WordCamp Netherlands 2026

Multisite for Enterprise – From Concept to Global WordPress Infrastructure

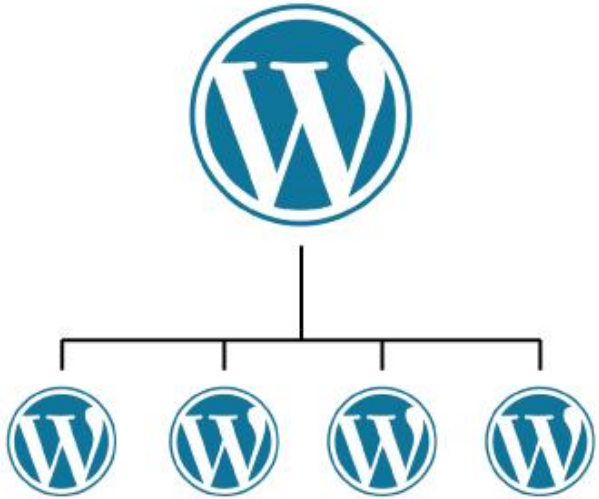
By Robert Windisch @ Syde

Multisite?

Different WordPress Installations

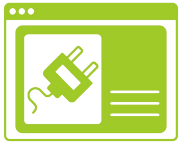


WordPress Multisite



https://wishdesk.com/blog/wordpress-multisitevs-multiple-single-sites

Benefits



Central updates for plugins and themes

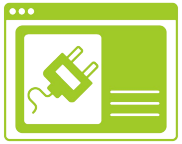


Global content for all sites



Central management of user access

Drawbacks



Central updates for plugins and themes



Plugins can have issues



Single point of failure

Plugins in Multisite

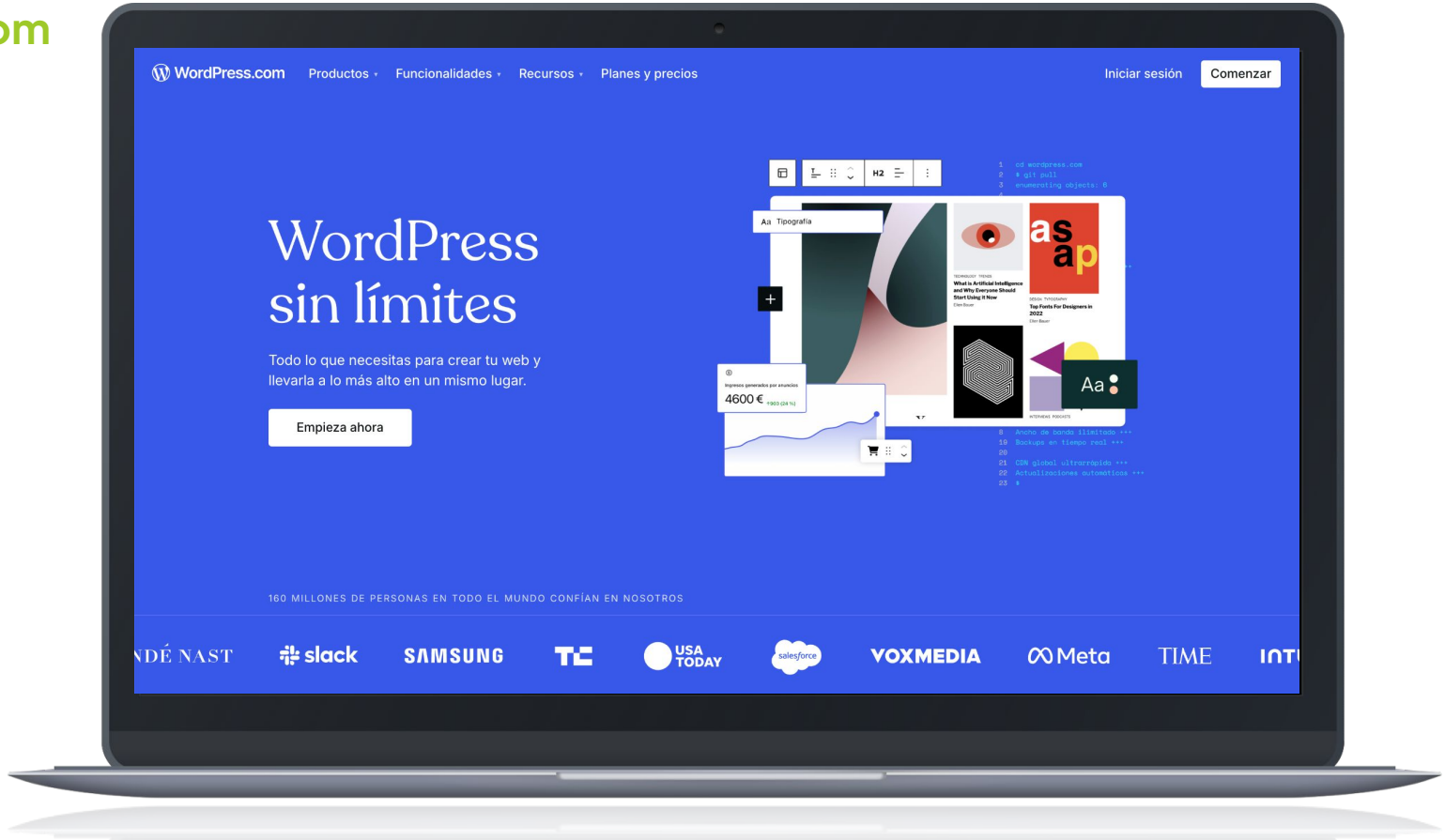
- non-compatible
- passively compatible
- actively compatible

Possibilities with Multisite

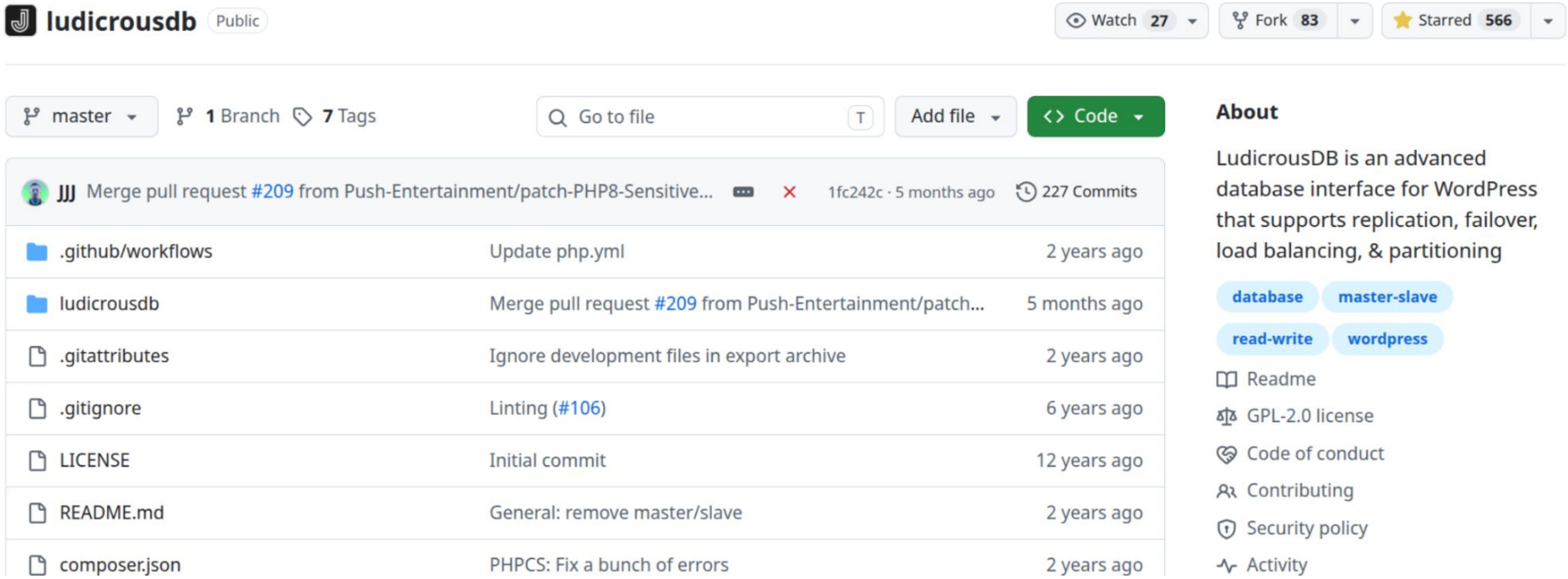
- Use different themes for each site
- Enable certain plugins only on specific sites
- Connecting Domains with sites

Multisite in Practice

WordPress.com



More than 1000 sites?


ludicrousdb
Public
Watch 27
Fork 83
Starred 566
master
1 Branch
7 Tags
Go to file
T
Add file
Code
JJJ Merge pull request #209 from Push-Entertainment/patch-PHP8-Sensitive...
1fc242c · 5 months ago
227 Commits
.github/workflows
Update php.yml
2 years ago
ludicrousdb
Merge pull request #209 from Push-Entertainment/patch...
5 months ago
.gitattributes
Ignore development files in export archive
2 years ago
.gitignore
Linting (#106)
6 years ago
LICENSE
Initial commit
12 years ago
README.md
General: remove master/slave
2 years ago
composer.json
PHPCS: Fix a bunch of errors
2 years ago
About
LudicrousDB is an advanced database interface for WordPress that supports replication, failover, load balancing, & partitioning
database
master-slave
read-write
wordpress
Readme
GPL-2.0 license
Code of conduct
Contributing
Security policy
Activity

LudicrousDB

LudicrousDB is an advanced database interface for WordPress that supports replication, fail-over, load balancing, and partitioning, based on Automattic's HyperDB drop-in.

0. Installation

Files

Copy the main `ludicrousdb` plugin folder & its contents to either:

- `wp-content/plugins/ludicrousdb/`
- `wp-content/mu-plugins/ludicrousdb/`

It does not matter which one; LudicrousDB will figure it out. The folder name should be exactly `ludicrousdb`. Be careful when you do "Download ZIP" from github and unzip.

Drop-ins

WordPress supports a few "drop-in" style plugins, used for advanced overriding of a few specific pieces of functionality.

LudicrousDB includes 3 basic database drop-ins:

- `db.php` <-> `wp-content/db.php` - Bootstrap for replacement `$wpdb` object
- `db-error.php` <-> `wp-content/db-error.php` - Endpoint for fatal database error output to users
- `db-config.php` <-> `ABSPATH/db-config.php` - For configuring your database environment

Sample Configuration 2: Partitioning

This example shows a setup where the multisite blog tables have been separated from the global dataset.

```
$wpdb->add_database( array(
    'host'     => 'global.db.example.com',
    'user'     => 'globaluser',
    'password' => 'globalpassword',
    'name'     => 'globaldb',
) );

$wpdb->add_database( array(
    'host'     => 'blog.db.example.com',
    'user'     => 'bloguser',
    'password' => 'blogpassword',
    'name'     => 'blogdb',
    'dataset'  => 'blog',
) );

$wpdb->add_callback( 'my_db_callback' );

// Multisite blog tables are "{$base_prefix}{$blog_id}_*"
function my_db_callback( $query, $wpdb ) {
    if ( preg_match("/^{$wpdb->base_prefix}\d+_/i", $wpdb->table) ) {
        return 'blog';
    }
}
```

3. Network topology / Datacenter awareness

When your databases are located in separate physical locations there is typically an advantage to connecting to a nearby server instead of a more distant one. The read and write parameters can be used to place servers into logical groups of more or less preferred connections. Lower numbers indicate greater preference.

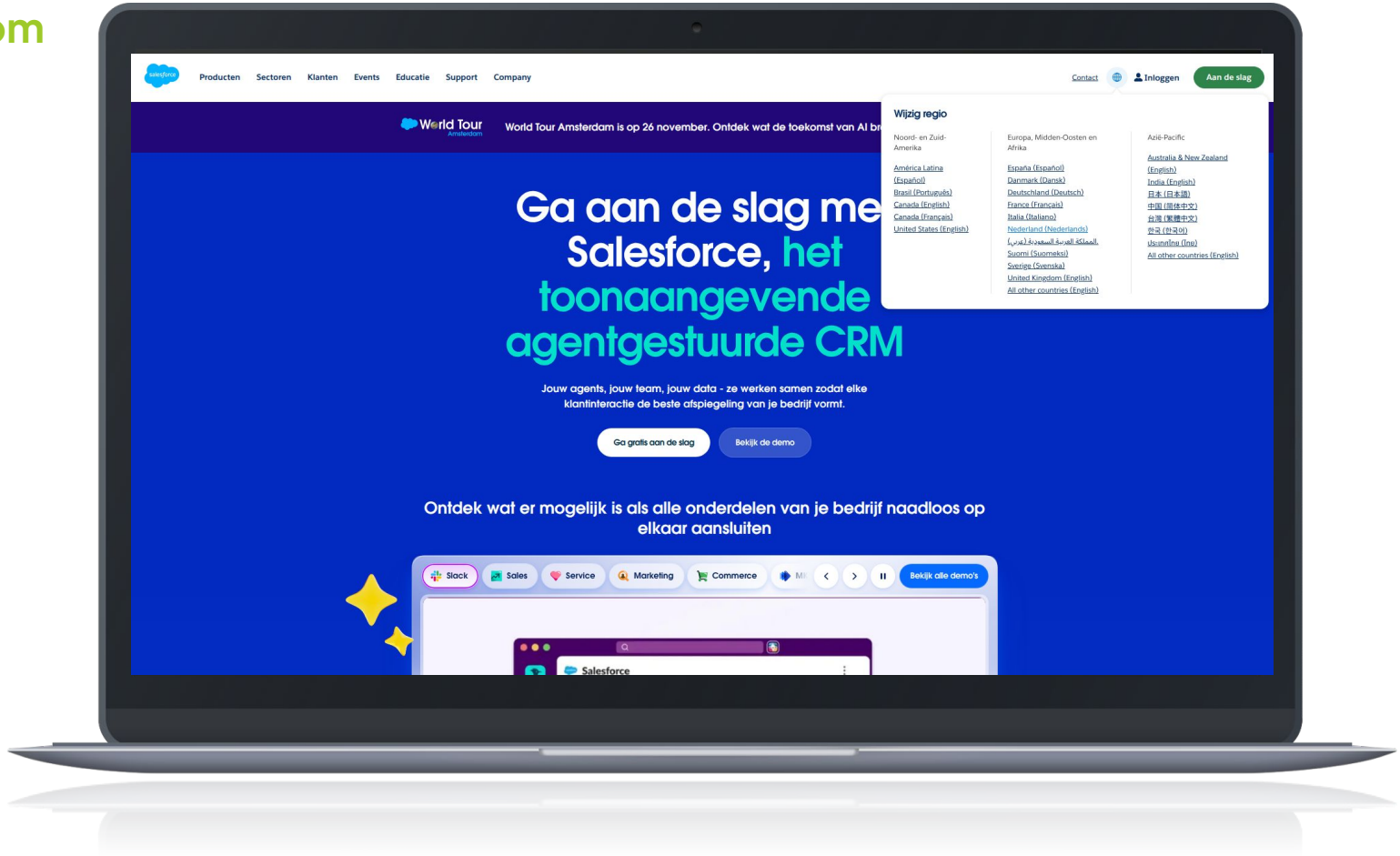
This configuration instructs LudicrousDB to try reading from one of the local replicas at random. If that replica is unreachable or refuses the connection, the other replica will be tried, followed by the primary, and finally the remote replicas in random order.

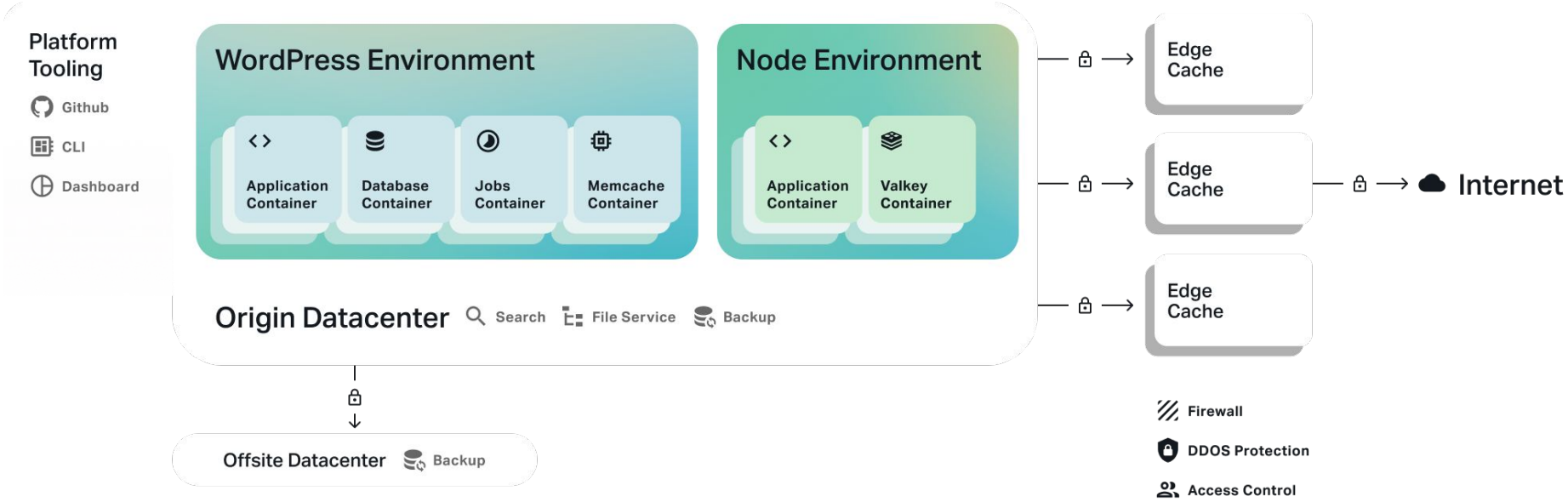
```
Local replica 1:   'write' => 0, 'read' => 1,
Local replica 2:   'write' => 0, 'read' => 1,
Local primary:     'write' => 1, 'read' => 2,
Remote replica 1:  'write' => 0, 'read' => 3,
Remote replica 2:  'write' => 0, 'read' => 3,
```

In the other datacenter, the primary would be remote. We would take that into account while deciding where to send reads. Writes would always be sent to the primary, regardless of proximity.

```
Local replica 1:   'write' => 0, 'read' => 1,
Local replica 2:   'write' => 0, 'read' => 1,
Remote replica 1:  'write' => 0, 'read' => 2,
Remote replica 2:  'write' => 0, 'read' => 2,
Remote primary:    'write' => 1, 'read' => 3,
```

Salesforce.com

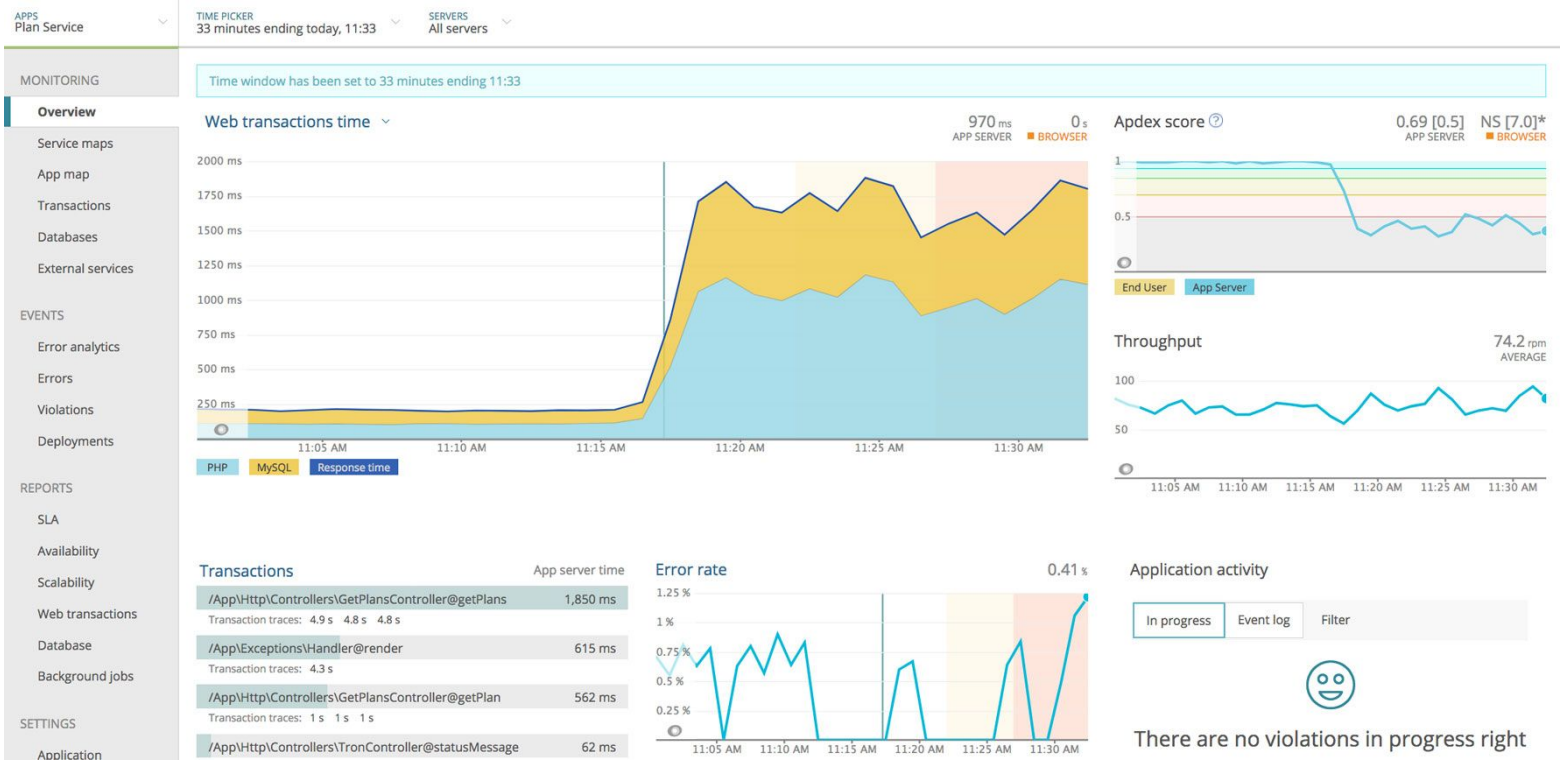



Platform Tooling
Github
CLI
Dashboard
WordPress Environment
Application Container
Database Container
Jobs Container
Memcache Container
Node Environment
Application Container
Valkey Container
Origin Datacenter
Search
File Service
Backup
Offsite Datacenter
Backup
Edge Cache
Edge Cache
Edge Cache
Internet
Firewall
DDOS Protection
Access Control

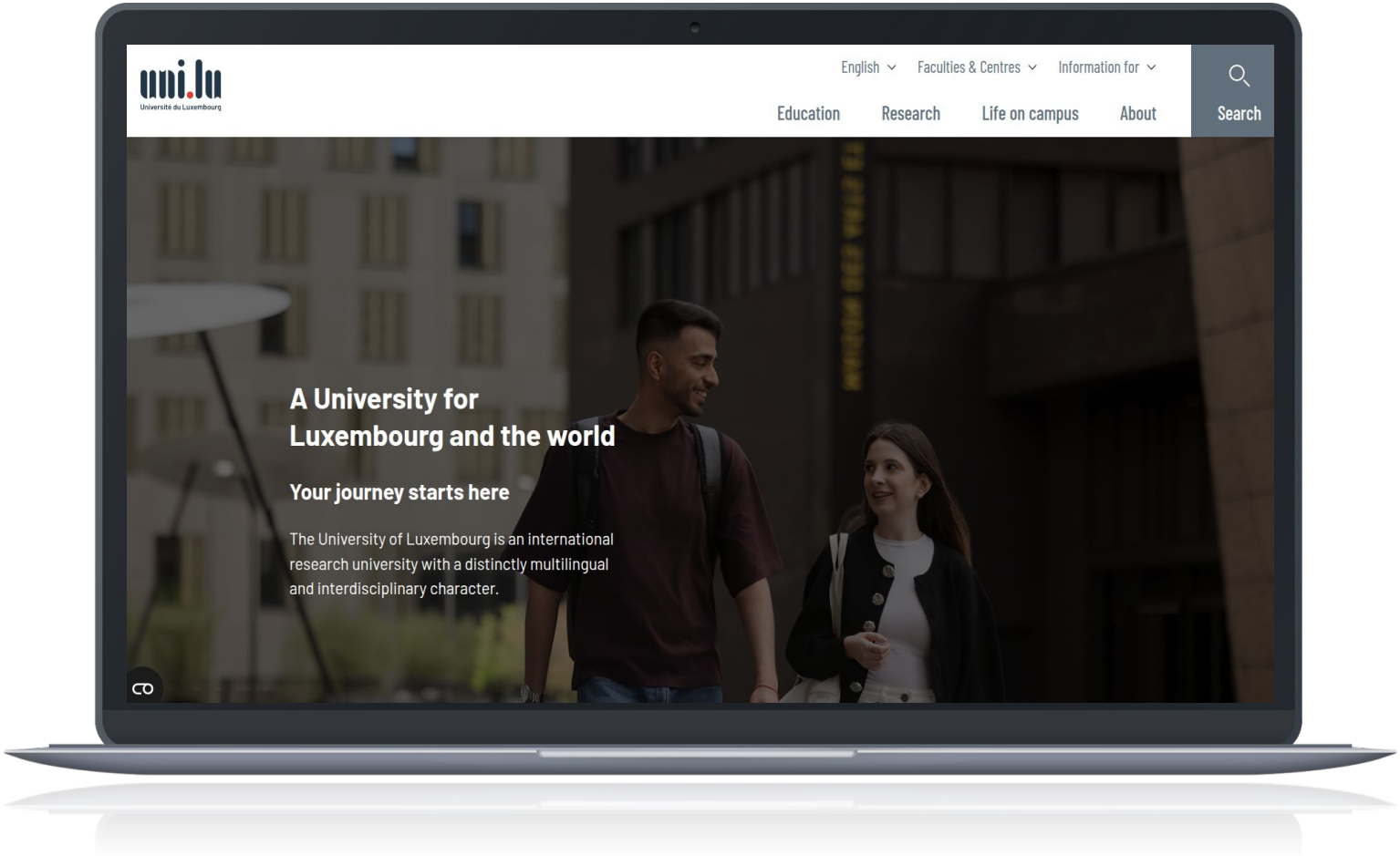
Basics of Enterprise scaling

- Read-only file system
- web servers just run code
- caching (page, object, query, CDN)
- **Do not run slow code!**

New Relic


APPS
Plan Service
TIME PICKER
33 minutes ending today, 11:33
SERVERS
All servers
MONITORING
Overview
Service maps
App map
Transactions
Databases
External services
EVENTS
Error analytics
Errors
Violations
Deployments
REPORTS
SLA
Availability
Scalability
Web transactions
Database
Background jobs
SETTINGS
Application
Time window has been set to 33 minutes ending 11:33
Web transactions time
970 ms APP SERVER
0 s BROWSER
2000 ms
1750 ms
1500 ms
1250 ms
1000 ms
750 ms
500 ms
250 ms
11:05 AM
11:10 AM
11:15 AM
11:20 AM
11:25 AM
11:30 AM
PHP
MySQL
Response time
Apdex score
0.69 [0.5] APP SERVER
NS [7.0]* BROWSER
1
0.5
End User
App Server
Throughput
74.2 rpm AVERAGE
100
50
11:05 AM
11:10 AM
11:15 AM
11:20 AM
11:25 AM
11:30 AM
Transactions
App server time
/App\Http\Controllers\GetPlansController@getPlans 1,850 ms
Transaction traces: 4.9 s 4.8 s 4.8 s
/App\Exceptions\Handler@render 615 ms
Transaction traces: 4.3 s
/App\Http\Controllers\GetPlansController@getPlan 562 ms
Transaction traces: 1 s 1 s 1 s
/App\Http\Controllers\TronController@statusMessage 62 ms
Error rate
0.41 %
1.25 %
1 %
0.75 %
0.5 %
0.25 %
11:05 AM
11:10 AM
11:15 AM
11:20 AM
11:25 AM
11:30 AM
Application activity
In progress
Event log
Filter
There are no violations in progress right

University of Luxembourg



WP-CLI examples

Run a command (here: plugin deactivation) for every site in a multisite network:

```
for URL in $(wp site list --field=url); do wp plugin deactivate {plugin} --url="$URL"; done
```

Add a user with a certain role in all sites in a multisite network (if you already have a user):

```
for URL in $(wp site list --field=url); do wp user add-role {username} {role} --url="$URL"; done
```

Commands

Command	Description
wp ability	Lists, inspects, and executes abilities registered via the WordPress Abilities API.
wp admin	Open /wp-admin/ in a browser.
wp block	Manages WordPress block editor blocks and related entities.
wp cache	Adds, removes, fetches, and flushes the WP Object Cache object.
wp cap	Adds, removes, and lists capabilities of a user role.
wp cli	Reviews current WP-CLI info, checks for updates, or views defined aliases.
wp comment	Creates, updates, deletes, and moderates comments.
wp config	Generates and reads the wp-config.php file.
wp core	Downloads, installs, updates, and manages a WordPress installation.
wp cron	Tests, runs, and deletes WP-Cron events; manages WP-Cron schedules.
wp db	Performs basic database operations using credentials stored in wp-config.php.
wp dist-archive	Create a distribution archive based on a project's .distignore file.
wp embed	Inspects oEmbed providers, clears embed cache, and more.
wp eval	Executes arbitrary PHP code.
wp eval-file	Loads and executes a PHP file.
wp export	Exports WordPress content to a WXR file.
wp find	Find WordPress installations on the filesystem.
wp help	Gets help on WP-CLI, or on a specific command.
wp i18n	Provides internationalization tools for WordPress projects.
wp import	Imports content from a given WXR file.
wp language	Installs, activates, and manages language packs.
wp maintenance-mode	Activates, deactivates or checks the status of the maintenance mode of a site.
wp media	Imports files as attachments, regenerates thumbnails, or lists registered image sizes.
wp menu	Lists, creates, assigns, and deletes the active theme's navigation menus.
wp network	Perform network-wide operations.
wp option	Retrieves and sets site options, including plugin and WordPress settings.
wp package	Lists, installs, and removes WP-CLI packages.
wp plugin	Manages plugins, including installs, activations, and updates.
wp post	Manages posts, content, and meta.
wp post-type	Retrieves details on the site's registered post types.
wp profile	Quickly identify what's slow with WordPress.
wp rewrite	Lists or flushes the site's rewrite rules, updates the permalink structure.
wp role	Manages user roles, including creating new roles and resetting to defaults.
wp scaffold	Generates code for post types, taxonomies, plugins, child themes, etc.
wp search-replace	Searches/replaces strings in the database.
wp server	Launches PHP's built-in web server for a specific WordPress installation.
wp shell	Opens an interactive PHP console for running and testing PHP code.
wp sidebar	Lists registered sidebars.
wp site	Creates, deletes, empties, moderates, and lists one or more sites on

When not to use?

Do not try to be WordPress.com!

- Different clients
- Maybe even WooCommerce
- One clients leaves. Have fun with the migration!

Bonus question

What is the difference between network-wide activated and mu-plugins?

#multimultisite

Showing rows 0 - 0 (1 total, Query took 0.0001 seconds.)

```
SELECT * FROM `wp_site`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Re

Show all | Number of rows: 25 Filter rows: Searc

Extra options

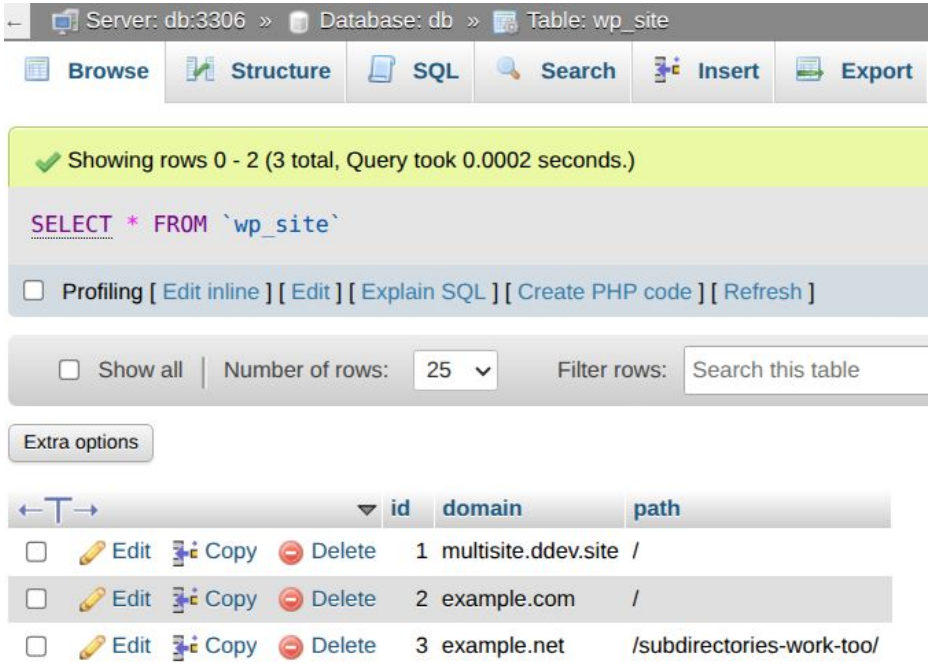
	id	domain	path
Edit Copy Delete	1	multisite.ddev.site	/

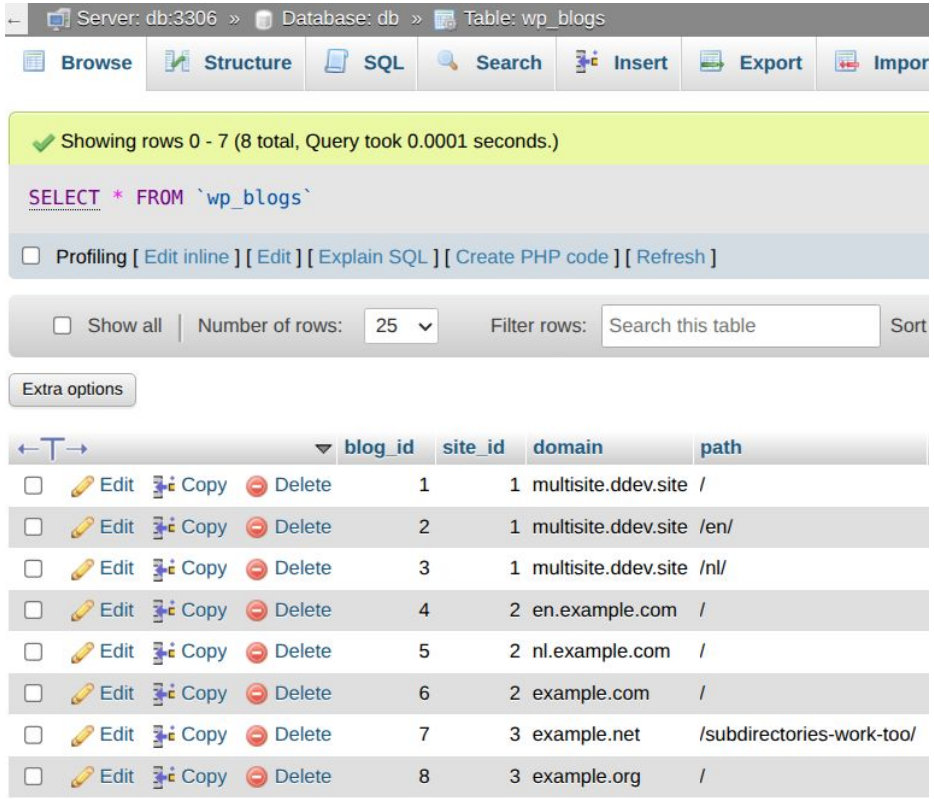
Server: db:3306 » Database: db » Table: wp_blogs

Browse | Structure | SQL | Search | Inse

Table structure | Relation view

#	Name	Type	Collation	Attribu
1	blog_id	bigint(20)		UNSIGNE
2	site_id	bigint(20)		UNSIGNE
3	domain	varchar(200)	utf8mb4_unicode_520_ci	
4	path	varchar(100)	utf8mb4_unicode_520_ci	
5	registered	datetime		


Server: db:3306 » Database: db » Table: wp_site
Browse
Structure
SQL
Search
Insert
Export
Showing rows 0 - 2 (3 total, Query took 0.0002 seconds.)
SELECT * FROM `wp_site`
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all | Number of rows: 25 Filter rows: Search this table
Extra options
id domain path
Edit Copy Delete 1 multisite.ddev.site /
Edit Copy Delete 2 example.com /
Edit Copy Delete 3 example.net /subdirectories-work-too/


Server: db:3306 » Database: db » Table: wp_blogs
Browse
Structure
SQL
Search
Insert
Export
Impor
Showing rows 0 - 7 (8 total, Query took 0.0001 seconds.)
SELECT * FROM `wp_blogs`
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all | Number of rows: 25 Filter rows: Search this table Sort
Extra options
blog_id site_id domain path
Edit Copy Delete 1 1 multisite.ddev.site /
Edit Copy Delete 2 1 multisite.ddev.site /en/
Edit Copy Delete 3 1 multisite.ddev.site /nl/
Edit Copy Delete 4 2 en.example.com /
Edit Copy Delete 5 2 nl.example.com /
Edit Copy Delete 6 2 example.com /
Edit Copy Delete 7 3 example.net /subdirectories-work-too/
Edit Copy Delete 8 3 example.org /



WP Multi Network

By Triple J Software, Inc.

Details | Reviews | Installation | Development

Description

Turn your WordPress Multisite installation into many multisite networks, surrounding one global set of users.

- Reveals hidden WordPress Multisite functionality.
- Includes a “Networks” top-level Network-Admin menu.
- Includes a List Table for viewing available networks.
- Allows moving subsites between networks.
- Allows global administrators to create new networks with their own sites and

Version

Last update

Active instal

WordPress v

Tested up to

PHP version

Languages

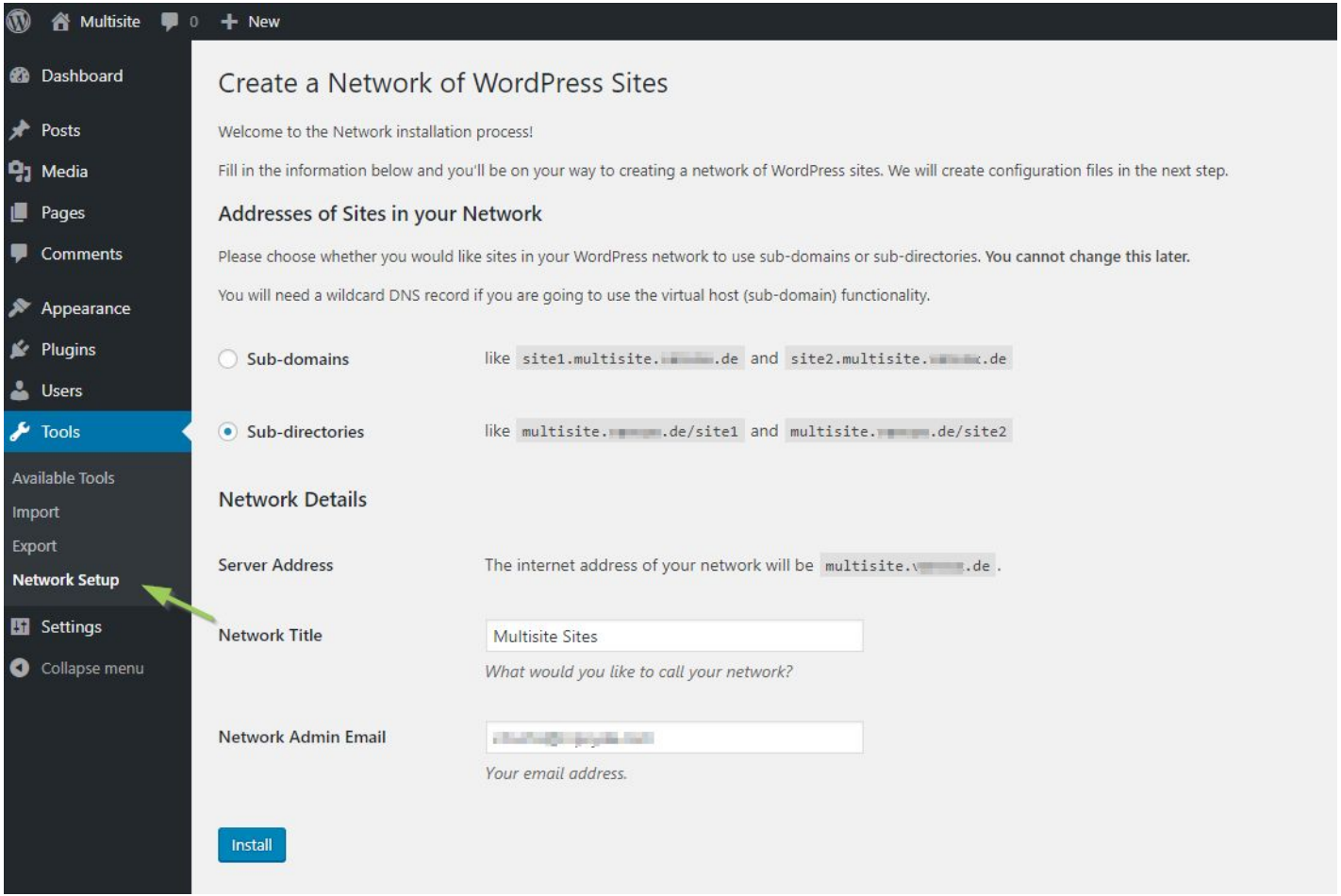
How to activate?

Installation

Allow Multisite in wp-config.php

The first step is to activate the Multisite feature in the file *wp-config.php*.

1. Set up a FTP connection to your website.
2. Open the file *wp-config.php*, which is is located in the main directory of your WordPress, and add the line
   `define('WP_ALLOW_MULTISITE', true);`
   above the line:
   `/* That's all, stop editing! Happy blogging. */`


Multisite
0
New
Dashboard
Posts
Media
Pages
Comments
Appearance
Plugins
Users
Tools
Available Tools
Import
Export
Network Setup
Settings
Collapse menu
Create a Network of WordPress Sites
Welcome to the Network installation process!
Fill in the information below and you'll be on your way to creating a network of WordPress sites. We will create configuration files in the next step.
Addresses of Sites in your Network
Please choose whether you would like sites in your WordPress network to use sub-domains or sub-directories. You cannot change this later.
You will need a wildcard DNS record if you are going to use the virtual host (sub-domain) functionality.
Sub-domains
like site1.multisite.
.de
and site2.multisite.
.de
Sub-directories
like multisite.
.de/site1
and multisite.
.de/site2
Network Details
Server Address
The internet address of your network will be multisite.
.de.
Network Title
Multisite Sites
What would you like to call your network?
Network Admin Email
Your email address.
Install

1. Set up a FTP connection to your website.
2. Add the first code snippet to your *wp-config.php* directly above the line
   `/* That's all, stop editing! Happy blogging. */`
   The snippet looks like this, but adapted to your own site:

```
define('MULTISITE', true);
define('SUBDOMAIN_INSTALL', true);
define('DOMAIN_CURRENT_SITE', 'My Website');
define('PATH_CURRENT_SITE', '/');
define('SITE_ID_CURRENT_SITE', 1);
define('BLOG_ID_CURRENT_SITE', 1);
```

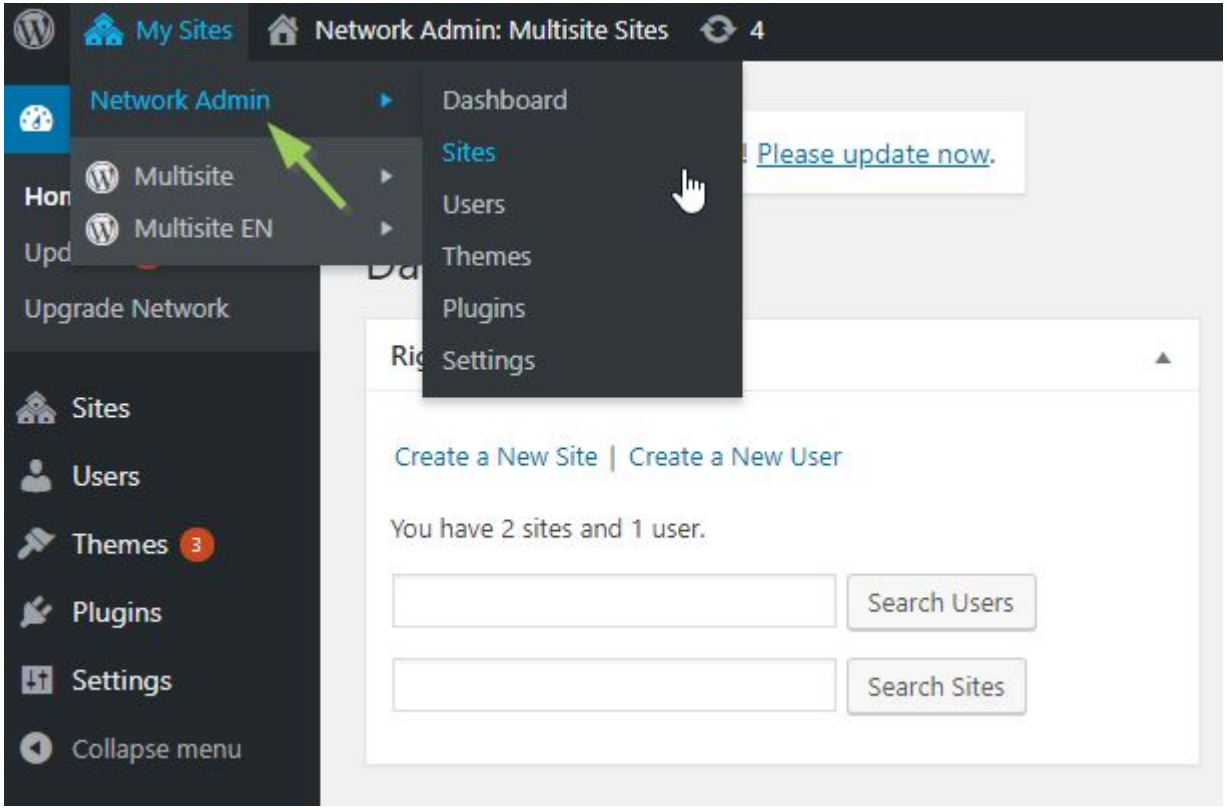
3. Add the second code snippet to the *.htaccess* file and replace other WordPress rules.

```
RewriteEngine On
RewriteBase /
RewriteRule ^index\.php$ - [L]

# add a trailing slash to /wp-admin
RewriteRule ^([_0-9a-zA-Z-]+/)?wp-admin$ $1wp-admin/ [R=301,L]

RewriteCond %{REQUEST_FILENAME} -f [OR]
RewriteCond %{REQUEST_FILENAME} -d
RewriteRule ^ - [L]
RewriteRule ^([_0-9a-zA-Z-]+/)?(wp-(content|admin|includes).*) $2 [L]
RewriteRule ^([_0-9a-zA-Z-]+/)?(.*\.php)$ $2 [L]
RewriteRule . index.php [L]
```

Done!


My Sites
Network Admin: Multisite Sites
4
Network Admin
Multisite
Multisite EN
Dashboard
Sites
Users
Themes
Plugins
Settings
Please update now.
Upgrade Network
Sites
Users
Themes 3
Plugins
Settings
Collapse menu
Create a New Site | Create a New User
You have 2 sites and 1 user.
Search Users
Search Sites

Using WP-CLI

Search resources

wp core multisite-convert

Transforms an existing single-site installation into a multisite installation.

View Open Issues (0) View Closed Issues (18) Create New Issue

Creates the multisite database tables, and adds the multisite constants to wp-config.php. For those using WordPress with Apache, remember to update the `.htaccess` file with the appropriate multisite rewrite rules. Review the multisite documentation for more details about how multisite works.

What did we learn?

Summary

- Multisite is stable and scalable
- Available on every WordPress installation. Just activate it!
- Find me and also ask me Multisite questions.

Questions?

Thank you!

Robert Windisch

@nullbytes