Polylang documentation

Copyright 2011 F. Demarle

Table of contents

Credits

Most of the flags icons included with Polylang are coming from <u>famfamfam</u> and are public domain.

Setting up a WordPress multilingual site with Polylang

Requirements

Polylang requires PHP5 and WordPress 3.1 or higher.

Installation

- Download the latest version of <u>Polylang</u>
- Extract all the files.
- Upload everything (keeping the directory structure) to the `/wp-content/plugins/` directory.
- Activate the plugin through the 'Plugins' menu in WordPress.

Creating the languages

Polylang comes with a list of more than 40 predefined languages. Go to the languages page found in the WordPress settings menu. Choose a language in the dropdown list and click on the 'add language' button.

If you need a language not included in the list, or if you are not happy with the default settings of a language, you can edit it yourself. There are three fields for each language.

- <u>Full name</u>: it is the name which will be displayed on your site. For example: English, Français, Deutsch, Italiano.

- <u>Locale</u>: it is the code used by WordPress for each language. **It is very important to enter the exact WordPress locale for the language**, otherwise the theme translation will not work. It is generally made with two lower case characters, <u>an underscore</u>, and two upper case characters. For example: en_US, fr_FR, de_DE. There are however some locales made with only two lower case characters.

- <u>Language code</u>: this code will be used to build the url. It is best to use the (lower case) 2-characters ISO 639-1 language. For example: en, fr, de.

Don't forget to click on the 'add language' button.

Redo the operation for all languages you need for your site. There is no limit for the number of languages.

Settings

- <u>Default language</u>: it will for example be used when someone visits your homepage for the first time.

- If you are installing Polylang on an existing site, an option will be displayed in red: There are posts, pages, categories or tags without language set. Do you want to set them all to default language ? You can check this to quickly set up the language for all existing posts, pages, categories and post tags. It will also work for custom post types and custom taxonomies. It is important to note that the URL for posts, pages, categories, or tags is not modified.

- Detect browser language: check this option and when someone will visit your homepage for the first time it will be displayed according to his browser preferences. If his browser preferences do not include any language of your site, the default language will be used.

- URL modifications:

  - Keep /language/ in pretty permalinks: the monthly archives URL will look like:

    ```
    http://www.yoursite.com/language/en/2011/11/
    ```

  - Remove /language/ in pretty permalinks: the monthly archives URL will look like:

    ```
    http://www.yoursite.com/en/2011/11/
    ```

  - Hide URL language information for the default language allows you to hide language information only for the default language. In this case the monthly archives URL in the default language will look like:

    ```
    http://www.yoursite.com/2011/11/
    ```

Don't forget to click on the 'save changes' button.

WordPress localization (l10n)

By default, WordPress comes only with English. For all other languages, you need to get the corresponding .mo file. You can find all WordPress language files at: http://svn.automattic.com/wordpress-i18n/

You then have to upload these files to the `/wp-content/languages/` directory.

Theme internationalization (i18n) and localization (l10n)

This step is very important otherwise your theme will not be translated when displaying posts or pages in a language different than your WordPress default language.

Check with the theme author that the theme is internationalized (meaning that it is ready for translation) and localized (meaning translated) to the languages you need. If it's not the case, either you have the possibility to ask him to do the work (great!) or you will have to do it yourself. You can refer to a post I wrote in the forum and of course to the codex.

I18n means editing php files of your theme and requires limited programmer competence. L10n means only creating a new .po and .mo file and requires only linguistic competence. I personally use 'poedit' to do this.

Check that your theme includes the following line (or something similar) in the file header.php:

```
<html <?php language_attributes(); ?>>
```

This line must be placed just after the DOCTYPE line and above the <head> line. It is not mandatory but I understood that some browsers and robots use this line to detect your page language.

Widgets

Go to the widget pages and add the language switcher widget to your sidebar. There are currently four options.

- Displays language names
- Displays flags
- Displays as dropdown

- Forces link to front page

If you uncheck both "Displays language names" and "Displays flags", the widget will display language names. If you check "Displays as dropdown", flags will not be displayed whatever you checked "Display flags" or not. The widget will display languages only if there are existing posts or pages in that languages.

If you check "Forces link to front page", the widget will always send the visitor to the front page in the right language, otherwise it will try to find the translated page (and link to the front page if none was found).

Some flags are supplied with the plugin. However, you may want to use other flags. For each language, the plugin will look for the file locale.png in the 'local_flags' directory. For example, if the locale is en_US, it will look for en_US.png. Don't use the 'flags' directory as your modifications may be overwritten by future releases.

The language switcher will try to find the translated page. If not it will link to the homepage in the right language. A known issue in v0.4.x is that if you check "Display as dropdown", the widget will never look for the translated page and will always link to the homepage in the right language.

For all widgets you have now the possibility to choose to display it for all languages (default) or only one language. Use the language dropdown list at the bottom of the widget options to set up this.

Navigation menus

If your theme supports navigations menus. You can create one menu per language in the Wordpress menus page. Then go the menu tab in the language settings page and affect the right menu to each language.

You have the possibility to add a language switcher at the end of the menu. Just check "Displays a language switcher at the end of the menu". If you uncheck both "Displays language names" and "Displays flags", the language switcher will display language names.

If you check "Forces link to front page", the language switcher will always send the visitor to the front page in the right language, otherwise it will try to find the translated page (and link to the front page if none was found).

Don't forget to click on the 'save changes' button.

Translating categories or post tags

It is very important to set the language for all existing categories and post tags, otherwise they won't be displayed. Translating categories and post tags is however not mandatory.

To set the language, just choose it it the dropdown list below the description field.

For each language, all categories with no translation are available in a dropdown list. Just choose the translation you need if available.

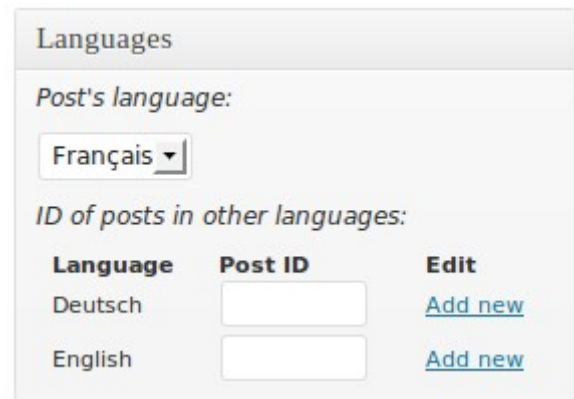The operation for post tags and custom taxonomies is the same.

Translating posts or pages

It is very important to set the language for all existing posts and pages, otherwise they won't be displayed. Translating posts and pages is however not mandatory.

To set the language, just choose it the dropdown list in the Languages box on the top right of the 'edit post' panel.

For each language you have three options:

- Do nothing if you don't want to translate your post.

- Fill the "Post ID" field with the post id if the translation is already existing. To get the id of a post you can for example click on the "Get shortlink" button. You will get something like: `http://www.yoursite.com/wordpress/?p=xxx` where xx is the id of the post your are currently editing.

- If you want to create a new translation, just click on the "Add new" link in the right language.

The operation for pages and custom post types is exactly the same.

Note that the categories are filtered according to the post's language. The same for the tag cloud. If you create new categories or post tags, they will be automatically in the current post's language. However, you will have to go on the Categories (or Post tags) admin panel to set translations if needed.

Choosing the admin language

You can choose the admin language (it's independent of the default language) in the personal options sections of the Users → Your profile settings page. If you don't choose one, the admin will be displayed in the WordPress default language.

Removing Polylang

If you deactivate Polylang, all what you have done is kept in memory in your database. If you delete Polylang using the link in the plugins panel, all your data related to languages and translations will be lost. But all your posts, pages, categories and tags will be kept intact except the language information. It means for example that all your posts will be accessible at the same url as before.

Frequently asked questions

The language filter is lost when using the search form

Your theme uses the template searchform.php (as Twenty Eleven does) or hardcoded the search form and JavaScript is disabled. Unfortunately Polylang currently does not support this. So you have the following alternatives to get the things work well:
- Use the 'get_search_form' function and place your custom search form in functions.php as described in the codex. The plugin also works well if you use the default search form provided by WordPress.

- Enable JavaScript (unfortunately you can't control this for your visitors so the first solution is better)

I activated the plugin and my posts are not displayed any more

You MUST define a language for all your posts and pages otherwise they will not pass the language filter...

I activated the plugin and my categories and post tags are not displayed any more

You MUST define a language for all your categories and post tags otherwise they will not pass the language filter...

Is Polylang compatible with multisite ?

Yes. Since v0.5.

Is it possible to use a static front page ?

Yes. You have to create one front page per language. Set translations as usual and then go to the WordPress 'Reading settings' panel. Check 'Front page displays a static page' and choose one of the page you have just created (the language doesn't matter). Do the same for posts page.

Is it possible to display a language switcher without using the widget ?

It is possible to get a language switcher where you want in your theme without using the widget. For this, you can simply add in your theme the following instruction:

```
pll_the_languages();
```

See the documentation for theme and plugin programmers section for more about the function parameters.

Please don't use anymore:

```
do_action('the_languages', $args);
```

It has been deprecated in v0.5 and may be removed in any next release.

Can I use my own flags for the language switcher ?

Yes. You have to use PNG or JPG files and name them with the WordPress locale. For example, en_US.png. Then upload these files in the `/polylang/local_flags` directory. Don't use the `/polylang/flags` directory as your files would be removed when automatically updating the plugin.

Polylang does not come with a lot flags. Where can I find other flags ?

There are many sources. I included some of the famfamfam flags which I renamed.

How to display the widget flags horizontally ?

The appearance of your site is controlled by CSS rules placed in the file style.css in your theme directory. The appearance of the language switcher is controlled by the classes "lang-item" for all languages and "current-lang-item" for the current languages.

So you can add the following lines in style.css

```
.lang-item {

    display: inline;

}
```

The topic also was discussed on the forum: [Flags vs. Text Links in widget](#)

How to display posts or pages whatever the language ?

This functionality is deactivated by default. Edit your WordPress wp-config.php file. Add the following line somewhere (above /* That's all, stop editing! Happy blogging. */ !)

```
define('PLL_DISPLAY_ALL', true);
```

Now you will have the possibility not to define a language for posts, pages, categories and post tags. These will be displayed whatever the language.

How to know the current language in the theme ?

WordPress provides at least two function for the theme or plugin author to know the current language:

- `get_locale()` returns the WordPress locale in the format `en_US`
- `get_bloginfo('language')` returns the locale in the format `en-US`

Note the difference between '_' and '-' in the two functions.

You can look at the following forum topics:

- [Return the current language as variable for your template](#)
- [How to translate/switch specific contents on templates](#)

API for theme and plugin programmers

Here is a short description of the small API provided with Polylang

pll_the_languages

Displays a language switcher.

Usage: `pll_the_languages($args);`

`$args` is an optional array parameter. Options are:

- `'dropdown'` => displays a list if set to 0, a dropdown list if set to 1 (default: 0)
- `'show_names'` => displays language names if set to 1 (default: 1)
- `'show_flags'` => displays flags if set to 1 (default: 0)
- `'hide_if_empty'` => hides languages with no posts (or pages) if set to 1 (default: 1)
- `'force_home'` => forces link to homepage if set to 1 (default: 0)
- `'echo'` => `echoes if set to 1 (default: 1)`

pll_get_post

Returns the post (or page) translation

Usage: `pll_get_post($post_id, $slug);`

Both parameters are required:

- `'$post_id'` => id of the post you want the translation
- `'$slug'` => 2-letters code of the language

returns the id of the translated post or page as integer.

pll_get_term

Returns the category (or post tag) translation

Usage: `pll_get_term($term_id, $slug);`

Both parameters are required:

- `'$term_id'` => id of the term you want the translation
- `'$slug'` => 2-letters code of the language

returns the id of the translated term as integer.

pll_is_front_page

Acts as WordPress is_front_page function but knows about translated front pages

Usage: `pll_is_front_page();`

returns a boolean.