

Grandes Migrações

Passando de Qualquer Plataforma Para o WordPress

Vinicius Massuchetto

WordCamp Curitiba 2012

Junho de 2012

Download

Esta apresentação está disponível em:
[https://github.com/viniciusmassuchetto/
wordpress-presentation-migrations](https://github.com/viniciusmassuchetto/wordpress-presentation-migrations)

Sobre o Que Falaremos

- Por que tratar do assunto
- Migrações no WordPress
- Migrações em plataformas suportadas
- Migrações de outras plataformas
- Estratégias de migração

Por que falar sobre migrações com o Cliente?

Por que falar sobre migrações com o Cliente?

- Indexação de conteúdo

Por que falar sobre migrações com o Cliente?

- Indexação de conteúdo
- Manutenção de usabilidade

Por que falar sobre migrações com o Cliente?

- Indexação de conteúdo
- Manutenção de usabilidade
- Reestruturação do conteúdo

Por que falar sobre migrações com a equipe de desenvolvimento?

Por que falar sobre migrações com a equipe de desenvolvimento?

- Análise de complexidade

Por que falar sobre migrações com a equipe de desenvolvimento?

- Análise de complexidade
- Análise de correlação e criação de estruturas

Por que falar sobre migrações com a equipe de desenvolvimento?

- Análise de complexidade
- Análise de correlação e criação de estruturas
- Resolução de velhos problemas

Por que falar sobre migrações com a equipe de desenvolvimento?

- Análise de complexidade
- Análise de correlação e criação de estruturas
- Resolução de velhos problemas
- Definição de estratégias

Por que falar sobre migrações com a equipe de desenvolvimento?

- Análise de complexidade
- Análise de correlação e criação de estruturas
- Resolução de velhos problemas
- Definição de estratégias
- Delegação de tarefas

Por que falar sobre migrações com a equipe de desenvolvimento?

- Análise de complexidade
- Análise de correlação e criação de estruturas
- Resolução de velhos problemas
- Definição de estratégias
- Delegação de tarefas
- Elaboração de manuais

Por que falar sobre migrações com a equipe de desenvolvimento?

- Análise de complexidade
- Análise de correlação e criação de estruturas
- Resolução de velhos problemas
- Definição de estratégias
- Delegação de tarefas
- Elaboração de manuais
- Definição do tempo de projeto

E na verdade, as migrações são..

E na verdade, as migrações são..

- Uma etapa de projeto que poderia ser melhor discutida

E na verdade, as migrações são..

- Uma etapa de projeto que poderia ser melhor discutida
- Uma das partes mais importantes da implantação de projetos web

O que são migrações para o WordPress?

O que são migrações para o WordPress?

- WordPress → WordPress

O que são migrações para o WordPress?

- WordPress → WordPress
- Plataformas Suportadas → WordPress

O que são migrações para o WordPress?

- WordPress → WordPress
- Plataformas Suportadas → WordPress
- Outras Plataformas → WordPress

WordPress → WordPress

WordPress → WordPress

- Geralmente tranquila quando feita corretamente

WordPress → WordPress

- Geralmente tranquila quando feita corretamente
- Requer menos processamento e recuperação de conteúdos externos

WordPress → WordPress

- Geralmente tranquila quando feita corretamente
- Requer menos processamento e recuperação de conteúdos externos
- Casos:

WordPress → WordPress

- Geralmente tranquila quando feita corretamente
- Requer menos processamento e recuperação de conteúdos externos
- Casos:
 - Quando a URL não muda

WordPress → WordPress

- Geralmente tranquila quando feita corretamente
- Requer menos processamento e recuperação de conteúdos externos
- Casos:
 - Quando a URL não muda
 - Quando a URL muda

WordPress → WordPress: Mesma URL

WordPress → WordPress: Mesma URL

- Copiar a base

WordPress → WordPress: Mesma URL

- Copiar a base
- Modificar o `wp-config.php`

WordPress → WordPress: Mesma URL

```
mysqldump -u user -ppass db | bzip2 > dump.sql.bz2  
scp dump.sql.bz2 user@host:~/  
ssh user@host "bunzip2 dump.sql.bz2"  
ssh user@host "mysql -u user -ppass db < dump.sql"
```


WordPress → WordPress: URLs Diferentes

WordPress → WordPress: URLs Diferentes

- No WordPress, muitas URLs ficam persistentes no banco de dados

WordPress → WordPress: URLs Diferentes

- No WordPress, muitas URLs ficam persistentes no banco de dados
- Buscar e substituir não resolve

WordPress → WordPress: URLs Diferentes

```
$meta_value = array (  
    "key1" => "test1",  
    "key2" => "test2",  
    "url"   => "http://example.com"  
);  
add_post_meta($post_id, "meta_key", $meta_value);
```

WordPress → WordPress: URLs Diferentes

```
$meta_value = array (  
    "key1" => "test1",  
    "key2" => "test2",  
    "url"  => "http://example.com"  
);  
add_post_meta($post_id, "meta_key", $meta_value);
```

```
a:3:{  
    s:4:"key1";s:5:"test1";  
    s:4:"key2";s:5:"test2";  
    s:3:"url";s:18:"http://example.com";  
}
```

WordPress → WordPress: URLs Diferentes

```
a:3:{  
  s:4:"key1";s:5:"test1";  
  s:4:"key2";s:5:"test2";  
  s:3:"url";s:18:"http://example2.com";  
}
```

WordPress → WordPress: URLs Diferentes

```
a:3:{  
    s:4:"key1";s:5:"test1";  
    s:4:"key2";s:5:"test2";  
    s:3:"url";s:18:"http://example2.com";  
}
```

```
$meta_value = get_post_meta($post_id, "meta_key");
```

WordPress → WordPress: URLs Diferentes

```
a:3:{  
    s:4:"key1";s:5:"test1";  
    s:4:"key2";s:5:"test2";  
    s:3:"url";s:18:"http://example2.com";  
}
```

```
$meta_value = get_post_meta($post_id, "meta_key");
```

```
array(1) {  
    [0]=>  
        bool(false)  
}
```


WordPress → WordPress: URLs Diferentes

- Para mudança de URLs deve-se fazer a substituição adequadamente, via plugin ou script

WordPress → WordPress: URLs Diferentes

- Para mudança de URLs deve-se fazer a substituição adequadamente, via plugin ou script
- Exemplos:
 - Script: Search and Replace Tool
 - Plugin: WordPress Move
 - Plugin: Search and Replace
 - Plugin: WP Migrate Tool

WordPress → WordPress: URLs Diferentes

```
mysqldump -u user -ppass db | bzip2 > dump.sql.bz2  
scp dump.sql.bz2 user@host:~/  
ssh user@host "bunzip2 dump.sql.bz2"  
ssh user@host "mysql -u user -ppass db < dump.sql"
```

e..

WordPress → WordPress: URLs Diferentes

```
TABLES=""  
for T in `mysql -u user -ppass db -N -s -e "SHOW TABLES"`; do  
    TABLES="tables[]=${TABLES}&"  
done
```

WordPress → WordPress: URLs Diferentes

```
TABLES=""  
for T in `mysql -u user -ppass db -N -s -e "SHOW TABLES"``; do  
    TABLES="tables[]=${TABLES}&"  
done  
  
SEARCH="http://example.com"  
REPLACE="http://otherexample.com"
```

WordPress → WordPress: URLs Diferentes

```
TABLES=""  
for T in `mysql -u user -ppass db -N -s -e "SHOW TABLES"``; do  
    TABLES="tables[]=${TABLES}&"  
done
```

```
SEARCH="http://example.com"  
REPLACE="http://otherexample.com"
```

```
ACCESS="user=${USER}&pass=${PASS}&host=${HOST}&data=${DB}"  
DATA="${TABLES}&${ACCESS}&srch=${SEARCH}&rplc=${REPLACE}"
```

WordPress → WordPress: URLs Diferentes

```
TABLES=""  
for T in `mysql -u user -ppass db -N -s -e "SHOW TABLES"``; do  
    TABLES="tables[]=${TABLES}&"  
done
```

```
SEARCH="http://example.com"  
REPLACE="http://otherexample.com"
```

```
ACCESS="user=${USER}&pass=${PASS}&host=${HOST}&data=${DB}"  
DATA="${TABLES}&${ACCESS}&srch=${SEARCH}&rplc=${REPLACE}"
```

```
curl -d "$DATA" http://host/searchreplacedb2.php?step5
```

WordPress → WordPress: URLs Diferentes

```
a:3:{  
  s:4:"key1";s:5:"test1";  
  s:4:"key2";s:5:"test2";  
  s:3:"url";s:23:"http://otherexample.com";  
}
```


WordPress → WordPress: URLs Diferentes

```
a:3:{  
  s:4:"key1";s:5:"test1";  
  s:4:"key2";s:5:"test2";  
  s:3:"url";s:23:"http://otherexample.com";  
}
```

```
$meta_value = get_post_meta($post_id, "meta_key");
```

WordPress → WordPress: URLs Diferentes

```
array(1) {  
    [0]=>  
    array(3) {  
        ["key1"]=>  
        string(5) "test1"  
        ["key2"]=>  
        string(5) "test2"  
        ["url"]=>  
        string(23) "http://otherexample.com"  
    }  
}
```

WordPress → Suportados

WordPress → Suportados

- Blogger
- LiveJournal
- Movable Type
- RSS
- Tumblr

WordPress → Suportados

- Procedimento direto através de arquivos de exportação

WordPress → Suportados

- Procedimento direto através de arquivos de exportação
- Para mudança de URLs deve-se recorrer a métodos similares às migrações WordPress → WordPress.

Tópicos a Serem Levados em Conta

Tópicos a Serem Levados em Conta

- Tecnologia

Tópicos a Serem Levados em Conta

- Tecnologia
- Estrutura

Tópicos a Serem Levados em Conta

- Tecnologia
- Estrutura
- Referências e relações internas

Tópicos a Serem Levados em Conta

- Tecnologia
- Estrutura
- Referências e relações internas
- Conteúdo

Tópicos a Serem Levados em Conta

- Tecnologia
- Estrutura
- Referências e relações internas
- Conteúdo
- Mídias

Tópicos a Serem Levados em Conta

- Tecnologia
- Estrutura
- Referências e relações internas
- Conteúdo
- Mídias
- URLs

Tecnologias Empregadas

Tecnologias Empregadas

- Configuração dos servidores

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)
 - `ini_set("memory_limit", -1)`

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)
 - `ini_set("memory_limit", -1)`
 - `set_time_limit(0)`

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)
 - `ini_set("memory_limit", -1)`
 - `set_time_limit(0)`
 - OU..

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)
 - `ini_set("memory_limit", -1)`
 - `set_time_limit(0)`
 - ou.. ajax recursivo

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)
 - `ini_set("memory_limit", -1)`
 - `set_time_limit(0)`
 - ou.. ajax recursivo
- Sistema de banco de dados da estrutura a ser migrada

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)
 - `ini_set("memory_limit", -1)`
 - `set_time_limit(0)`
 - ou.. ajax recursivo
- Sistema de banco de dados da estrutura a ser migrada
- Modo de obtenção de dados (socket, webservice, csv)

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)
 - `ini_set("memory_limit", -1)`
 - `set_time_limit(0)`
 - ou.. ajax recursivo
- Sistema de banco de dados da estrutura a ser migrada
- Modo de obtenção de dados (socket, webservice, csv)
- Linguagem a serem escritos os scripts de migração

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)
 - `ini_set("memory_limit", -1)`
 - `set_time_limit(0)`
 - ou.. ajax recursivo
- Sistema de banco de dados da estrutura a ser migrada
- Modo de obtenção de dados (socket, webservice, csv)
- Linguagem a serem escritos os scripts de migração
- Preferência:

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)
 - `ini_set("memory_limit", -1)`
 - `set_time_limit(0)`
 - ou.. ajax recursivo
- Sistema de banco de dados da estrutura a ser migrada
- Modo de obtenção de dados (socket, webservice, csv)
- Linguagem a serem escritos os scripts de migração
- Preferência: PHP

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)
 - `ini_set("memory_limit", -1)`
 - `set_time_limit(0)`
 - ou.. ajax recursivo
- Sistema de banco de dados da estrutura a ser migrada
- Modo de obtenção de dados (socket, webservice, csv)
- Linguagem a serem escritos os scripts de migração
- Preferência: PHP, MySQL

Tecnologias Empregadas

- Configuração dos servidores
(safe mode, parâmetros de compilação)
 - `ini_set("memory_limit", -1)`
 - `set_time_limit(0)`
 - ou.. ajax recursivo
- Sistema de banco de dados da estrutura a ser migrada
- Modo de obtenção de dados (socket, webservice, csv)
- Linguagem a serem escritos os scripts de migração
- Preferência: PHP, MySQL, de dentro do WordPress

Tecnologias Empregadas

Tecnologias Empregadas

- Migrar através do próprio WordPress:

Tecnologias Empregadas

- Migrar através do próprio WordPress:
 - Facilidade e padronização de manipulação dos dados

Tecnologias Empregadas

- Migrar através do próprio WordPress:
 - Facilidade e padronização de manipulação dos dados
 - Garantia de integridade

Tecnologias Empregadas: wpdb

Tecnologias Empregadas: wpdb

- Classe wpdb

Tecnologias Empregadas: wpdb

- Classe wpdb
 - query()

Tecnologias Empregadas: wpdb

- Classe wpdb
 - query()
 - get_results()

Tecnologias Empregadas: wpdb

- Classe wpdb
 - query()
 - get_results()
 - get_var()

Tecnologias Empregadas: wpdb

```
global $db;  
$db = new wpdb(  
    IMPORT_USER,  
    IMPORT_PASS,  
    IMPORT_DB,  
    IMPORT_SERVER);
```

Tecnologias Empregadas: Tratamento

Tecnologias Empregadas: Tratamento

- Funções de tratamento:

Tecnologias Empregadas: Tratamento

- Funções de tratamento:
 - `remove_accents()`

Tecnologias Empregadas: Tratamento

- Funções de tratamento:
 - `remove_accents()`
 - `sanitize_title()`

Tecnologias Empregadas: Tratamento

- Funções de tratamento:
 - `remove_accents()`
 - `sanitize_title()`
 - `normalize_whitespace()`

Tecnologias Empregadas: Tratamento

- Funções de tratamento:
 - `remove_accents()`
 - `sanitize_title()`
 - `normalize_whitespace()`
 - `make_clickable()`

Tecnologias Empregadas: Tratamento

- Funções de tratamento:
 - `remove_accents()`
 - `sanitize_title()`
 - `normalize_whitespace()`
 - `make_clickable()`
 - `capital_P_dangit()`

Tecnologias Empregadas: Abstração de Banco

Tecnologias Empregadas: Abstração de Banco

- Funções de relação com o banco de dados:

Tecnologias Empregadas: Abstração de Banco

- Funções de relação com o banco de dados:
 - `wp_insert_post()`

Tecnologias Empregadas: Abstração de Banco

- Funções de relação com o banco de dados:
 - `wp_insert_post()`
 - `wp_insert_term()`

Tecnologias Empregadas: Abstração de Banco

- Funções de relação com o banco de dados:
 - `wp_insert_post()`
 - `wp_insert_term()`
 - `wp_set_post_terms()`

Tecnologias Empregadas: Abstração de Banco

- Funções de relação com o banco de dados:
 - `wp_insert_post()`
 - `wp_insert_term()`
 - `wp_set_post_terms()`
 - `wp_insert_attachment()`

Tecnologias Empregadas: Abstração de Banco

- Funções de relação com o banco de dados:
 - `wp_insert_post()`
 - `wp_insert_term()`
 - `wp_set_post_terms()`
 - `wp_insert_attachment()`
 - `wp_update_attachment_metadata()`

Tecnologias Empregadas: Abstração de Banco

- Funções de relação com o banco de dados:
 - `wp_insert_post()`
 - `wp_insert_term()`
 - `wp_set_post_terms()`
 - `wp_insert_attachment()`
 - `wp_update_attachment_metadata()`
 - `update_post_meta()`

Tecnologias Empregadas: Exemplo de Rotina

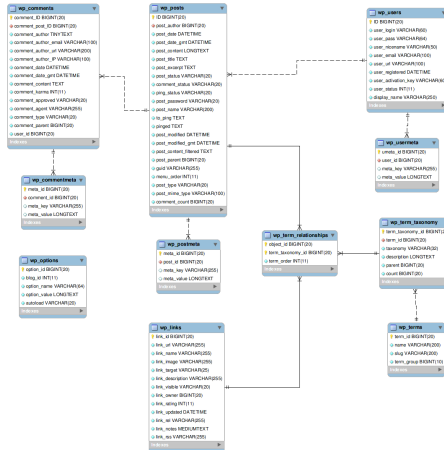
```
$bd = new wpdb(..);  
$sql = "  
    SELECT ...  
    FROM ...  
";  
foreach ($bd->get_results($sql) as $p) {  
  
    /* formatacao de $post_attrs com base em $p */  
  
    wp_insert_post($post_attrs);  
  
}
```

Tecnologias Empregadas: Exemplo de Rotina

```
add_action('wp', 'import');
function import() {
    import_post_id($_REQUEST['post_id']);
    $post_id = $db->get_var($sql);
    ?>
    <script type="text/javascript">
        jQuery.ajax({ /* .. */
            data: {
                'ajax': 1,
                'post_id': <?php echo $post_id; ?>
            }
        });
    </script>
    <?php
}
```

Estrutura

Estrutura



Estrutura: Mapeamento

Estrutura: Mapeamento

- Identificação das entidades

Estrutura: Mapeamento

- Identificação das entidades (post types)

Estrutura: Mapeamento

- Identificação das entidades (post types)
- Identificação dos atributos

Estrutura: Mapeamento

- Identificação das entidades (post types)
- Identificação dos atributos (custom fields)

Estrutura: Mapeamento

- Identificação das entidades (post types)
- Identificação dos atributos (custom fields)
- Identificação das relações

Estrutura: Mapeamento

- Identificação das entidades (post types)
- Identificação dos atributos (custom fields)
- Identificação das relações (taxonomias ou relações de plugins)

Estrutura: Mapeamento

Exemplo: Linha de tabela estabelecimento

- (int) id
- (str) nome
- (str) descricao
- (str) rua
- (str) bairro
- (fk) cidade
- (fk) estado
- (fk) tipo
- (fk) foto

Estrutura: Mapeamento

Estrutura: Mapeamento

- Post type estabelecimento
 - `post_title:` nome
 - `post_content:` descricao

Estrutura: Mapeamento

- Post type estabelecimento
 - `post_title:` nome
 - `post_content:` descricao
- Chaves meta
 - `meta_key:` rua
 - `meta_key:` bairro
 - `meta_key:` _id

Estrutura: Mapeamento

- Post type estabelecimento
 - `post_title:` nome
 - `post_content:` descricao
- Chaves meta
 - `meta_key:` rua
 - `meta_key:` bairro
 - `meta_key:` _id
- Taxonomia
 - Tipo de Estabelecimento

Estrutura: Mapeamento

- Post type estabelecimento
 - `post_title:` nome
 - `post_content:` descricao
- Chaves meta
 - `meta_key:` rua
 - `meta_key:` bairro
 - `meta_key:` _id
- Taxonomia
 - Tipo de Estabelecimento
- Anexo
 - Foto

Estrutura: Mapeamento

- cidade (??)
- estado (??)

Estrutura: Mapeamento

- cidade (??)
- estado (??)

Soluções possíveis:

- Taxonomia
- `post_parent`
- Relação via custom fields
- Relação via plugin
(Advanced Custom Fields, Post Relationships)

Relações Internas

Relações Internas

- Solução depende do escopo de projeto

Relações Internas

- Solução depende do escopo de projeto
- Algumas transferências devem ser em camadas (relações recursivas)

Relações Internas

- Solução depende do escopo de projeto
- Algumas transferências devem ser em camadas (relações recursivas)
- Exemplos:

Relações Internas

- Solução depende do escopo de projeto
- Algumas transferências devem ser em camadas (relações recursivas)
- Exemplos:
 - Taxonomia “Tipo de Estabelecimento”

Relações Internas

- Solução depende do escopo de projeto
- Algumas transferências devem ser em camadas (relações recursivas)
- Exemplos:
 - Taxonomia “Tipo de Estabelecimento”
 - Itens “Leia também” de um post

Relações Internas: Inserção sob Demanda

```
require_once(ABSPATH . 'wp-admin/includes/taxonomy.php');

function get_or_create_category($cat_name) {

    if ($cat_id = get_cat_ID($cat_name))
        return $cat_id;

    $cat_id = wp_create_category($cat_name);
    return $cat_id;
}
```

Relações Internas: Primeira camada de migração

```
foreach ($posts as $p) {  
    /* Post ja importado */  
    if ($id = get_new_id($p->id))  
        continue;  
  
    /* Insere post sem nenhum conteudo */  
    $post_attrs = get_new_post_attrs($p);  
    $post_id = wp_insert_post($post_attrs);  
    update_post_meta($post_id, '_old_id', $p->id);  
}
```

Formatação: Basicamente o conteúdo

- Elaboração de inventário de conteúdo
- Conversão de caminhos
- Conversão de embeds em shortcodes
- Conversão de estruturas HTML editáveis
- Remoção de scripts

Formatação: Capturando Links

```
preg_match_all('/< *a [^>]*href *= *["\']?([^\\"'\']*)/i',  
    $content, $links);  
foreach ($links[1] as $link) {  
    $page_content = file_get_contents($link);  
    $page_title = preg_match('/<title>(.*?)</title>/',  
        $page_content, $title);  
  
    /* ... verificacoes ... */  
  
    $linkdata = array(  
        'link_name' => $title[1],  
        'link_url' => $link  
    );  
    wp_insert_link($linkdata);  
}
```

Migração de Mídias

Migração de Mídias

- Não se contentar com as mídias inseridas no conteúdo

Migração de Mídias

- Não se contentar com as mídias inseridas no conteúdo
 - Sem rastreabilidade pelo software

Migração de Mídias

- Não se contentar com as mídias inseridas no conteúdo
 - Sem rastreabilidade pelo software
 - Sujeitas à estrutura própria de armazenamento

Migração de Mídias

- Não se contentar com as mídias inseridas no conteúdo
 - Sem rastreabilidade pelo software
 - Sujeitas à estrutura própria de armazenamento
- Montar também a biblioteca de mídias e galerias do conteúdo

Migração de Mídias

```
preg_match_all('/<img[^>]*src *= *["\']?([^"\']*)/i',  
    $post_content, $media);  
foreach ($media[1] as $img_url) {  
    /* (2) */  
}
```

Migração de Mídias

```
if (!$filename = valid_file($img_url))
    continue;

$upload = wp_upload_dir(/* 'yyyy/mm' */);
download($m, $path = $upload['path'] . '/' . $filename);
$wp_filetype = wp_check_filetype($file);
$attachment = array(
    'guid'           => $img_url,
    'post_mime_type' => $wp_filetype['type'],
    'post_title'     => $filename,
    'post_status'    => 'inherit',
);
```


Migração de Mídias

```
$attach_id = wp_insert_attachment($attachment, $path, $post_id);  
$attach_data = wp_generate_attachment_metadata($attach_id,  
    $path);  
wp_update_attachment_metadata($attach_id, $attach_data);
```

Migração de Mídias: Serviço Dinâmico

- Ao invés do download pode-se fazer o serviço dinâmico de mídias. Veja o arquivo `wp-includes/ms-files.php`

Migração de Mídias: Serviço Dinâmico

```
http://exemplo.com/imagens/foto.jpg
```

```
RewriteRule $^$(*.*)?imagens/(.*)  
    wp-includes/ms-files.php?file=$2 [L]
```

```
http://exemplo.com/wp-includes/ms-files.php?file=foto.jpg
```

Migração de Mídias: Serviço Dinâmico

```
$file = $_GET['file']  
  
header( 'Content-Type: ' . $mimetype );  
header( 'Content-Length: ' . filesize( $file ) );  
header( 'X-Accel-Redirect: ' . str_replace( WP_CONTENT_DIR, '',  
    $file ) );  
header( 'X-Sendfile: ' . $file );  
readfile( $file );
```

URLs

- Para cada conteúdo migrado, sempre guardar a referência para o conteúdo antigo

URLs

- Para cada conteúdo migrado, sempre guardar a referência para o conteúdo antigo
- Na nova estrutura, verificar os meios de acesso no conteúdo e redirecionar para o novo.

URLs

- Para cada conteúdo migrado, sempre guardar a referência para o conteúdo antigo
- Na nova estrutura, verificar os meios de acesso no conteúdo e redirecionar para o novo.
- Casos:
 - A partir de referências \$_REQUEST
 - A partir de regras de URL

URLs: Referências \$_REQUEST

```
add_action('wp', 'check_migrated');  
function check_migrated() {  
  
    if (!isset($_REQUEST['old_website_id']))  
        return false;  
  
    if ($new_post =  
        get_post_from_old_id($_REQUEST['old_website_id']))  
        wp_redirect(get_permalink($new_post->ID));  
  
    return false;  
  
}
```


URLs: Referências a partir de regras

```
/* URL a ser migrada */  
  
http://example.com/old_id/123  
  
/* Escolher uma variavel para identificar este padrao */  
  
add_action('query_vars', 'add_query_vars');  
function add_query_vars($vars) {  
    array_push($vars, 'old_id');  
    return $vars;  
}
```

URLs: Referências a partir de regras

```
/* Adiciona a regra no set de regras */  
  
add_action('rewrite_rules_array', 'rewrite_rules');  
function rewrite_rules($rules) {  
    $new_rules = array(  
        'old_id/(.+?)/?$'  
    );  
    return $rules + $new_rules;  
}
```

URLs: Referências a partir de regras

```
/* Configura o flush de URLs */

add_action('wp_loaded', 'flush_rules');
function flush_rules() {
    if (!isset($rules['old_id/(.+?)/?$'])) {
        global $wp_rewrite;
        $wp_rewrite->flush_rules();
    }
}
```

Considerações Finais

- Migrações devem definitivamente ser incluídas como componentes de projeto
- O WordPress oferece um bom conjunto de ferramentas para se trazer dados para dentro de sua estrutura.
- É cabível o desenvolvimento de plugins específicos para a migração de diferentes estruturas

Referências

- Codex: Moving WordPress
http://codex.wordpress.org/Moving_WordPress
- GitHub / interconnectit / Search-Replace-DB
<https://github.com/interconnectit/Search-Replace-DB>