

ADMINISTRA WORDPRESS AUTOMÁTICAMENTE CON WP-CLI

WordCamp Málaga 2013

Mike Schroder | @GetSource
Developer, WordPress Specialist con DreamHost

Mike Schroder, alias DH-Shredder, alias @GetSource

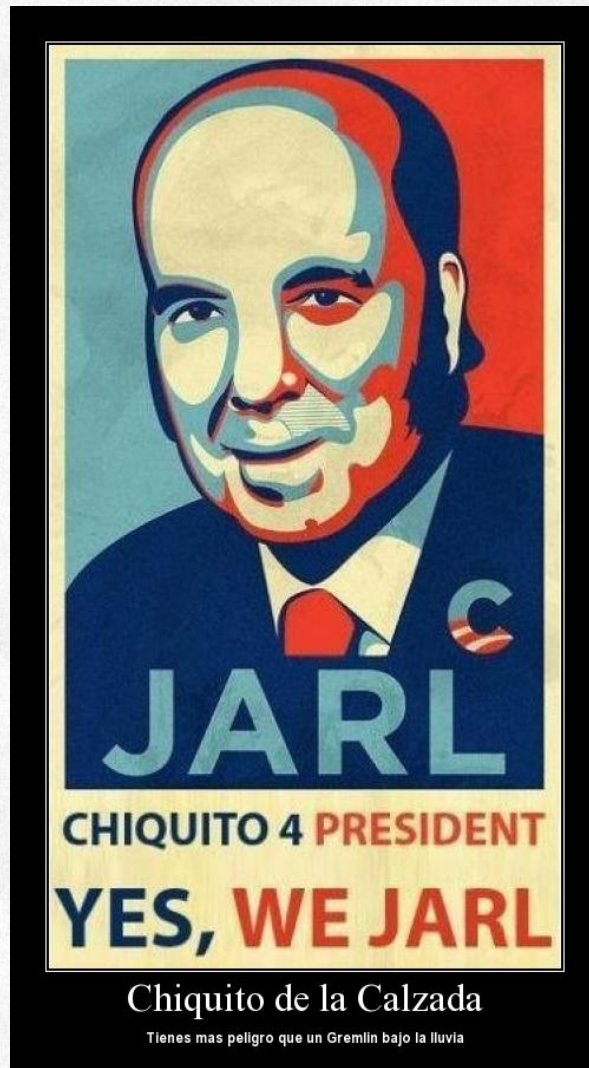


CREDENTIALS

Empleado Feliz de DreamHost
Contribuador a WordPress Core
WordPress 3.5 Recent Rockstar
Contribuidor a WP-CLI

RANDOM FACTS

Multicultural
Encanta Café y Sailing
Bloga en getsource.net



HAY DOS GRUPOS DE GENTE

LOS QUE **UTILIZAN** LA LÍNEA DE COMANDOS

LOS QUE VAN A UTILIZAR LA LÍNEA DE COMANDOS

NO TENGAS MIEDO DE LA CLI

ES TU AMIGO.

TE GUSTA LA CLI?

WP-CLI TE MEJORA LA VIDA

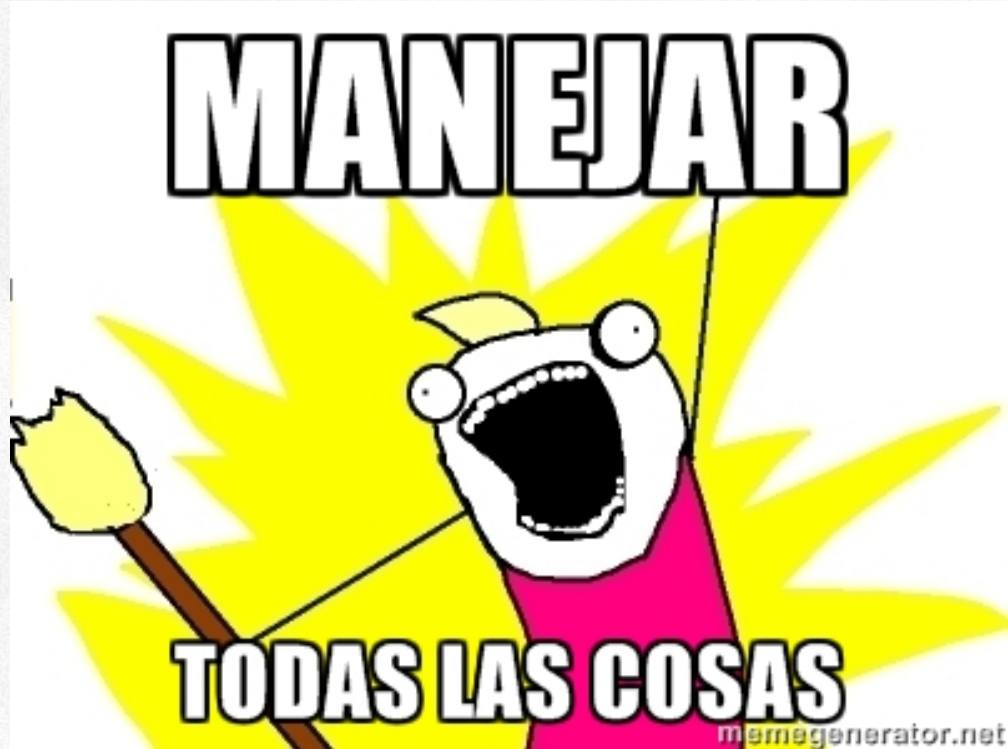
SUPER-GUAY SOFTWARE LIBRE **PARA MANEJAR WORDPRESS**

**MANEJADO POR CRISTI BURCĂ (SCRIBU)
Y ANDREAS CRETEN**

UTILIZA **WORDPRESS** SÍ MISMO PARA OPERARSE

¡AUTOMATIZATE!

¿QUE PUEDO HACER?



EN SERIO.

ACTUALIZA WORDPRESS

```
$ wp core update
```


INSTALLA UN PLUGIN

```
$ wp plugin install <slug o .zip>
```

ELIGE UN TEMA

```
$ wp theme activate <slug>
```


GUARDA LA BASE DE DATOS

```
$ wp db export [ficha]
```

ACTUALIZA LOS PLUGINS

```
$ wp plugin update-all
```


ACTUALIZA LOS TEMAS

```
$ wp theme update-all
```

SUSTITUIR TEXTO EN DB

```
$ wp search-replace <viejo> <nuevo>
```


¿QUÉ ES ÉSTA MÁGIA?



CONVENCIDO.
¿CÓMO PUEDO OBTENERLO?

¿QUÉ REQUIERE PARA USARLO?

- Acceso SSH a tu directorio WordPress
- PHP 5.3.2+
- WordPress 3.4+
- Tanta RAM para que corren los procesos shell de WordPress
- Linux o MacOS (por ahora)

INSTALA COMO USADOR LOCAL

Instrucciones Completas:

wp-cli.org

```
$ curl http://wp-cli.org/installer.sh | bash
```

¡ESPERA; HAY MÁS!

¡WP-CLI ES EXTENSÍBLE!

AÑÁDELO LOCALMENTE

Ejecutar/Alias:

```
$ wp --require='cmd_name.php'
```

wp-cli.yml

```
require: /path/to/cmd_name.php
```


Ó EN LOS PLUGINS

```
if ( defined('WP_CLI') && WP_CLI ) {  
    include( PLUGIN_DIR . '/lib/wp-cli.php' );  
}
```

COMANDO PERSONALIZADO: BACKUP

```
$ wp migrate backup [ficha.tar.gz]  
[--no-db] [--db-name=<ficha.sql>]
```


COMANDO PERSONALIZADO: BACKUP

Strategía

- Utiliza el comando internal para guardar la base de datos
- Crea un .tar.gz combinando las fichas del instalación y la base de datos

AÑADA COMMANDO

```
<?php
WP_CLI::add_command( 'migrate', 'DH_Migrate_Command' );

/**
 * DreamHost Migrate Plugin
 *
 * @package DH_Migrate_Command
 * @subpackage commands/community
 * @maintainer Mike Schroder
 */
class DH_Migrate_Command extends WP_CLI_Command {

// ...
```


DEFINE EL SUBCOMMANDO

```
/**
 * Guarda WordPress entero, incluyendo core, plugins y la base de datos.
 *
 * ## OPTIONS
 *
 * <backup-filename>
 * : Fichero donde guardar la copia de seguridad
 *
 * [--no-db]
 * : No crea copia de seguridad de la base de datos
 *   dentro del archivo creado
 *
 * [--db-name=<filename>]
 * : Guarda el base de datos en el archivo con el nombre <filename>
 *
 * ...
```

DEFINE EL SUBCOMMANDO

\$args

Argumentos Independientes

\$assoc_args

--arg=value Argumentos Asociativos

```
* ...  
*/  
function backup( $args, $assoc_args ) {  
  
// ...
```



```
function backup( $args, $assoc_args ) {
    $filename = $dbname = null;
    $backup_directory = '../';

    // Si no especifica un nombre para el fichero,
    // usa "Site's Title.tar.gz"
    if ( empty( $args ) )
        $filename = $backup_directory . get_bloginfo() . '.tar.gz';
    else
        $filename = $args[0];

    // ...
}
```

```
// Si especifica --no-db, no guarda la base de datos.
if ( ! isset( $assoc_args['no-db'] ) ) {

    $dbname = isset( $assoc_args['db-name'] ) ?
        $assoc_args['db-name'] : 'database_backup.sql';

    WP_CLI::run_command(
        array( 'db', 'export', $backup_directory . $dbname ), array() );
}

// ...
```

```
// Utilizamos esc_cmd para filtrar los argumentos automaticamente.
// No podemos utilizar --exclude-vcs, porque no funciona en OSX.
WP_CLI::line( "Guardano en $filename ..." );
$result = WP_CLI::launch( \WP_CLI\Utils\esc_cmd( "
    tar
        --exclude '.git'
        --exclude '.svn'
        --exclude '.hg'
        --exclude '.bzip'
        -czf %s . -C %s %s
", $filename, $backup_directory, $dbname ), false );

// ...
```



```
// Si creimos una copia de la base de datos, borra la copia temporal.
if ( $dbname && ! unlink( $backup_directory . $dbname ) )
    WP_CLI::warning(
        "No pude borrar la base de datos temporal, '$dbname'." );

if ( 0 == $result ) {
    WP_CLI::success( "Copia de seguridad creado." );
} else {
    WP_CLI::error( "Creación de la copia de seguridad ha fallado." );
}

} // Termina backup

} // Termina DH_Migrate_Command
```

APRENDE MÁS:

WP-CLI:

- wp-cli.org
- getsource.net/tag/wp-cli/
- halfelf.org/2012/command-line-wp

Chuleta de CLI:

- files.fosswire.com/2007/08/fwunixref.pdf