила розбору URI на складові, але я їх прибрав. Все не влізло.
  'index.php?pagename=$matches[1]&feed=$matches[2]'
  '(.?.+?)/(feed|rdf|rss|rss2|atom)/?$' => 'index.php?pagename=$matches[1]&feed=$matches[2]'
  '(.?.+?)/page/?([0-9]{1,})/?$' => 'index.php?pagename=$matches[1]&paged=$matches[2]'
  '(.?.+?)/comment-page-([0-9]{1,})/?$' => 'index.php?pagename=$matches[1]&cpage=$matches[2]'
  '(.?.+?)(?:/([0-9]+))?/?$' => 'index.php?pagename=$matches[1]&page=$matches[2]'
];
```

URI & RegExp

```
/tag/wp/page/2 => tag/([^/]+)/page/?([0-9]{1,})/?$ => index.php?tag=wp&paged=2
```

Query_String & WP_Query

```
'tag=wp&paged=2' => new WP_Query([ 'tag' => 'wp', 'paged' => 2]);
```

rest of things*

Loading and Displaying

Loading posts using WP_Query

Locating template

Displaying posts in template to client

almost

The End

Important Question #1

What are these rules ?

Important Question #1 — What are these rules ?

[`match_rule` => `match_query`]

Important Question #1 — What are these rules ?

Keys - **Match Rule**

```
tag/([^/]+)/page/?([0-9]{1,})/?$
```

This is **regular expression** what suppose to match to URI and (may be) return some captured matches.

Important Question #1 — What are these rules ?

Value - Match Query

```
index.php?tag=$matches[1]&paged=$matches[2]
```

Pattern string: at palce of $matches[\d{1,}] WP will substitute some captured groups from RegExp Match Rule

Important Question #2 — What are these rules ?

Match result is Query Vars string

https://codex.wordpress.org/WordPress_Query_Vars

```
'tag=wp&paged=2' && [ 'tag' => 'wp', 'paged' => 2 ]
```

Special vars that passed to WP_Query as array. WP use it to get posts to client.

Important Question #2

Why would you change rules?

Important Question #2 — Why would you change rules?

Rewrite Rules are "excessive"

```
$rewrite_rules = [
  '^wp-json/?$' => 'index.php?rest_route=/'
  '^wp-json/(.*)?' => 'index.php?rest_route=/$matches[1]'
  '^index.php/wp-json/?$' => 'index.php?rest_route=/'
  '^index.php/wp-json/(.*)?' => 'index.php?rest_route=/$matches[1]'
  'category/(.+?)/feed/(feed|rdf|rss|rss2|atom)/?$' => 'index.php?category_name=$matches[1]&feed=$matches[2]'
  'category/(.+?)/(feed|rdf|rss|rss2|atom)/?$' => 'index.php?category_name=$matches[1]&feed=$matches[2]'
  'category/(.+?)/embed/?$' => 'index.php?category_name=$matches[1]&embed=true'
  'category/(.+?)/page/?([0-9]{1,})/?$' => 'index.php?category_name=$matches[1]&paged=$matches[2]'
  'category/(.+?)/?$' => 'index.php?category_name=$matches[1]'
  'tag/([^/]+)/feed/(feed|rdf|rss|rss2|atom)/?$' => 'index.php?tag=$matches[1]&feed=$matches[2]'
  'tag/([^/]+)/(feed|rdf|rss|rss2|atom)/?$' => 'index.php?tag=$matches[1]&feed=$matches[2]'
  'tag/([^/]+)/embed/?$' => 'index.php?tag=$matches[1]&embed=true'
  'tag/([^/]+)/page/?([0-9]{1,})/?$' => 'index.php?tag=$matches[1]&paged=$matches[2]'
  'tag/([^/]+)/?$' => 'index.php?tag=$matches[1]'
   // Тут було ще 70 правил розбору URI, але я їх прибрав. Все не влізло.
  '(.?.+?)/feed/(feed|rdf|rss|rss2|atom)/?$' => 'index.php?pagename=$matches[1]&feed=$matches[2]'
  '(.?.+?)/(feed|rdf|rss|rss2|atom)/?$' => 'index.php?pagename=$matches[1]&feed=$matches[2]'
  '(.?.+?)/page/?([0-9]{1,})/?$' => 'index.php?pagename=$matches[1]&paged=$matches[2]'
  '(.?.+?)/comment-page-([0-9]{1,})/?$' => 'index.php?pagename=$matches[1]&cpage=$matches[2]'
  '(.?.+?)(?:/([0-9]+))?/?$' => 'index.php?pagename=$matches[1]&page=$matches[2]'
];
```

Important Question #2 — Why would you change rules?

It's fun! And inspire to do research!

Important Question #2 — Why would you change rules?

Sometimes, it's question of security.

Important Question #2 — Why would you change rules?

Flexible control of WP_Query

Important Question #3

How to chagne rules?

Important Question #3 — How to chagne rules?

add_rewrite_rule

```
add_rewrite_rule( 'filter/(.*)\/?$', 'index.php?filter=$matches[1]', 'top' );
```

- =) Simple usage
- =(Hard to debug

Important Question #3 — How to chagne rules?

"Filters" – best instrument to do work.

```
add_action( 'post_rewrite_rules', function( array $rewrite_rules ) : array {
    $rewrite_rules[ '^api/' ] = 'index.php?do_api_list=1';
    return $rewrite_rules;
} );
```

Using filters on `rewrite_rules`

- =) There are lot of them
- =) You can change rules not only for index.php
- =(Order is important
- =(Require understanding of Regular Expressions
- =(Don't forget to flush rules in case if they contain url entities (e.g. taxonomy terms)

Important Question #4

How to generate Links?

Important Question #4 — How to generate Links?

add_permastruct

```
add_action( 'init', function() {
    add_permastruct( 'tvshow', '/videoteka/%tvshow%.html');
});
```

Але, тут багато "але"...

- =) Простота використання
- =(Націлене на taxonomy та post_type
- =(Лімітоване API
- =(Багато додаткової роботи

Important Question #4 — How to generate Links?

add_rewrite_tag

```
// Adding Rewrite Tag
add_rewrite_tag( '%random%', '(.+)', 'random=' );
// If you declare tag as public it will apear here wp-admin/options-permalinks.php
add_filter( 'available_permalink_structure_tags', function( array $tags ) : array {
    $tags['random'] = 'Some random var';
    return $tags;
});
// Or 'post_link', or any other link filter.
add_filter( 'post_type_link', function( string $permalink ) : string {
    if (false === strpos( $permalink, '%random%' ) ) {
        return $permalink;
    }
    return str_replace( '%random%', rand(1000, 4000), $permalink);
});
```

- Simplicity
- Helps `add_permstruct` be actual

Important Question #4 — How to generate Links?

Using link filters directly

For example `term_link` or `post_type_link`

```
// Some categories can have different locations
add_filter( 'term_link', function( $term_link, $term, $taxonomy ) {
    if ( 'superduper' !== $term->taxonomy ) {
        return $term_link;
    }
    if ( '1' !== get_term_meta( $term->term_id, 'special' ) ) {
        return $term_link;
    }
    return home_url( get_option( 'alternative_base_superduper' ) . '/' . $term->slug );
}, 10, 3);
```

Important Question #4 — How to generate Links?

Manually

¯\_(ツ)_/¯

Cases – shortly about examples

Case #1 — Months Translit

`https://example.ua/blog/2018/april/20`

Want

vs

`https://example.ua/blog/2018/04/20`

Have

Code @ https://github.com/butuzov/WP-Using-Non-Default-URIs

Will help us

- Changes to rewrite_rules - `post_rewrite_rules`, `date_rewrite_rules`
- New links generations - `post_link`, `day_link` `month_link`
- Query String substitutions using ... `query_string`

Case #2 & #3 —Removing "extra" rules

Easy to use `rewrite_rules`when there are no rewrite rules

Code @ https://github.com/butuzov/WP-Using-Non-Default-URIs

Filtering

- Global Rules — `category_rewrite_rules`, `category_rewrite_rules`, `post_format_rewrite_rules`, `post_rewrite_rules`, `date_rewrite_rules`, `root_rewrite_rules`, `search_rewrite_rules`, `page_rewrite_rules`,`tag_rewrite_rules`
- Dynamic rules — `{$post_type}_rewrite_rules`, `{$taxonomy}_rewrite_rules`
- All Rules — `rewrite_rules_array`

Case #4 — Changing default archive functionality

Changing perpage options for different archives

https://github.com/butuzov/WP-Using-Non-Default-URIs

Implementation

- Using filter `parse_query` altering behaviour.
- When can/need - change matched query.
- Don't forget flush rules

Case #5 — widespread categories

Category can have own URL base

Код @ https://github.com/butuzov/WP-Using-Non-Default-URIs

Implementation

- Save meta at `` `term_meta` ``
- Generate separate rules via ${taxonomy}_rewrite_rule
- Generate separate links for terms

Case #6 — Upcoming

Showing `` `future` `` posts

Code @ https://github.com/butuzov/WP-Using-Non-Default-URIs

Implementation

- Shows `` `future` `` posts on separate upcoming `` `upcoming` ``
- Add rule with new `` `query_var` `` - `` `upcoming` ``
- Parsign $wp_query with `upcoming` - will change WP behaviour WP

The End

Thank You!

Addvertisment

Debug Bar Rewrite Rules

https://wordpress.org/plugins/debug-bar-rewrite-rules/

Plugin that helps to work with rewrite rules

- Testing rules argainst your URL
- Search in rules
- What filters used.

Resources

- https://codex.wordpress.org/Query_Overview
- Example: https://github.com/butuzov/WP-Using-Non-Default-URIs

Questions?

made.ua   butuzov@made.ua

butuzov   butuzov   butuzov   butuzov