

Magickal MU Plugins



Mikel King

*If you don't use what you make then you failed to
make something useful*

Twitter:

@MikelKing

GitHub:

<https://github.com/mikelking>

LinkedIn:

<https://www.linkedin.com/in/mikelking>

x1:

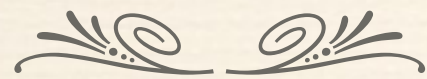
mikelking

1. It frames this as I am the authority on the subject and I believe that there are many different avenues

2. I don't feel like the quality answers are achievable in such a short period

3. I prefer intelligent conversations to volley ball Q & A

Photo of me



circa 1993 while in the USCG



Aspects of this presentation are available on GitHub

- ❖ <https://github.com/mikelking/magickal-mu-plugins>
- ❖ https://github.com/mikelking/singleton_base

Some assumptions

- ❖ That the majority of you in this room are developers.
- ❖ That you have an understanding about what a MU plugin is.
- ❖ That you have at least dabbled in writing your own plugin.

- ❖ First how many of you actively code in PHP every day?
- ❖ How many practice OOP?
- ❖ How many have heard of codesniffing?
- ❖ How many have heard of unit testing?
- ❖ How many have heard of DRY?
- ❖ How many have heard of design patterns?
- ❖ How many people have sites that have hundreds even thousands of lines of code in their functions.php file?

Over 9000 lines of code

If it needs to be on the site
then put it in a plugin!!!

Problem solved
MU Plugins

No quite

MU Plugins like the functions.php

- ❖ Can NOT be turned OFF
- ❖ Are active for ALL sites in a MultiSite
- ❖ Are pervasive across your entire WordPress Install

Feels like a huge bummer

**Solution is building blocks
like legos**

This library of blocks

- ❖ Relies heavily on OOP
- ❖ Especially on the principles of inheritance

Inheritance in people



Inheritance in people

```
<?php
```

```
class Father extends Grand_Father {  
    const FIRST_NAME    = 'Mikel';  
    const LAST_NAME     = 'King';  
    const DEFAULT_TZ    = 'America/New_York';  
    const STATUS        = 'ALIVE';  
    const EYE_COLOUR    = 'Hazel';  
    const ALLERGIES     = 'Chocolate';  
}
```

```
<?php
```

```
class Son Extends Father {  
    const FIRST_NAME    = 'Jarett';  
    const LAST_NAME     = 'King';  
    const DEFAULT_TZ    = 'America/New_York';  
    const STATUS        = 'ALIVE';  
}
```


Execution

```
$mikel = Father::get_instance();  
$mikel->print_stats();  
print( '=====' . PHP_EOL );  
  
$jk = Son::get_instance();  
$jk->print_stats();  
print( '=====' . PHP_EOL );
```

Class name: Father
Name: Mikel King
Hair colour: Dark Brown
Eye colour: Hazel
Status: ALIVE
Known allergies: Chocolate

Class name: Son
Name: Jarett King
Hair colour: Dark Brown
Eye colour: Hazel
Status: ALIVE
Known allergies: Chocolate

Inheritance in people



Inheritance in people

```
<?php
```

```
class Grand_Father extends Great_Grand_Mother {  
    const FIRST_NAME = 'James';  
    const LAST_NAME  = 'King';  
    const DEFAULT_TZ = 'America/New_York';  
    const STATUS      = 'ALIVE';  
}
```

```
<?php
```

```
class Father extends Grand_Father {  
    const FIRST_NAME = 'Mikel';  
    const LAST_NAME  = 'King';  
    const DEFAULT_TZ = 'America/New_York';  
    const STATUS      = 'ALIVE';  
    const EYE_COLOUR  = 'Hazel';  
    const ALLERGIES    = 'Chocolate';  
}
```


Inheritance in people

```
<?php
```

```
class Great_Grand_Mother extends Great_Great_Grand_Father {  
    const FIRST_NAME = 'Mary';  
    const LAST_NAME  = 'Luurtsema';  
    const DEFAULT_TZ = 'America/New_York';  
    const STATUS     = 'ALIVE';  
    const ALLERGIES  = 'None';  
}
```

```
<?php
```

```
class Great_Great_Uncle extends Great_Great_Grand_Father {  
    const FIRST_NAME = 'James';  
    const LAST_NAME  = 'Johnson';  
    const DEFAULT_TZ = 'America/New_York';  
    const STATUS     = 'ALIVE';  
    const ALLERGIES  = 'Chocolate';  
}
```



```
<?php
```

```
class Great_Great_Grand_Father {  
    const FIRST_NAME    = 'Howard';  
    const LAST_NAME     = 'Johnson';  
    const DEFAULT_TZ    = 'America/Los_Angeles';  
    const STATUS        = 'Deceased';  
    const HAIR_COLOUR   = 'Dark Brown';  
    const EYE_COLOUR    = 'Dark Brown';  
    const ALLERGIES      = 'Chocolate';  
  
    public static $instance;  
  
    public static $class_name;
```



```
public function print_stats() {  
    print( 'Class name: ' . get_class( static::$instance ) . PHP_EOL );  
    print( 'Name: ' . static::FIRST_NAME . ' ' . static::LAST_NAME .  
PHP_EOL );  
    print( 'Hair colour: ' . static::HAIR_COLOUR . PHP_EOL );  
    print( 'Eye colour: ' . static::EYE_COLOUR . PHP_EOL );  
  
    print( 'Status: ' . static::STATUS . PHP_EOL );  
  
    $this->print_allergies();  
}  
  
public function print_allergies() {  
    print( 'Known allergies: ' . static::ALLERGIES . PHP_EOL );  
}
```



```
public static function get_instance() {  
    $new_instance = new static();  
    self::set_tz();  
  
    if ( self::$instance === null || ( self::$instance != $new_instance ) ) {  
        self::$instance = $new_instance;  
    }  
  
    return( self::$instance );  
}  
  
public static function set_tz() {  
    date_default_timezone_set( static::DEFAULT_TZ );  
}  
}
```


Unfortunately People != Programs

- ❖ People are the result of binary inheritance
- ❖ Classes in PHP are unary
- ❖ This example is a single tree

- ❖ All of these classes extend the Great-Great-Grandfather class
- ❖ Great-Great-Grandfather is the root of the tree
- ❖ Great-Great-Grandfather is based upon a simple singleton design pattern
- ❖ These descendant classes work because of late static binding

So who wants to look at
some real code?

Twitter:

@MikelKing

GitHub:

<https://github.com/mikelking>

LinkedIn:

<https://www.linkedin.com/in/mikelking>

x1:

mikelking



40
3305



315



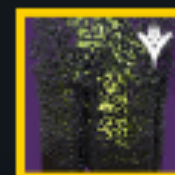
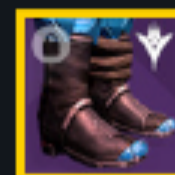
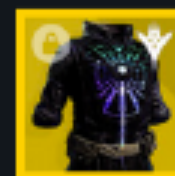
297



241



89



Magickal MU Plugins



Mikel King

*If you don't use what you make then you failed to
make something useful*