



INTRODUCING WORDPRESS MULTITENANCY



@cliffseal



THE BIG IDEA

Multiple independent instances of WordPress running off one, non-hacked core directory (and possibly sharing themes and plugins, too)

We use this stuff for



evermore

(evermoresites.com)



WHY MULTITENANCY?

Here's what multitenancy can offer you over individual installations:

- Smaller footprint
- Faster code deployment
- Easier updating process all around
- Use single instances of themes & plugins

***NOT* MULTISITE**

Multiple databases
Less plugin hiccups



WHAT'S NEW WITH MULTITENANCY?

- **Symlinked plugins:** Smart people have been talking about multitenancy (and doing it) for years. Some best practices have emerged, while WordPress 3.9 included this critical final step.
- **GitHub mirrors:** things like WordPress core being mirrored on Github have made the whole process easier.



HOW THE BIG IDEA WORKS

(For Evermore, At Least)



HOW IS EACH SITE STRUCTURED?

index.php

wp-config.php

wp => ../wp

content

uploads

mu-plugins => ../repo/mu-plugins

plugins => ../repo/plugins

themes => ../repo/themes



WHAT'S IN INDEX.PHP?

```
// WordPress view bootstrapper  
define( 'WP_USE_THEMES', true );  
require( './wp/wp-blog-header.php' );
```



WHAT'S IN WP-CONFIG.PHP?

```
define( 'WP_HOME', 'https://' .  
basename( __DIR__ ) );  
  
define( 'WP_SITEURL', 'https://' .  
basename( __DIR__ ) . '/wp' );  
  
define( 'WP_CONTENT_DIR',  
dirname( __FILE__ ) . '/content' );  
  
define( 'WP_CONTENT_URL', 'https://' .  
$_SERVER['HTTP_HOST'] . '/content' );
```



WHAT'S IN THE CONTENT DIRECTORY?

The `/uploads` directory stays as-is—that always needs to be independent.

The other symlinks (**mu-plugins**, **plugins**, **themes**) point to single directories on the server containing one copy of themes & plugins available on the sites. These are source-controlled, so we can test updates locally and then deploy them to each server.



WHAT'S IN THE WP DIRECTORY?

You need a standard WordPress core directory at the other end of your symlink. Simply add in your own **wp-config.php**:

```
include_once( $_SERVER[ 'DOCUMENT_ROOT' ] .  
' /wp-config.php' );
```

This file never gets updated either in a standard core update or in a Git checkout.



WHAT'S IN THE WP DIRECTORY? (CONT.)

If you use the GitHub mirror as a remote, you can wait on the tag to come through and run:

```
cd /my/wp/directory  
git fetch --tags  
git checkout -f tags/4.3.1
```

...and now everyone is updated. Alternatively, you can push to a remote that checks itself out (like you might do with the content dir). You can roll back versions easily, too.

1.COMBINE STEPS

2.WRITE SCRIPT

3.?????

4.PROFIT!!!

4 POINT PLAN:

1. Start Up

2. Cash In

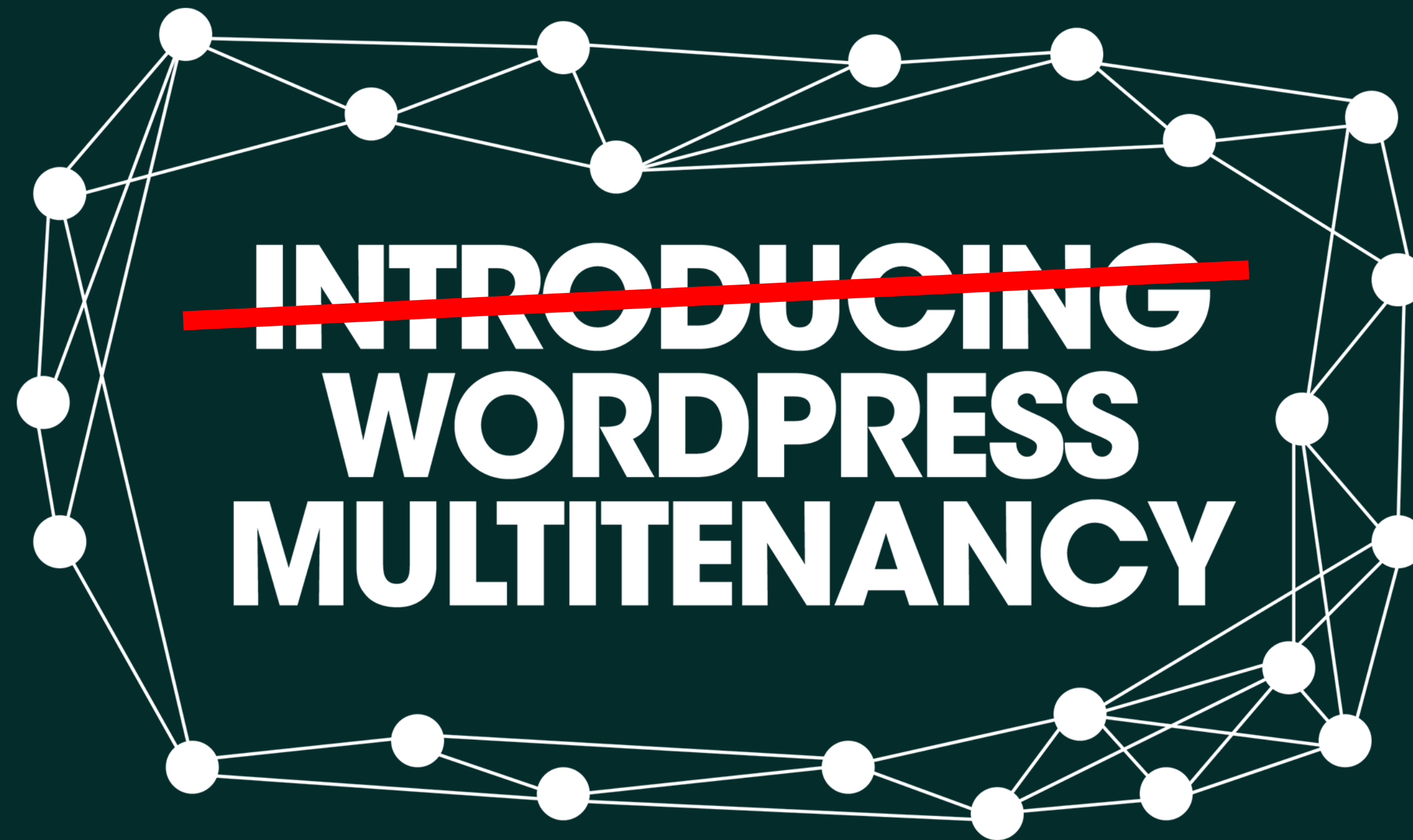
3. Sell Out

4. Bro Down



WHAT SHOULD WE LOOK OUT FOR?

- Some plugins (wrongly) **assume a more standard WordPress structure**. This can break things. Patch their plugin and submit it to them, because they will (probably) not care.
- **Use subdirectories inside /plugins** when symlinking, as single file plugins and individual mu-plugins can have trouble.
- **Test deployment processes** for multiple production servers.



~~INTRODUCING~~ WORDPRESS MULTITENANCY



@cliffseal