TRANSLATE YOUR THEME OR PLUGIN TO 10 LANGUAGES IN LESS THAN A WEEK

NodeBig
Linux
PHP
Cappella
Developer
Full
Nginx
Entrepreneur
A
Designer
Speaker
Stack
Data
Nerd
Consultant
Systems
Mac
System
Programmer
Public
CTO
Admin
Architect

› Hi, I am Dovy Paukstys

http://dovy.io

REDUX

Owlet
Rest Assured.
T-Mobile
07:47
100%
Baby Bear's vitals are all normal.
Heart Rate
156
BPM
80
LOW
220
HIGH
Oxygen Level
95 %
85
LOW
100
GOOD

Internationalization

Quite possibly the most painful thing you will ever do.

Really, it just plain sucks. Well, it used to.

WHAT IS INTERNATIONALIZATION?

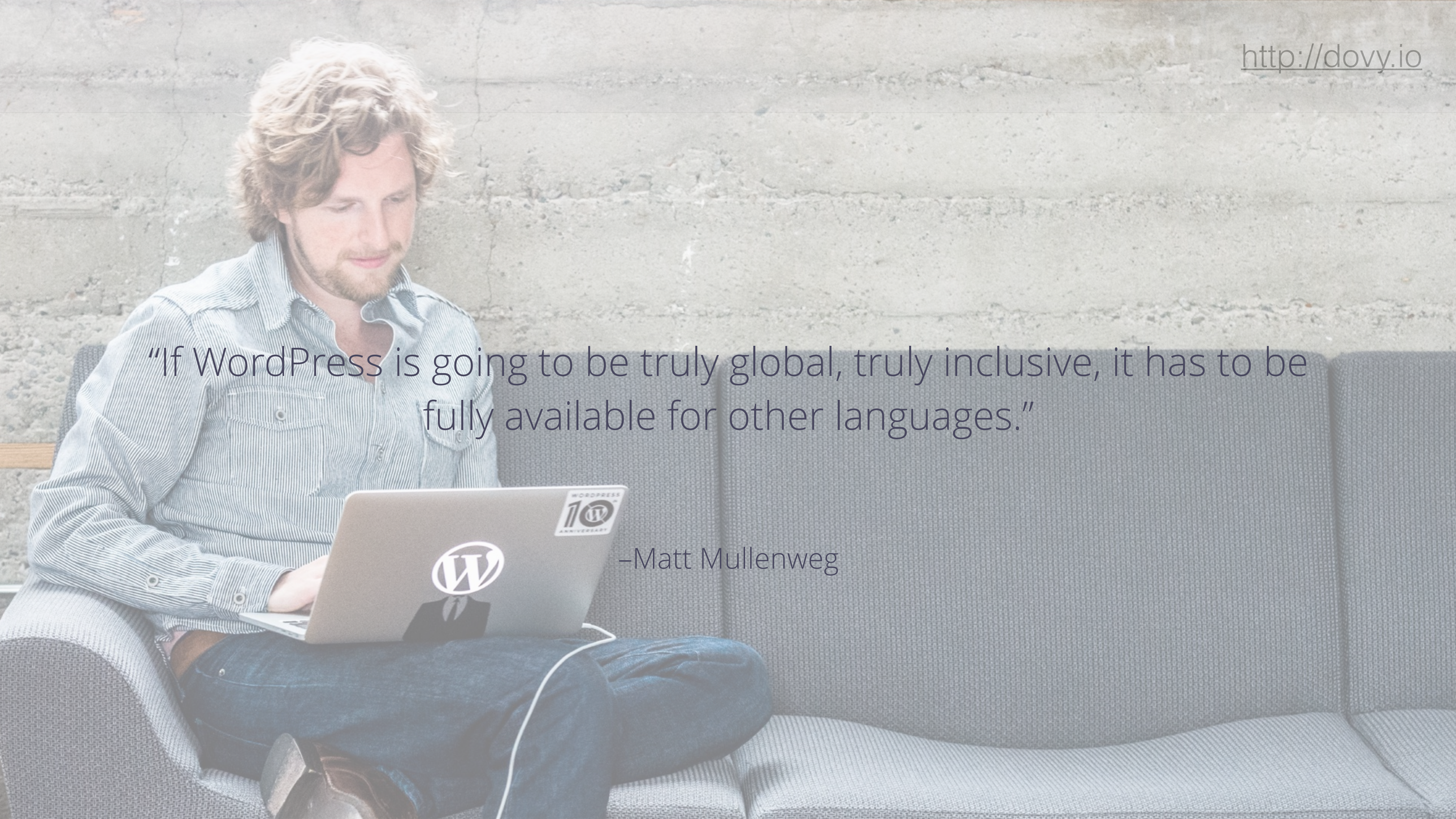
Turning this: “My name is Dovy.”

To this: “Mi nombre es Dovy.”

Or this: “Το όνομά μου είναι Dovy.”

Or even this: “私の名前はDovyです。”

Ooo, how about this: “اسمي دوفي.”



"If WordPress is going to be truly global, truly inclusive, it has to be fully available for other languages."

–Matt Mullenweg

Does your user base?

WordPress

FOR WORDPRESS.ORG DEVS

The WordPress.org repo is now offering free translations for themes, and is slowly rolling the same offering out to all plugin developers. This will be coming out in 2015.

If you do any premium offering, or want more control over your translations, then this presentation is pertinent to you.

POEDIT

Poedit is the best translations editor for translating apps and sites (that use gettext). Thanks to its distraction-free approach, you'll translate faster and easier with it.

GREAT app for editing language files.

PAINFUL to manage many different translations.

http://poedit.net

GLOTPRESS

Open Source

Weak (Improving) Theme

Painful to Setup

Difficult to maintain?

https://github.com/GlotPress/GlotPress

TRANSIFEX

Basis of this presentation.

Free to Open Source

Affordable

https://www.transifex.com/

LOCALIZATION DOESN'T HAVE TO BE HARD

The Transifex localization platform makes it easy to collect, translate and deliver digital content, web and mobile apps in multiple languages.

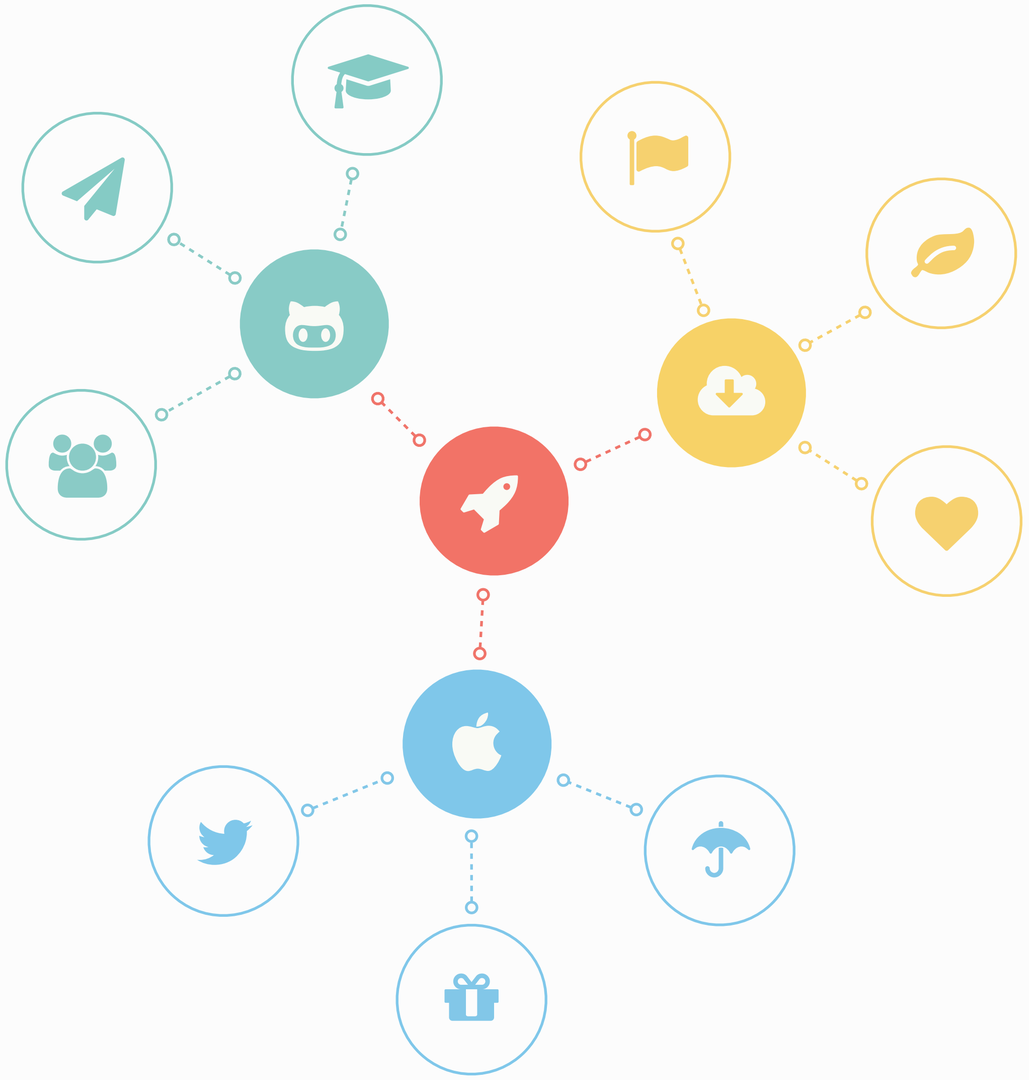
TRY IT FOR FREE

SEE HOW IT WORKS

ORGANIZATIONS GOING GLOBAL WITH TRANSIFEX

Avoid the Pain

Some useful tips to help you get started.

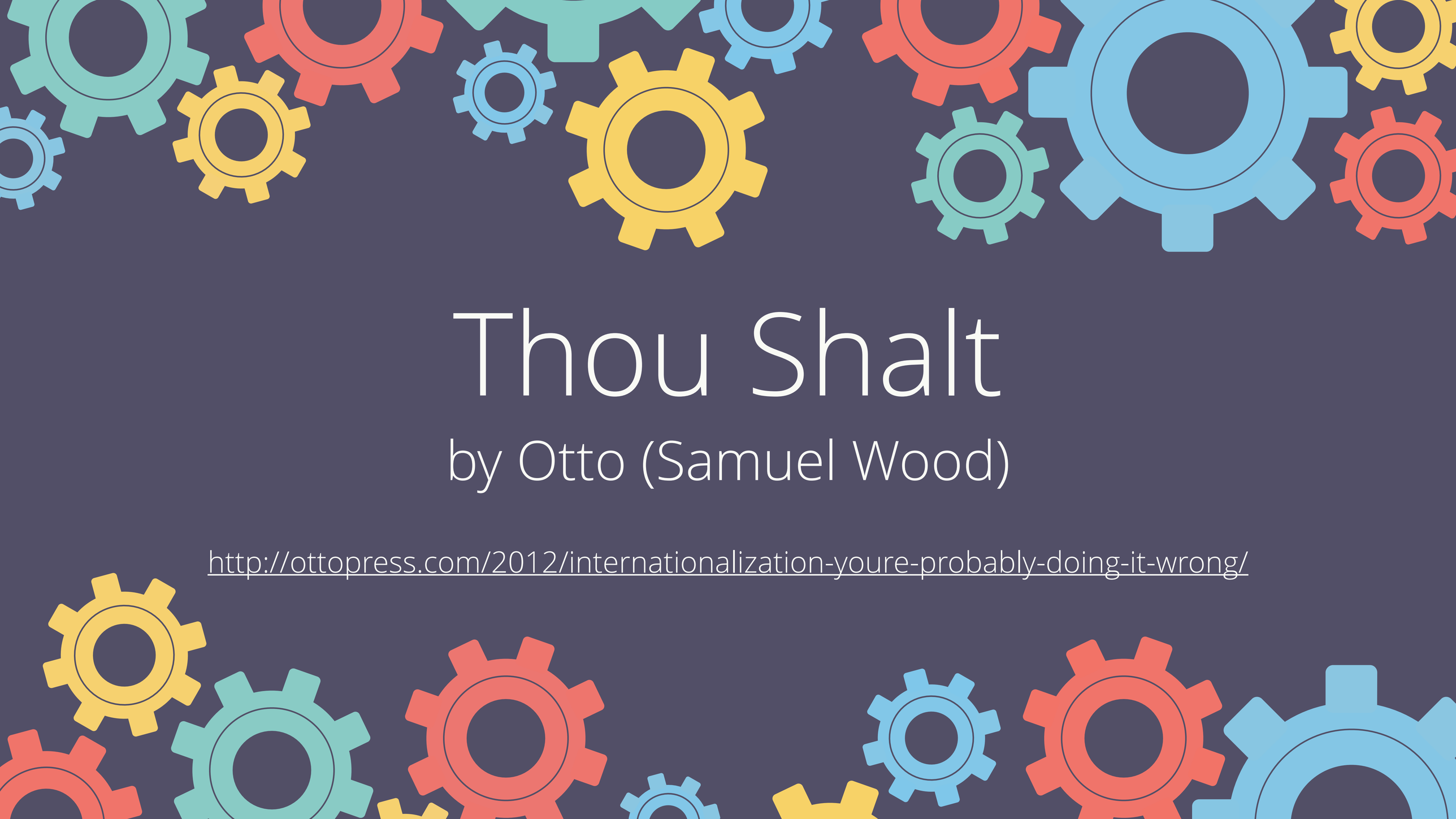


- __() - Return string translation.
- _e() - Same as __() but echo's immedately.
- _x() - Differentiate identical strings in different contexts.
- _ex() - Same as _x() with instant echo.
- _n() - Retrieving the plural or single form based on an amount.
- esc_attr__() - return translated string escaped to be an attribute
- esc_attr_e() - echo translated string escaped to be an attribute
- esc_attr_x() - return translated string with context escaped to be an attribute
- esc_html__() - return translated string escaped to be HTML
- esc_html_e() - echo translated string escaped to be HTML
- esc_html_x() - return translated string with context escaped to be HTML
- n_noop() - Register plural strings in POT file, but don't translate them.
- _nx_noop() - Register plural strings with context in POT file, but don't translate them.
- translate_nooped_plural() - Translate the result of _n_noop() or _nx_noop()

Don’t worry,

there’s a repo for that.

http://dovy.io/



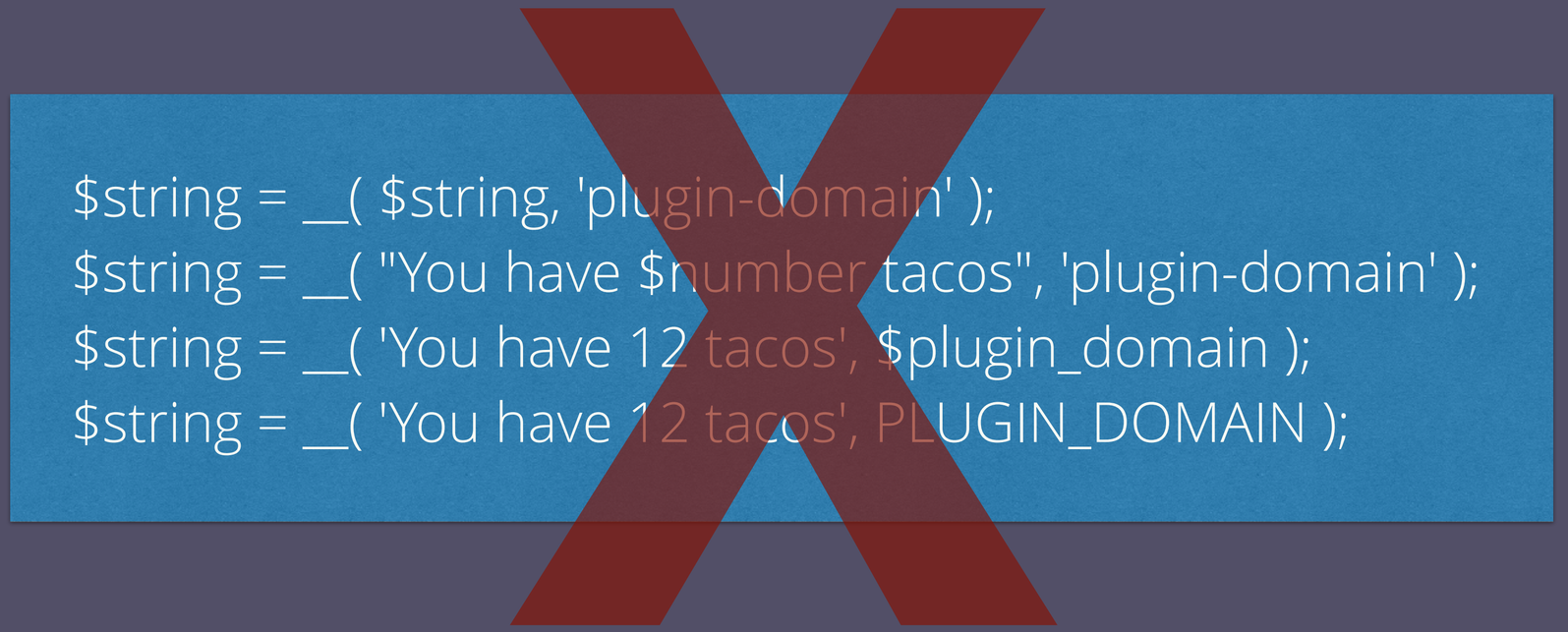
Thou Shalt

by Otto (Samuel Wood)

http://ottopress.com/2012/internationalization-youre-probably-doing-it-wrong/

THOU SHALT #1

NOT use PHP variables of any kind inside a translation function's strings.

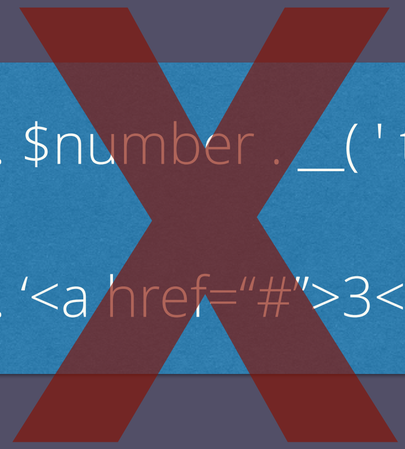


THOU SHALT #2

Always translate phrases and not words.

```
$string = __( 'You have ', ‘plugin' ) . $number . __( ' tacos.’, ‘plugin-domain’ );

$string = __( 'You have ', ‘plugin' ) . ‘<a href=“#”>3</a> ’. __( ' tacos', ‘plugin-domain’ );
```



```
$string = sprintf( __( 'You have %d tacos', ‘plugin-domain' ), $number );

$string = sprintf( _n( 'You have %d taco.', 'You have %d tacos.', $number, ‘plugin-domain' ), $number );
```

THOU SHALT #3

Disambiguate when needed.

```
$string = _x( 'Buffalo', 'an animal', 'plugin-domain' );
$string = _x( 'Buffalo', 'a city in New York', 'plugin-domain' );
$string = _x( 'Buffalo', 'a verb meaning to confuse somebody', 'plugin-domain' );
```

Each string is given a context, so treated as 3 different strings,
not the same one. Context gives way to better translations.

THOU SHALT (4)

NOT put unnecessary HTML markup into the translated string.

```
$string = sprintf( __( '<h3>You have %d tacos</h3>', ‘plugin-domain' ), $number );
```

```
$string = ‘<h3>' . sprintf( __('You have %d tacos', 'plugin-domain'), $number ) . ’</h3>';

$string = sprintf( __( 'You have %s tacos', 'plugin-domain'), ‘<strong>’ . $number . '</strong>' );
```

GET STARTED NOW

This example GitHub repo will:

Guide you through the setup of Transifex.

Generate your translation file (.pot).

Automatically push your language file to Transifex.

Automatically pull from Transifex in the proper format.

Check for missing textdomains.

Generate your project zip for you.

https://github.com/reduxframework/grunt-transifex-wordpress/

GET SOME HELP!

A Great open-source initiative to help translate WordPress plugins. Excellent resource.

http://wp-translations.org/developers-wp-translations/

I worked with FX Bernard to create the first instance of this flow, but he has since taken it to a new level.

Let’s Dig In

https://github.com/reduxframework/grunt-transifex-wordpress/

Questions?