Polylang documentation

Copyright 2011-2012 F. Demarle

This documentation refers to Polylang v0.8.6 and is licensed under
a Creative Commons Attribution-ShareAlike 3.0 Unported License.

Table of contents

License....3
Credits....3
Setting up a WordPress multilingual site with Polylang....4
- Requirements....4
- Important notice on permalinks....4
- Installation....4
- Creating the languages....4
- Settings....5
- Theme internationalization (i18n) and localization (l10n)....6
- Widgets....6
- Navigation menus....7
- Strings translation....7
- Translating categories or post tags....7
- Translating posts or pages....8
- Synchronization of meta content....8
- Choosing the admin language....8
- Removing Polylang....9

Frequently asked questions....9
- The language filter is lost when using the search form....9
- I activated the plugin and my posts are not displayed any more....9
- I activated the plugin and my categories and post tags are not displayed any more....9
- Is Polylang compatible with multisite ?....9
- Is it possible to use a static front page ?....9
- Is it possible to display a language switcher without using the widget ?....9
- Can I use my own flags for the language switcher ? 10
- Polylang does not come with a lot flags. Where can I find other flags ?....10
- How to display the widget flags horizontally ?....10
- My flags are misaligned in the Twenty Eleven menu....10
- I use an Artisteer generated theme and my menu does not display correctly....10
- How to know the current language in the theme ?....10
- How to make translatable user defined strings in my plugin or theme ?....11
- Not all my widget titles are listed in the strings translation table....11
- I want different featured images on translated posts....11
- I want different page template on translated pages....11
- How to make my custom post types and taxonomies multilingual?....11
- How to add content in a different language than the current one?....11
- Is it possible to query multiple languages?....11
- How to display links to posts translations within the loop?....12
- How to display the default language post if the translation does not exists?....12

API for themes and plugins programmers....12

pll_the_languages....12
pll_current_language....13
pll_get_post....13
pll_get_term....13
pll_is_front_page....13
pll_home_url....14
pll_register_string....14
pll__....14
pll_e....14
Filter : pll_copy_post_metas....14
Filter : pll_the_language_link....15
Filter : pll_flag_title....15
Filter : pll_the_languages filter....16
Filter : pll_get_post_types....16
Filter : pll_get_taxonomies....16
Filter : pll_rewrite_rules....16
List of options which can be set in wp-config.php....17
Option : PLL_FILTER_HOME_URL....17
Option : PLL_SYNC....17

License

Polylang - Copyright 2011-2012 F. Demarle

Polylang is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License, published by the Free Software Foundation, either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Credits

- Most of the flags icons included with Polylang are coming from famfamfam and are public domain.
- Icons used are modifications of Fugue icons by Yusuke Kamiyamane. The icons are licensed under Creative Common Attribution 3.0 License. Thanks to Randy Jensen !
- Wherever third party code has been used, credit has been given in the code's comments.
- Thanks to translators:
  - German contributed by Christian Ries
  - Russian contributed by yoyurec
  - Greek contributed by theodotos
  - Dutch contributed by AlbertGn
  - Hebrew contributed by ArielK
  - Polish contributed by Peter Paciorkiewicz
  - Latvian contributed by @AndyDeGroo
  - Italian contributed by Luca Barbetti
  - Danish contributed by Compute
  - Spanish contributed by Curro
- Special thanks to RavanH and @AndyDeGroo who intensively tested, helped debugging and improved Polylang.

Setting up a WordPress multilingual site with Polylang

Requirements

Polylang requires PHP5 and WordPress 3.1 or higher.

Important notice on permalinks

When using the non default permalinks, pages (and posts as well if your permalink structure is /%postname%/), must not have a language code as permalink. For example if you set the English language, don't use the permalink `http://www.yoursite.com/en/` for a page (or post), otherwise you may get unexpected results !

Installation

- Download the latest version of Polylang
- 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 50 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 five fields for each language.

- Full name: it is the name which will be displayed on your site. For example: English, Français, Deutsch, Italiano.
- Locale: 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, an underscore, and two upper case characters. For example: en_US, fr_FR, de_DE. There are however some locales made with only two or three lower case characters. For example: ar, el, ckb.
- Language code: 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.
- Text direction: most languages are written left to right, but some are written right to left (Arabic, Hebrew and some others). Just choose the right option for your language. Polylang will thus inform the theme it has to load the LTR or RTL style. Note that not all themes are compatible with both text directions (of course Twenty Ten and Twenty eleven are !)
- Order: it is possible to choose the order in which the language will be displayed in the language switcher

Don't forget to click on the 'add language' button and redo the operation for all languages you need for your site. There is no limit for the number of languages.

By default, WordPress comes only with English. For all other languages, Polylang will try to automatically download corresponding WordPress language files. If, for some reason, it does not succeed, you can try to download them at: `http://svn.automattic.com/wordpress-i18n/`. You then would have to upload these files to the `/wp-content/languages/` directory.

Polylang will also try to download Twenty Ten and Twenty Eleven language files. If for some reason it does not work, you can download them at the same address as above and upload the files to `/wp-content/themes/twentyten/languages/` or `/wp-content/themes/twentyeleven/languages/` depending on the theme you are using.

Settings

- Default language: 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/`
  - Add language information to all URL (not recommended). The option has no effect if you don't use pretty permalinks. For example, the pages URL will look like:

    `http://www.yoursite.com/en/your_page_title/`
  - Redirect the language page to the homepage. The option has no effect if you don't use a static front page. For example, the home URL will look like:

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

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

Important notice

I do not recommend to add the language information to all URL. In fact, most plugins similar to Polylang do add the language code to all URL. There is no issue when the plugin is working well. However, if for some reason the plugin is de-activated, every URL loose the language information and thus all external links are broken. Polylang was first designed to avoid this as much as possible and so, by default, adds the language information to the URL only when it is needed (mainly the archives) and does not add it when it is not necessary (posts, pages, categories, post tags). Thus, if the plugin is de-activated, most importants links will not break.

However, since quite a lot of users requested the feature (which can be useful when migrating from another plugin), it was then decided to add the option in version 0.7.

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.

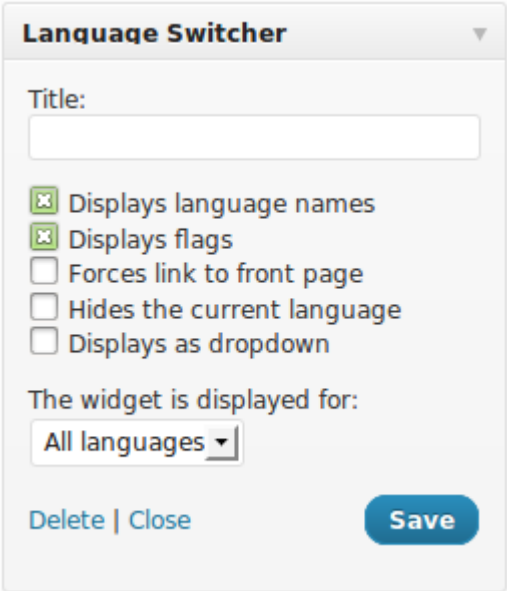
Check that your theme includes the following line (or something similar) in the file header.php:

```
<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 five options:



- Displays language names
- Displays flags
- Forces link to front page
- Hides the current language
- Displays as dropdown

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.

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).

If you check “Hides the current language”, the widget will never display the current language.

Whatever the chosen options, the widget will display a language only if at least one post or one page has been published in this language.

Some flags are supplied with the plugin. However, you may want to use your own flags. For each language, the plugin will look for the file locale.png in the 'wp_content/polylang' directory. For example, if the locale is en_US, it will look for en_US.png. **Don't use the 'polylang/flags' directory as your modifications may be overwritten when updating Polylang**.

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

Your theme must support custom navigation menus for the following to work (for programmers, it means that the theme must call 'register_nav_menu') and must not hardcode the menu name in the function 'wp_nav_menu', i.e.

`wp_nav_menu(array('theme_location' => 'primary'));` is OK

`wp_nav_menu(array('menu' => 'nav menu'));` is not OK

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).

If you check “Hides the current language”, the language switcher will never display the current language.

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

Strings translation

Polylang allows translating user defined strings such as site title, tagline, and widget titles. Plugins may add other strings to this list.

The table has 3 columns.

- Name: It's provided for sorting convenience
- String: the string to translate. For, example the site title is displayed how it was entered in the “General settings” panel
- Translations: Here you can translate the string in all defined languages.

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

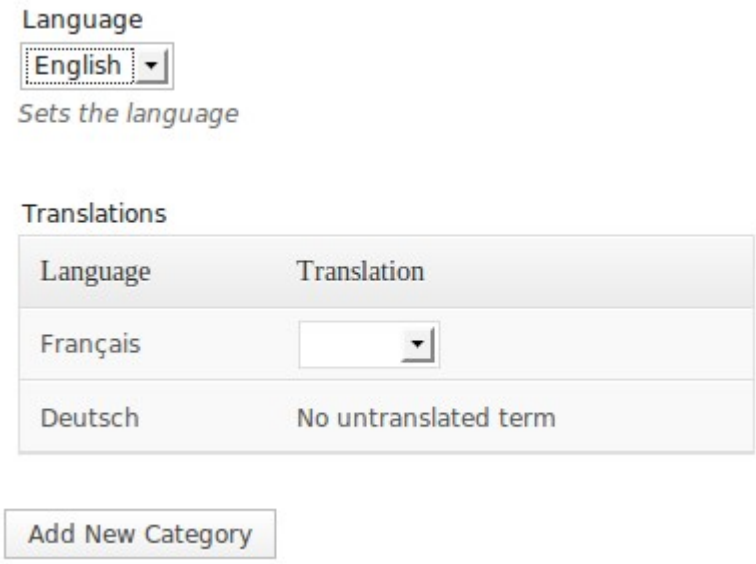
Translating categories or post tags

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 in the dropdown list below the description field.

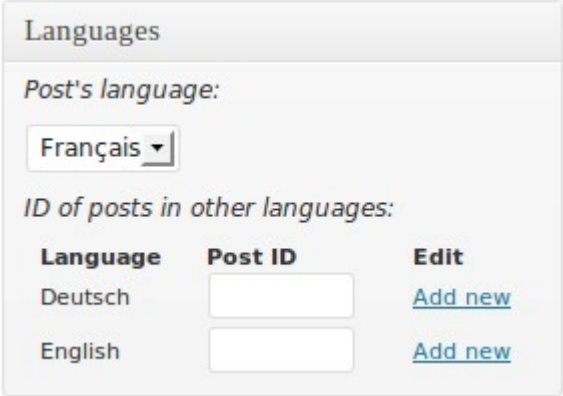
For each language, all categories with no translation are available in a dropdown list. Just choose the translation you need if available.

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=xx` 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. If translated categories and post tags are available, they will be automatically set in the post translation. The same for page parent.

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.

Synchronization of meta content

Polylang does synchronizes some meta content between translations of posts or pages. The default synchronized meta contents are:

- Categories and post tags (as well as other custom taxonomies) when they are translated
- Featured image and page template
- Post format
- Page hierarchy

Although this feature will appears nice to some users, not everyone will be happy with it. You can stop this synchronization by adding this line in the file wp-config.php:

```
define('PLL_SYNC', false);
```

*M*ore information is available in the `option section`.

The `FAQ` can help you to unsynchronize the feature image or the page template only. Polylang does also offers the possibility to `synchronize your own custom fields`.

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 delete 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:

```
<ul><?php pll_the_languages();?></ul>
```

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 `/wp-content/polylang/` directory (create the directory if it does not exist). 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 language.

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

My flags are misaligned in the Twenty Eleven menu

If you are using the flags provided with Polylang, just add the following css rule to your style.css:

```
#access img {
      margin-bottom: 0px;
}
```

I use an Artisteer generated theme and my menu does not display correctly

The Artisteer themes do not use the standard way to create navigation menus and need some modifications to be compatible with Polylang. Look at the following forum topics:

- Modify Artisteer themes for page menus
- Modify Artisteer themes for custom menus

How to know the current language in the theme ?

WordPress provides at least two function for the themes or plugins authors 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`

How to make translatable user defined strings in my plugin or theme ?

You have to register strings on the admin side with the function `pll_register_string` and display them on frontend with `pll__` or `pll_e`.

Not all my widget titles are listed in the strings translation table

If you enter a title in the widget options and this title is not listed in the translation table, it is most likely because the widget uses the old Widget API (WP < 2.8). Obviously the advantage for the widget is to be compatible with WordPress older than 2.8 but unfortunately, there is no way to make Polylang automatically compatible with all widgets which use this old API.

It is however possible to make them compatible for people who have some PHP skills, using the Polylang API functions `pll_register_string` and `pll__`.

I want different featured images on translated posts

See `Changing back to using different featured images`.

I want different page template on translated pages

See `Changing back to using different page templates`.

How to make my custom post types and taxonomies multilingual?

You must register your custom post type (or taxonomy) in a function hooked to the 'init' action as usual.

Then you may need to register your post type or taxonomy into multilingual post types and taxonomies using the `pll_get_post_types` or `pll_get_taxonomies` filters. You do not need to use this filter if you set the parameter `show_ui` to `true`.

Finally, if you use labels which needed to be translated on the frontend, you will need to register your post type or taxonomy a second time in a function hooked to the 'wp' action.

How to add content in a different language than the current one?

Polylang does set the language of the theme based on the main query, but it is possible to add content in a different language. For example, you can display the list of the five most recent French post on an English page. Your custom query just needs to add the 'lang' parameter.

Example:

```
$posts = get_posts(array(
      'post_type' => 'post',
      'lang' => 'fr', // use language slug in the query
      'showposts' => 5
));
```

Is it possible to query multiple languages?

Yes. Example:

```
$posts = get_posts(array(
      'post_type' => 'post',
```

```
	'lang' => 'de,fr', // query German and French posts
	'showposts' => 5
));
```

How to display links to posts translations within the loop?

Example:

```
<?php while ( have_posts() ) : the_post(); ?>
<ul class='translations'>
	<?php pll_the_languages(array('post_id' => $post->ID)); ?>
</ul>
<?php the_content(); ?>
<?php endwhile; ?>
```

How to display the default language post if the translation does not exists?

Example of modified loop to get this result (supposing the default language is English):

```
<?php // force querying the English posts
query_posts(array('post_type' => 'post','lang' => 'en')); ?>
if (have_posts()) : ?>
	<?php while (have_posts()) : the_post(); ?>
	<?php global $post;
	// get translated post (in current language) if exists
	if($post_id = pll_get_post($post->ID, pll_current_language())) {
		$post = get_post($post_id);
		setup_postdata($post);
	}?>
... do what ever you want in your loop ...
```

API for themes and plugins 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)
- `'display_names_as'` => either 'name' or 'slug' (default: 'name')
- `'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, returns a string if set to 0 (default: 1)
- `'hide_if_no_translation'` => hides the language if no translation exists if set to 1 (default: 0)
- `'hide_current'`=> hides the current language if set to 1 (default: 0)
- `'post_id'`=> if set, displays links to translations of the post (or page) defined by post_id (default: null)

Starting from v0.6, you have to output yourself the ul tags if you don't use the dropdown option. Examples:

```
<ul><?php pll_the_languages(); ?></ul>
<?php pll_the_languages(array('dropdown'=>1));?>
<ul><?php pll_the_languages(array('show_flags'=>1,'show_names'=>0));?></ul>
```

pll_current_language

Returns the current language

Usage: `pll_current_language($value);`

- `'$value'` => (optional) either 'name' or 'locale' or 'slug', defaults to 'slug'

returns either the full name, or the WordPress locale (just as the WordPress core function 'get_locale' or the slug (2-letters code) of the current language.

pll_get_post

Returns the post (or page) translation

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

- `'$post_id'` => (required) id of the post you want the translation
- `'$slug'` => (optional) 2-letters code of the language, defaults to current 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);`

- `'$post_id'` => (required) id of the post you want the translation
- `'$slug'` => (optional) 2-letters code of the language, defaults to current language

returns the id of the translated term as integer.

pll_is_front_page

Removed in v0.8.2 as it is now useless. Use the standard WP `is_front_page` function instead.

pll_home_url

Returns the home page url in the current language

Usage: `pll_home_url();`

returns the url of the homepage in the current language, as a string.

pll_register_string

Allows plugins to add their own strings in the “strings translation” panel. The function must be called on admin side (the functions.php file is OK for themes).

Usage: `pll_register_string($name, $string);`

Both parameters are required:

- `'$name'` => name provided for sorting convenience (ex: 'myplugin')
- `'$string'` => the string to translate

Examples available on the forum:

- Adapt Atahualpa theme
- Adapt Ultimate taxonomy manager plugin

pll__

translates a string previously registered with pll_register_string

Usage: `pll__($string);`

The unique parameter is required:

- `'$string'` => the string to translate

returns the translated string.

pll_e

Echoes a translated string previously registered with pll_register_string

Usage: `pll_e($string);`

The unique parameter is required:

- `'$string'` => the string to translate

Filter : pll_copy_post_metas

Allows plugins to copy post metas when a new post (or page) translation is created.

Example:

```
add_filter('pll_copy_post_metas', 'copy_post_metas');
function copy_post_metas($metas) {
	return array_merge($metas, array('my_post_meta'));
}
```

Important note: Currently the synchronization process does not support post metas with a multiple

values

Filter : pll_the_language_link

Allows plugins to modify the translation link outputed by the language switcher (valid for the widget, the nav menus and the function pll_the_languages).

Filter arguments:

- `$url` => the translation URL to modify (set to null if there is no translation available)
- `$slug` => the 2-letters language iso-code of the translation
- `$locale` => the WordPress locale for the language of the translation

Example (to link to the posts list in the right language when there is no translation available):

```
add_filter('pll_the_language_link', 'my_link', 10, 2);
function my_link($url, $slug) {
	return $url === null ? home_url('?lang='.$slug) : $url;
}
```

Filter : pll_flag_title

Allows plugins to modify the title attribute of the flag in the language switcher (defaults to language name).

Filter arguments:

- `$title` => the title attribute
- `$slug` => the 2-letters language iso-code of the translation
- `$locale` => the WordPress locale for the language of the translation

Example:

```
add_filter('pll_flag_title', 'my_flag_title', 10, 2);
function my_flag_title($title, $slug) {
	switch($slug) {
		case 'en':
			return 'The blog in '.$title;
			break;
		case 'fr':
			return 'Le blog en '.$title;
			break;
		default:
			return $title;
			break;
	}
}
```

Filter : pll_the_languages filter

Allows plugins to modify the whole output of the language switcher.

Filter arguments:

- `$output` => the output of the language filter
- `$args` => the arguments of the function pll_the_languages

Example (to add a link to the flag file in the title attribute of the dropdown list which maybe useful used together with `a script adding a picture in the dropdown`):

```
add_filter('pll_the_languages', 'my_dropdown', 10, 2);
function my_dropdown($output, $args) {
	if (!$args['dropdown'])
		return $output;
	foreach (array('en_US', 'fr_FR', 'de_DE') as $lang) {
		$file = POLYLANG_DIR."/flags/$lang.png";
		$value = reset(explode('_', get_locale()));
		$output = str_replace("value='$value'",
			"value='$lang' title='$file'", $output);
	}
	return $output;
}
```

Filter : pll_get_post_types

Allows plugins to modify the list of post types which will be filtered by language. The default are post types which have the parameter show_ui set to true (which include posts and pages). The filter must be called after registering the post type in an init action.

Filter arguments:

- `$post_types` => the array of post type names

Filter : pll_get_taxonomies

Allows plugins to modify the list of taxonomies which will be filtered by language. The default are taxonomies which have the parameter show_ui set to true (which include categories and post tags). The filter must be called after registering the taxonomy in an init action.

Filter arguments:

- `$taxonomies` => the array of taxonomy names

Filter : pll_rewrite_rules

Allows plugins to inform Polylang of a new rewrite rules filter to use (allowing this rules to be filtered by language. This must be used

Filter arguments:

- `$rules` => the array rewrite rules to filter

Example:

```
// create rewrite rules with a filter 'something_rewrite_rules'
add_filter('generate_rewrite_rules', 'my_rules');
function my_rules($wp_rewrite) {
	$rules[] = ...
	$rules = apply_filters('my_own_rewrite_rules', $rules);
	$wp_rewrite->rules = array_merge(rules, $wp_rewrite->rules);
}
// add the filter (without '_rewrite_rules') to the Polylang list
add_filter('pll_rewrite_rules',
	create_function('$rules',
	'return array_merge($rules, array(\'my_own\'));'));
```

List of options which can be set in wp-config.php

It is possible to change the default way Polylang works by setting some options in wp-config.php. You can add a line such as:

```
define(OPTION, false);
```

above the comment:

```
/* That's all, stop editing! Happy blogging. */
```

Alternatively, it is possible to create a file named `pll-config.php` in the directory `/wp-content/polylang/` (you have to create the directory too) and set the options in this file instead of `wp-config.php`.

Option : PLL_FILTER_HOME_URL

Defaults to true.

This option allows Polylang to return the url of the homepage in the right language instead of the default homepage. Although the feature allows Polylang working with a lot of themes without any modification, it may break some themes. If you set it to false, you will need to modify your theme and replace calls to `home_url()`, `home_url('/')`, `bloginfo('url')`, `get_bloginfo('url')` by `pll_home_url()` if you want a link to the homepage in the right language.

Option : PLL_SYNC

Defaults to true.

This option allows Polylang to synchronize categories, post tags, featured image, post format, page template, page hierarchy, category hierarchy between translations. When set to false, the synchronization process is disabled, although Polylang still copy these meta when creating a translation.