WordCamp Europe
Basel 2025

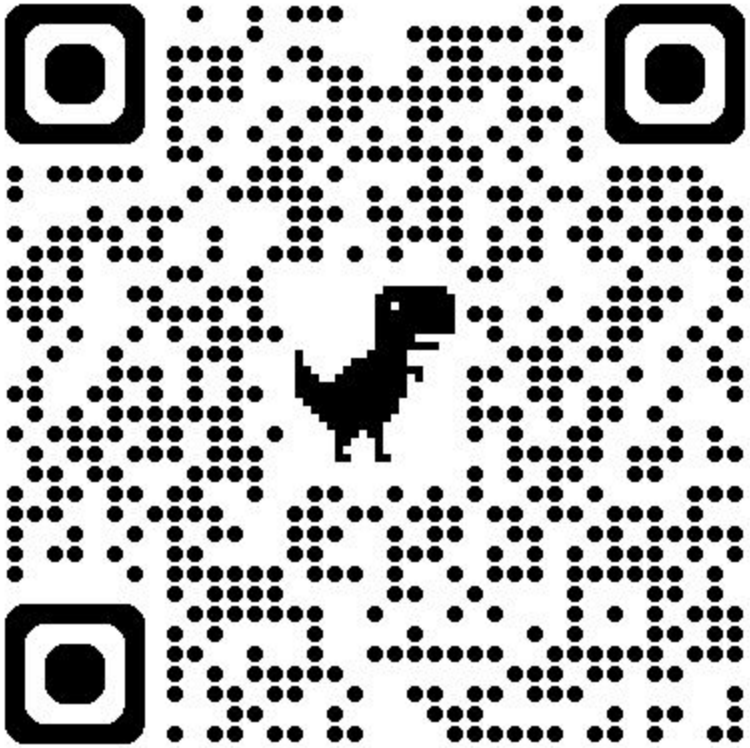
Multilingual WordPress

for developers

Dennis Ploetner

Multilingual WordPress

`for developers`



Try it yourself

Explore the Companion Plugin

Scan the QR code to explore the code examples from this session.

Everything is available in the GitHub repository!


i18n? L10n?
g11n? a11y? p13n?
... I'm lost.

i18n & L10n: What's the Difference?

i18n

The act or process of making a product suitable for international markets, typically by making text messages easily translatable and ensuring support of non-Latin character sets.

It can be done.

Developer Zone

L10n

The act or process of making a product suitable for use in a particular country or region.

It has been done.

Translator Zone

API Functions

The Good, the Bad and the Ugly

```php
<?php

$translated_string = __(
    original: 'Demo__ Text',
    'multilingual-wp4devs'
);

$translated_string_with_context = _x(
    text: 'Demo_x Text',
    context: 'Demo Context',
    domain: 'multilingual-wp4devs'
);

$stars = rand( 1, 5 );

_n(
    single: '%d star',
    plural: '%d stars',
    $stars,
    domain: 'multilingual-wp4devs'
);
```

```php
<?php

// phpcs:ignore WordPress.Security.EscapeOutput.UnsafePrintingFunction
_e(
    text: 'Demo_e Text',
    domain: 'multilingual-wp4devs'
);

// phpcs:ignore WordPress.Security.EscapeOutput.UnsafePrintingFunction
_ex(
    text: 'Demo_x Text',
    context: 'Demo Context',
    domain: 'multilingual-wp4devs'
);
```

```php
<?php

echo esc_html__( text: 'Demo__ Text', domain: 'multilingual-wp4devs' );

echo esc_attr__( text: 'Demo__ Text', domain: 'multilingual-wp4devs' );

esc_html_e( text: 'Demo_e Text', domain: 'multilingual-wp4devs' );

esc_attr_e( text: 'Demo_e Text', domain: 'multilingual-wp4devs' );

echo esc_html_x(
    text: 'Demo_x Text',
    context: 'Demo Context',
    domain: 'multilingual-wp4devs'
);

echo esc_attr_x(
    text: 'Demo_x Text',
    context: 'Demo Context',
    domain: 'multilingual-wp4devs'
);
```

How to be nice to translators

Give context, not just strings

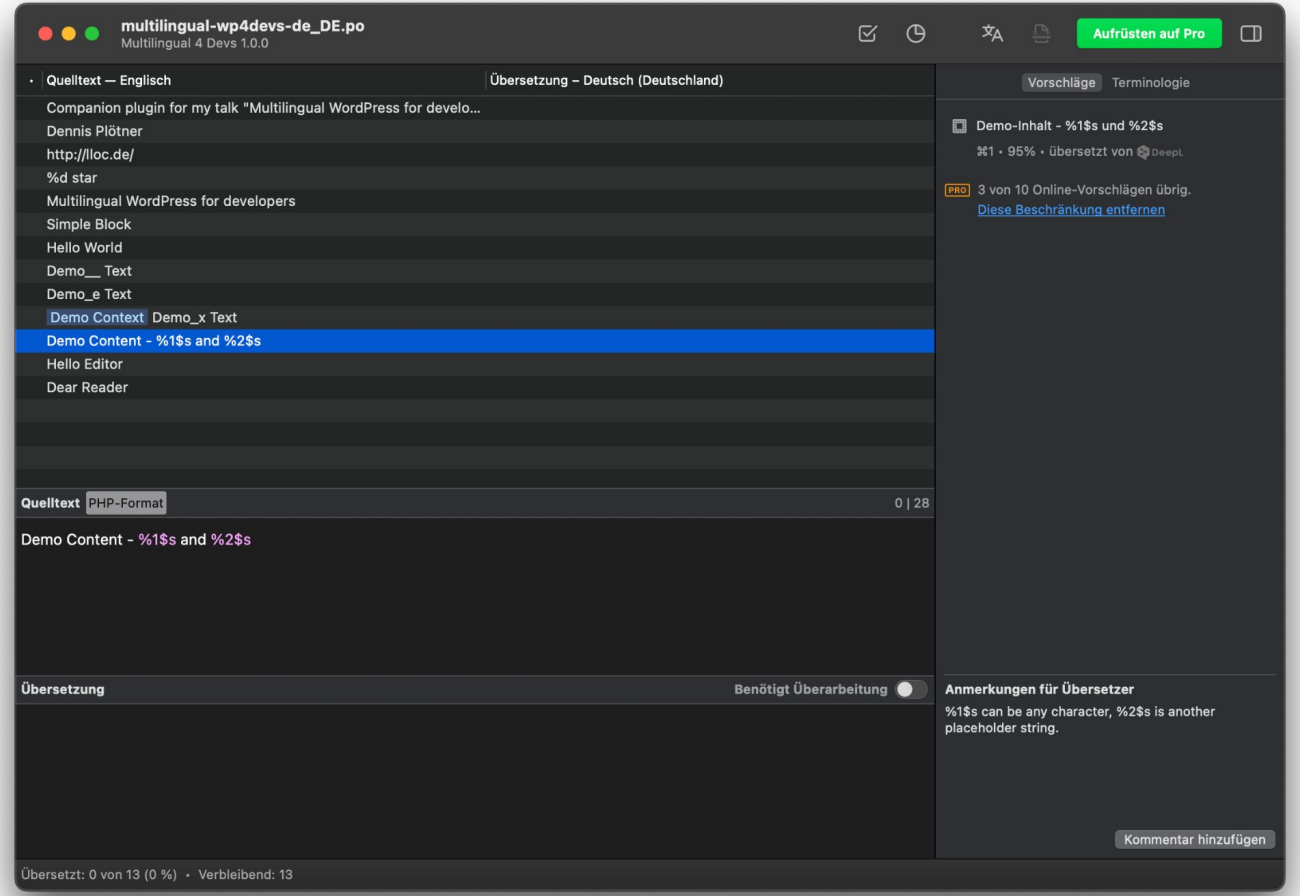
Use translator comments to clarify meaning, tone, or placeholders - they make a big difference in translation quality.

```php
<?php

$stars = wp_rand( 1, 5 );

$stars_str = sprintf(
    // translators: %d is an integer for a star-rating.
    _n( single: '%d star', plural: '%d stars', $stars, domain: 'multilingual-wp4devs' ),
    $stars
);

$demo_str = sprintf(
    // translators: %1$s can be any character, %2$s is another placeholder string.
    __( original: 'Demo Content - %1$s and %2$s', 'multilingual-wp4devs' ),
    ...values: 'A',
    'B'
);
```


multilingual-wp4devs-de_DE.po
Multilingual 4 Devs 1.0.0
Aufrüsten auf Pro
Quelltext — Englisch
Übersetzung – Deutsch (Deutschland)
Companion plugin for my talk "Multilingual WordPress for develo...
Dennis Plötner
http://lloc.de/
%d star
Multilingual WordPress for developers
Simple Block
Hello World
Demo__ Text
Demo_e Text
Demo Context Demo_x Text
Demo Content - %1$s and %2$s
Hello Editor
Dear Reader
Quelltext
PHP-Format
0 | 28
Demo Content - %1$s and %2$s
Übersetzung
Benötigt Überarbeitung
Vorschläge
Terminologie
Demo-Inhalt - %1$s und %2$s
⌘1 • 95% • übersetzt von DeepL
PRO
3 von 10 Online-Vorschlägen übrig.
Diese Beschränkung entfernen
Anmerkungen für Übersetzer
%1$s can be any character, %2$s is another placeholder string.
Kommentar hinzufügen
Übersetzt: 0 von 13 (0 %) • Verbleibend: 13

And what about JS?

Internationalization isn’t just for PHP

WordPress offers a dedicated i18n package for JavaScript — bringing translation functions to blocks, plugins, and custom scripts.

```
import { __ } from '@wordpress/i18n';
import { registerBlockType } from '@wordpress/blocks';

registerBlockType( blockNameOrMetadata: 'lloc/multilingual-wp4devs', settings: {
    apiVersion: 3,
    title: __( text: 'Simple Block', domain: 'multilingual-wp4devs' ),
    category: 'widgets',

    edit: () => {
        return <p>{ __( text: 'Hello Editor', domain: 'multilingual-wp4devs' ) }</p>;
    },

    save: () => {
        return <p>{ __( text: 'Dear Reader', domain: 'multilingual-wp4devs' ) }</p>;
    },
} );
```

There is more to explore

Dig deeper into WordPress i18n for PHP and JavaScript

All the PHP core functions in depth:
/wp-includes/l10n.php

Explore the JavaScript API powering translations:
npmjs.com/package/@wordpress/i18n

Translation files

Overview of .pot, .po, .mo, .json & .l10n.php files

What these files are for

- **.pot**

  Template file with all translatable strings (no translations yet)

- **.po**

  Editable file with original + translated strings

- **.mo**

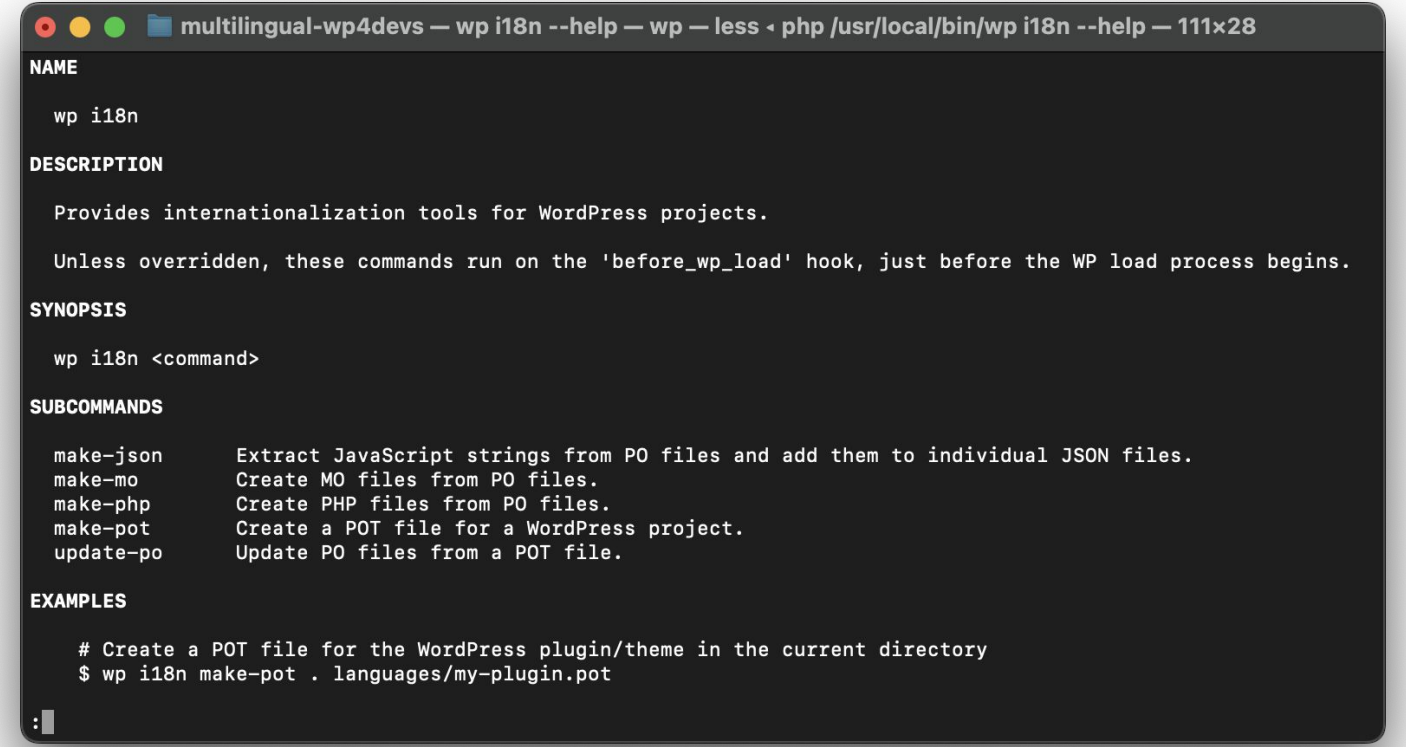
  Machine-readable binary file generated from .po

- **.json**

  JavaScript translation files in JED format, generated from .po files. These files are used to provide localized strings in JavaScript, enabling internationalization support for scripts in WordPress.

- **.l10n.php**

  Introduced in WordPress 6.5, these PHP-based translation files offer improved performance, leading to faster translation loading times.


multilingual-wp4devs — wp i18n --help — wp — less ◂ php /usr/local/bin/wp i18n --help — 111×28
NAME
wp i18n
DESCRIPTION
Provides internationalization tools for WordPress projects.
Unless overridden, these commands run on the 'before_wp_load' hook, just before the WP load process begins.
SYNOPSIS
wp i18n <command>
SUBCOMMANDS
make-json Extract JavaScript strings from PO files and add them to individual JSON files.
make-mo Create MO files from PO files.
make-php Create PHP files from PO files.
make-pot Create a POT file for a WordPress project.
update-po Update PO files from a POT file.
EXAMPLES
# Create a POT file for the WordPress plugin/theme in the current directory
$ wp i18n make-pot . languages/my-plugin.pot
:

Load text domains

How and when

```php
<?php

/**
 * The hook 'init' should be used to load the plugin's translation files.
 *
 * Don't use 'plugins_loaded' since it will create a warning!
 */
add_action(
    hook_name: 'init',
    static function () {
        load_plugin_textdomain(
            domain: 'multilingual-wp4devs',
            deprecated: false,
            plugin_rel_path: __DIR__ . '/languages'
        );
    }
);
```

```php
<?php

wp_register_script(
    handle: 'multilingual-wp4devs-script',
    plugins_url( path: 'js/index.js', plugin: __FILE__ ),
    array( 'wp-blocks', 'react', 'wp-i18n', 'wp-block-editor' ),
    ver: '1.0.0',
    args: true
);

register_block_type(
    block_type: 'lloc/multilingual-wp4devs',
    array(
        'api_version'   => 3,
        'editor_script' => 'multilingual-wp4devs-script',
    )
);

wp_set_script_translations(
    handle: 'multilingual-wp4devs-script',
    domain: 'multilingual-wp4devs',
    path: plugin_dir_path( file: __FILE__ ) . 'languages'
);
```

```php
<?php
/**
 * Multilingual WordPress for developers
 *
 * [...]
 *
 * Text Domain: multilingual-wp4devs
 * Domain Path: /languages/
 */
```

Skipping load_*_textdomain()

When WordPress handles translation loading for you

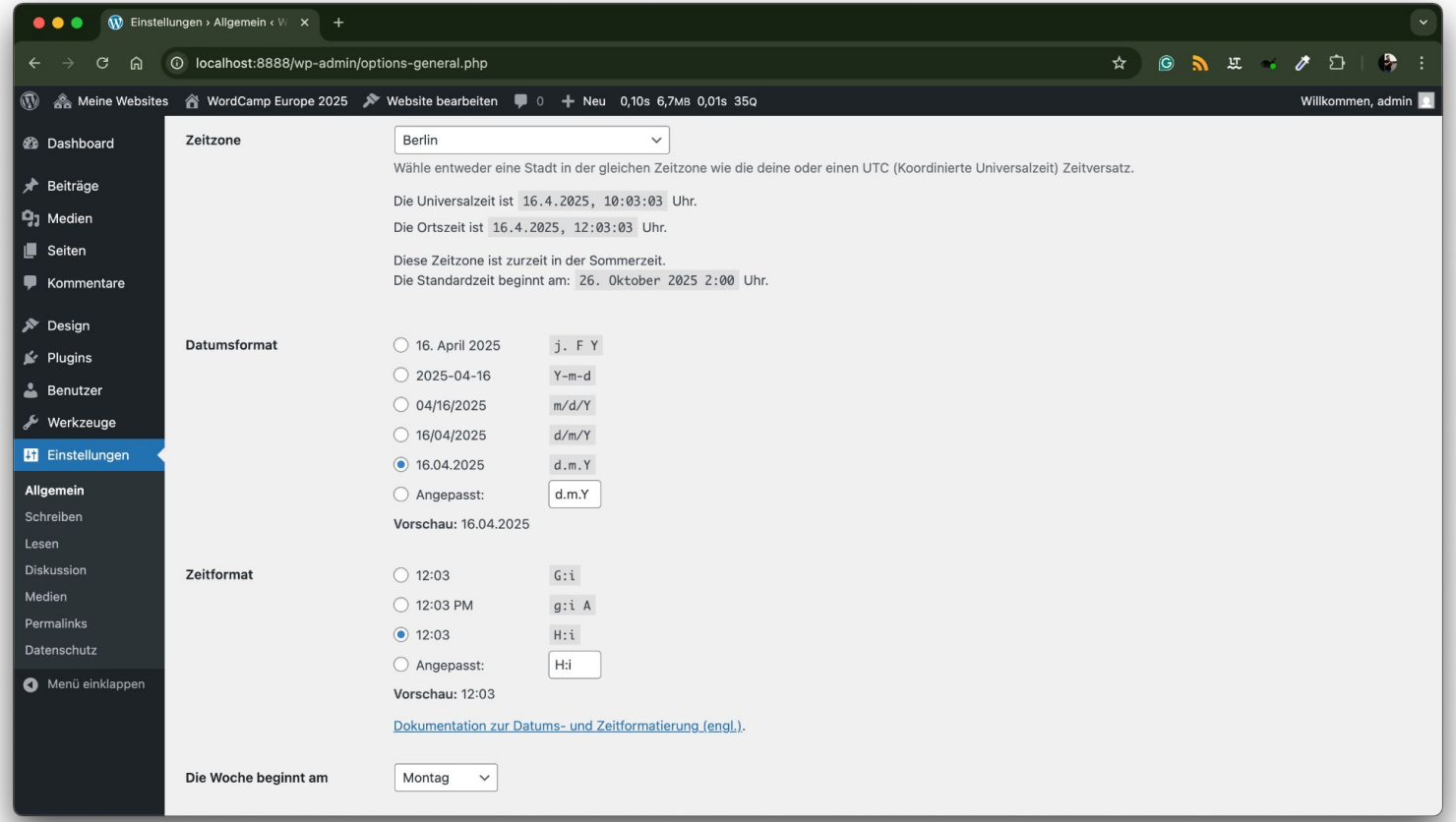
Your plugin or theme is hosted on WordPress.org and requires WP 4.6+

OR

It includes Text Domain and Domain Path headers and requires WP 6.8+

User Settings

What should I consider?


Einstellungen › Allgemein
localhost:8888/wp-admin/options-general.php
Meine Websites
WordCamp Europe 2025
Website bearbeiten
Neu
Willkommen, admin
Dashboard
Beiträge
Medien
Seiten
Kommentare
Design
Plugins
Benutzer
Werkzeuge
Einstellungen
Allgemein
Schreiben
Lesen
Diskussion
Medien
Permalinks
Datenschutz
Menü einklappen
Zeitzone
Berlin
Wähle entweder eine Stadt in der gleichen Zeitzone wie die deine oder einen UTC (Koordinierte Universalzeit) Zeitversatz.
Die Universalzeit ist 16.4.2025, 10:03:03 Uhr.
Die Ortszeit ist 16.4.2025, 12:03:03 Uhr.
Diese Zeitzone ist zurzeit in der Sommerzeit.
Die Standardzeit beginnt am: 26. Oktober 2025 2:00 Uhr.
Datumsformat
16. April 2025 j. F Y
2025-04-16 Y-m-d
04/16/2025 m/d/Y
16/04/2025 d/m/Y
16.04.2025 d.m.Y
Angepasst: d.m.Y
Vorschau: 16.04.2025
Zeitformat
12:03 G:i
12:03 PM g:i A
12:03 H:i
Angepasst: H:i
Vorschau: 12:03
Dokumentation zur Datums- und Zeitformatierung (engl.).
Die Woche beginnt am
Montag

```php
<?php

$date = date_i18n(
    get_option( option: 'date_format' )
);

echo esc_html( $date );

$timestamp = mktime( hour: 13, minute: 42, second: 23, month: 7, day: 17, year: 2025 );
$date      = date_i18n(
    get_option( option: 'date_format' ),
    $timestamp
);

echo esc_html( $date );
```

```php
<?php

$number = number_format_i18n( number: 1234567.89 );

echo esc_html( $number );

$number = number_format_i18n(
    number: 1234567.89,
    decimals: 2
);

echo esc_html( $number );
```

Plugin territory

Multilingual Architectures

So, WordPress does not support multilingual content in core (yet)?

It is all about **relations**!

Dealing with Multilingual Content…

…using a regular single-site WordPress website can be fairly straightforward…

Single Site Approach

Pros:

- One interface for all languages
- Lower technical barrier
- Shared theme & plugin setup

Cons:

- Vendor lock-in
- Performance issues at scale
- Complex relationships (content, plugins & themes)

Dealing with Multilingual Content…

… using Multisite may be the more complex, but also more sophisticated & scalable approach…

Multisite Approach

Pros:

- Close to core behavior
- Full content separation
- Scales well for large projects

Cons:

- Complex to set up and maintain
- May seem less editor-friendly
- Custom work needed for shared content

Dealing with Multilingual Content…

… by paying extra for SaaS solutions to deal with the multilingual complexity for you…

SaaS Solutions

Pros:

- Quick setup
- Automatic translations
- Minimal changes to site structure

Cons:

- Privacy concerns
- Ongoing subscription costs
- Limited customization options

To summarize…

All approaches to multilingualism have their pros and cons. Find out which one suits your project!

Single Site Approach

Pros:

- "Easier"
- Shared resources
- Unified content management

Cons:

- Vendor lock-in
- Performance

Multisite Approach

Pros:

- Core WP
- Scalable
- Separated content

Cons:

- Complex
- Less intuitive

SaaS Solutions

Pros:

- Fast & simple setup
- Auto-translations
- Low impact on the site structure

Cons:

- Costs
- Privacy

🌐 4 Key Tips for Multilingual WordPress Development

🌍 Internationalization & Architecture

Think about internationalization from the start.
Choose the appropriate site architecture for your needs.

🔒 Secure Outputs

Always escape your outputs to avoid translation and security issues.

💬 Translator Context

Provide clear and ample context to translators for accurate string & content localization.

✅ Thorough Testing

Test your multilingual setup carefully across languages and interfaces.

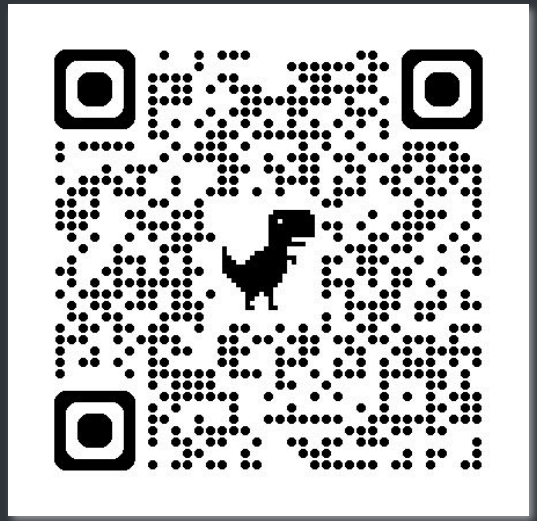
Got questions, multilingual headaches, or success stories?

Let's hear them!

syde

woo PRO Partner

VIP GOLD AGENCY PARTNER



Dennis Ploetner
(@realloc)

WCEU 2025

Thank you!