Wordpress:自定义各个类别每页显示文章数

在Wordpress后台的设置->阅读->博客页面至多显示
里面可以设置每页最多显示的文章数目，但是那个设置只能将所有的类别（首页、分类目录页、标签页、作者页）显示的文章数都设置成一个值。

但是在开发Wordpress主题的时候，有些需求需要修改不同类别的每页显示的文章数。比如首页显示10篇；分类页显示20篇；标签页显示30篇。这种情况下我们就得用代码来实现（虽然有插件可以满足这类需求，但是我们知道，插件装多了并不好。）。这里介绍一种简单的方法来达到这种需求。

首先，我们得知道Wordpress中内部提供了许多函数来判断某个页面的类别：

```
is_home()
is_paged()
is_single()
is_category()
is_tag()
```

上面的函数依次代表判断是否是首页、是否分页显示、是否是文章页、是否是分类目录页、是否是标签页。而且Wordpress中提供了各种钩子（Hooks ,这是什么意思自己去Wordpress网站上看吧），通过这些钩子我们可以修改Wordpress中的一些内部行为。其中我们知道的就是在查询的时候可以通过posts_per_page设置每页显示的文章数：

```
$args = array(
   'caller_get_posts' => 1
   'posts_per_page' => 10
);
query_posts($args);
```

上面的代码片段说明我这次查询需要每页最多显示10篇。好了，知道这些之后。我们可以通过重写pre_get_posts钩子来自定义不同类别每页显示的文章数。代码片段如下：

```
//////////////////////////////////////////////////////////////////////////////
 User: 过往记忆
 Date: 14-11-30
 Time: 03:59
 bolg:
 本文地址：/archives/1211
```

过往记忆博客，专注于hadoop、hive、spark、shark、flume的技术博客，大量的干货
过往记忆博客微信公共帐号：iteblog_hadoop

```
//////////////////////////////////////////////////////////////////////
function iteblog_posts_per_page($query) {
    if (is_home()) {
        $query->set('posts_per_page', 10);
    }
    if (is_category()) {
        $query->set('posts_per_page', 20);
    }
    if (is_tag()) {
        $query->set('posts_per_page', 30);
    } //endif
} //function

add_action('pre_get_posts', 'iteblog_posts_per_page');
```

这样就将首页显示10篇，分类页显示20篇，标签页显示30篇。可以将这个函数加到Wordpress当前使用主题的functions.php文件中。

上面的代码在正常情况下是可以运行正常的。但是有些情况下会失效，什么意思？有时候is_home()函数（其他函数类似）会误判当前类别页面是首页。比如如果你首页有如下查询语句

```
$args = array(
    'caller_get_posts' => 1,
    'paged' => $paged,
    'posts_per_page' => $total_show,
    'cat' => $show_first_id
 );
query_posts($args);
```

注意看，在查询的时候加入了'cat' => $show_first_id，这时候你在首页使用is_home()会失效，但是用is_category()却成功！这是什么情况？我们先看看is_home()函数在Wordpress内部是怎么实现的。

```
//////////////////////////////////////////////////////////////////////
 User: 过往记忆
 Date: 14-11-30
 Time: 03:59
 bolg:
```

本文地址：/archives/1211
过往记忆博客，专注于hadoop、hive、spark、shark、flume的技术博客，大量的干货
过往记忆博客微信公共帐号：iteblog_hadoop

```
//////////////////////////////////////////////////////////////////////
function is_home() {
 global $wp_query;

 if ( ! isset( $wp_query ) ) {
  _doing_it_wrong( __FUNCTION__,
     __( 'Conditional query tags do not work before
    the query is run. Before then, they always return false.' ), '3.1' );
  return false;
 }

 return $wp_query->is_home();
}
```

也就是通过返回$wp_query->is_home();来设置。而如果你在查询条件加入了'cat'条件，所以$wp_query->is_home()就变成false了！

那怎么办？我这里提供一种简单的方法来实现（如果你有更好的办法可以留言哦）。我们可以在查询条件中加入自定义的查询条件，如下：

```
$args = array(
    'caller_get_posts' => 1,
    'paged' => $paged,
    'posts_per_page' => $total_show,
    'cat' => $show_first_id,
    'is_home' => true
 );
query_posts($args);
```

当然在query_posts中无法使用，而且也不会报错。但是我们可以通过$query->get('is_home');来获取$args中的值！如下：

```
$is_home = $query->get('is_home');
if($is_home){
    echo '/ home page!!';
}
```

上面的代码会输出/ home page!!。得到这个启发，我们可以将上面自定义的iteblog_posts_per_page函数写成：

```
//////////////////////////////////////////////////////////////////////////
 User: 过往记忆
 Date: 14-11-30
 Time: 03:59
 bolg:
 本文地址：/archives/1211
 过往记忆博客，专注于hadoop、hive、spark、shark、flume的技术博客，大量的干货
 过往记忆博客微信公共帐号：iteblog_hadoop
//////////////////////////////////////////////////////////////////////////
function iteblog_posts_per_page($query) {
   $is_home = $query->get('is_home');
    if($is_home){
        $query->set('posts_per_page', 10);
    }
   if(empty($is_home)){
      if (is_category()) {
         $query->set('posts_per_page', 20);
      }
      if (is_tag()) {
         $query->set('posts_per_page', 30);
      } //endif
   }
} //function
add_action('pre_get_posts', 'iteblog_posts_per_page');
```

这样就可以解决问题了！！

本博客文章除特别声明，全部都是原创！
原创文章版权归过往记忆大数据（过往记忆）所有，未经许可不得转载。
本文链接: 【】（）