

WordPress

Type de post personnalisé

- **Enregistrement**
- **Utilisation avec l'interface d'administration**
- **Complément avec des champs personnalisés**

Type de post personnalisé: introduction

WordPress ne propose que deux types de contenu, les articles et les pages.

Pour chacun d'entre eux, vous disposez de deux champs de saisie : le titre et le contenu.

Les articles bénéficient en plus de deux facilités de taxonomie (classification), ils peuvent être rattachés à :

- une ou plusieurs catégories, organisées de manière hiérarchique dans une arborescence (taxonomie hiérarchique),
- une liste de mots-clés ou tags (taxonomie non hiérarchique).


Pour étendre ces fonctionnalités de base de WP, vous pouvez créer vos propres types de contenu sous forme de types de posts personnalisés, avec leurs taxonomies dédiées.

Remarque : Une taxonomie est un moyen de classer et grouper les articles ou les types de posts personnalisés.

Une taxonomie permet à l'auteur de catégoriser ses publications et au lecteur de pouvoir facilement rechercher et trier les posts.

Type de post personnalisé: enregistrement

- Un type de post personnalisé (custom post type) doit être enregistré (register) à chaque traitement d'une URL, pour être utilisé dans l'interface d'administration et affiché dans un thème.
- Bien qu'il soit possible de l'enregistrer dans le fichier functions.php d'un thème, WP préconise d'enregistrer un type de post personnalisé dans une extension :

 We recommend that you put custom post types in a plugin rather than a theme. This ensures that user content remains portable even if they change their theme.


- Fonction d'enregistrement d'un TPP :

```
register_post_type( string $post_type, array $args );
```

It is important that you prefix your post type functions and identifiers with a short prefix that corresponds to your plugin, theme, or website.

par exemple: `register_post_type('n41_book', $args);`

- Exécution de cette fonction d'enregistrement d'un TPP : **action hook 'init'**

 You must call `register_post_type()` before the `admin_init` and after the `after_setup_theme` action hooks. A good hook to use is the `init` action hook.

TPP: enregistrement, cas pratique

Enregistrer un TPP [n41_book](#) pour le site WP d'un éditeur de livres.

Ce TPP permettra de publier des posts de livres et d'utiliser des taxonomies associées aux livres pour les classer selon le genre, l'auteur et la langue d'origine.

Utiliser pour cela une extension [n41-editions](#), dans laquelle tous les libellés pour la gestion du TPP seront traduits en français :

```
<?php
/*
Plugin Name: Les Éditions N41
Plugin URI: https://n41.plugins.com
Description: Le catalogue des livres des Éditions N41.
Version: 1.0
Author: N41
Author URI: https://n41.com
*/
```

TPP: enregistrement, cas pratique

1. Coder dans le plugin une fonction qui va regrouper l'enregistrement du TPP et des taxonomies qui lui sont associées , puis l'exécution de cette fonction par le crochet d'actions 'init' :

```
/**
 * Enregistre un post personnalisé appelé "n41_book".
 *
 * @see get_post_type_labels() for label keys.
 */
function n41_editions_register_book_post_type_and_taxonomies() {

}

add_action( 'init',
            'n41_editions_register_book_post_type_and_taxonomies' );
```

TPP: enregistrement, cas pratique

2. Dans la fonction `n41_editions_register_book_post_type_and_taxonomies()`, initialiser le tableau `$labels` des libellés utilisés pour la gestion du TPP :

```
$labels = array(
    'name' => __('Books', 'n41-editions'),
    'singular_name' => __('Book', 'n41-editions'),
    'add_new' => __('Add New', 'n41-editions'),
    'add_new_item' => __('Add New Book', 'n41-editions'),
    'edit_item' => __('Edit Book', 'n41-editions'),
    'new_item' => __('New Book', 'n41-editions'),
    'view_item' => __('View Book', 'n41-editions'),
    'view_items' => __('View Books', 'n41-editions'),
    'search_items' => __('Search Books', 'n41-editions'),
    'not_found' => __('No books found.', 'n41-editions'),
    'not_found_in_trash' => __('No books found in Trash.', 'n41-editions'),
    'parent_item_colon' => __('Parent Book:', 'n41-editions'),
    'all_items' => __('All Books', 'n41-editions'),
    'archives' => __('Book Archives', 'n41-editions'),
    'attributes' => __('Book Attributes', 'n41-editions'),
    'insert_into_item' => __('Insert into book', 'n41-editions'),
    'uploaded_to_this_item' => __('Uploaded to this book', 'n41-editions'),
    'featured_image' => __('Book Cover Image', 'n41-editions'),
    'set_featured_image' => __('Set cover image', 'n41-editions'),
    'remove_featured_image' => __('Remove cover image', 'n41-editions'),
    'use_featured_image' => __('Use as cover image', 'n41-editions'),
    'filter_items_list' => __('Filter books list', 'n41-editions'),
    'items_list_navigation' => __('Books list navigation', 'n41-editions'),
    'items_list' => __('Books list', 'n41-editions')
);
```

TPP: enregistrement, cas pratique

3. Dans la fonction `n41_editions_register_book_post_type_and_taxonomies()`, initialiser le tableau d'arguments `$args`, passé à la fonction `register_post_type()` :

```
$args = array(
    'labels'           => $labels, // tableau des libellés pour la gestion du TPP
    'description'      => __('Book catalog', 'n41-editions'),
    'public'           => true,
    'hierarchical'     => false,
    'publicly_queryable' => true,
    'show_ui'          => true,
    'show_in_menu'     => true,
    'show_in_nav_menus' => true,
    'show_in_admin_bar' => true,
    'menu_position'    => null,
    'capability_type'  => 'post',
    'supports'         => array('title', 'editor', 'author', 'thumbnail', 'excerpt',
                                'comments', 'revisions', 'custom-fields'),
    'taxonomies'       => array('genre', 'author', 'original_language'),
    'has_archive'      => true,
    'rewrite'          => array('slug' => 'book'),
    'query_var'        => true
);
```

En bleu, la liste des fonctionnalités prises en charge par WP (supports).

En rouge, la liste des taxonomies qui devront être enregistrées par la fonction `register_taxonomy()`.

TPP: enregistrement, cas pratique

4. Dans la fonction `n41_editions_register_book_post_type_and_taxonomies()`, ajouter l'exécution de `register_post_type('n41_book', $args)` puis enregistrer les taxonomies énumérées dans `$args`, avec `register_taxonomy()` :

```
register_post_type( 'n41_book', $args );

register_taxonomy( 'n41_genre',
                  'n41_book',
                  array( 'labels'          => array('name'          => __('Genres', 'n41-editions'),
                                                         'singular_name' => __('Genre', 'n41-editions')),
                        'public'          => true,
                        'hierarchical'    => true,
                        'rewrite'         => true )
);

register_taxonomy( 'n41_author',
                  'n41_book',
                  array( 'labels'          => array('name'          => __('Authors', 'n41-editions'),
                                                         'singular_name' => __('Author', 'n41-editions')),
                        'public'          => true,
                        'hierarchical'    => true,
                        'rewrite'         => true )
);

register_taxonomy( 'n41_original_language',
                  'n41_book',
                  array( 'labels'          => array('name'          => __('Original languages', 'n41-editions'),
                                                         'singular_name' => __('Original language', 'n41-editions')),
                        'public'          => true,
                        'hierarchical'    => true,
                        'rewrite'         => true )
);
```

Taxonomies hiérarchisées, gestion similaire à la taxonomie 'category' d'un post article.

https://developer.wordpress.org/reference/functions/register_taxonomy/

TPP: enregistrement, cas pratique

5. Traduire les libellés utilisés dans le code de l'extension,
dans `wp-content/plugins/n41-editions/languages/ n41-editions-fr_FR .po` :

```
msgid "Book"
msgstr "Livre"

msgid "Books"
msgstr "Livres"

msgid "Add New"
msgstr "Ajouter"

msgid "Add New Book"
msgstr "Ajouter un livre"

msgid "Edit Book"
msgstr "Editer un livre"

msgid "New Book"
msgstr "Nouveau livre"

msgid "View Book"
msgstr "Voir un livre"

etc.
```

TPP: enregistrement, cas pratique

6. Générer le fichier binaire **n41-editions-fr_FR .mo** utilisé par WP, à partir du fichier **n41-editions-fr_FR .po**, avec le logiciel **Poedit**.
7. Dans le fichier principal de l'extension **n41-editions.php**, ajouter le code du chargement du fichier des traductions pour la langue courante (fr_FR).

```
/**
 * Charge les traductions de l'extension
 *
 */
function n41_editions_load_plugin_textdomain() {
    load_plugin_textdomain( 'n41-editions',
                           FALSE, basename( dirname( __FILE__ ) ) . '/languages/' );
}
add_action( 'plugins_loaded', 'n41_editions_load_plugin_textdomain' );
```

Utiliser la fonction **load_plugin_textdomain()** et la déclencher avec le hook **'plugins_loaded'**.

TPP: utilisation, cas pratique

8. Utilisation du TPP 'n41_book' enregistré, avec l'interface d'administration :

The screenshot displays the 'Edit book' interface for the book 'Le Joueur d'échecs'. The interface is divided into a left sidebar, a main content area, and a right sidebar.

Left Sidebar: Contains navigation links for 'Tableau de bord', 'Articles', 'Médias', 'Pages', 'Commentaires', 'Livres' (highlighted), 'Tous les livres', 'Ajouter', 'Genres', 'Auteurs', 'Langues d'origine', 'Apparence', 'Extensions', 'Utilisateurs', 'Outils', 'Réglages', 'N41 Recipes', and 'Réduire le menu'.

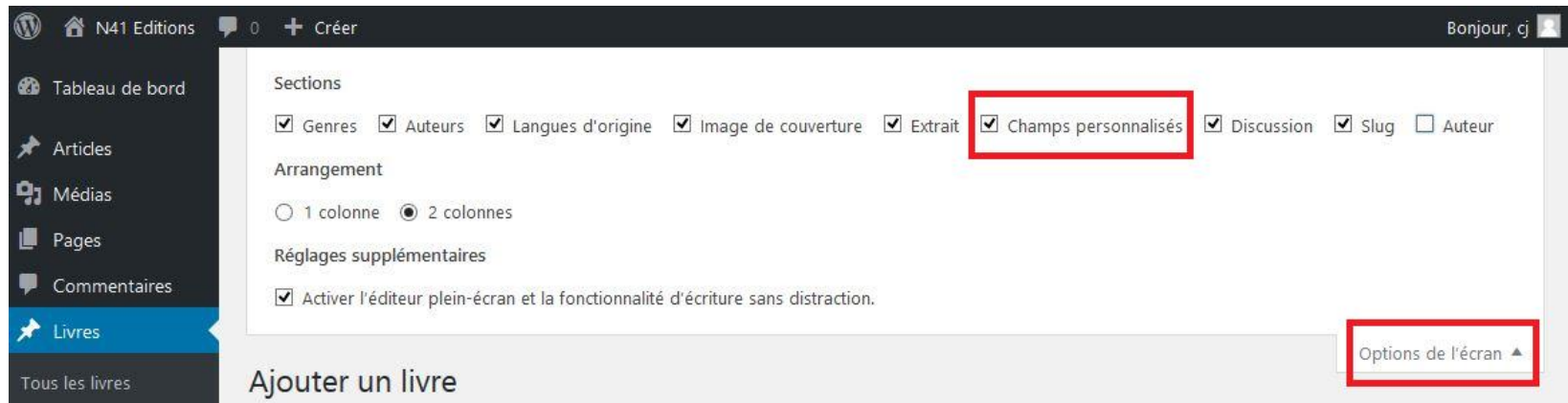
Main Content Area: The title 'Le Joueur d'échecs' is at the top. Below it is the 'Permalien' (http://localhost/wordpress/book/le-joueur-dechecs/) and a 'Modifier' button. A 'Ajouter un média' button is also present. The text editor shows three paragraphs of text, with the first paragraph starting with 'Czentowic, champion d'échecs arrogant, esprit borné à outrance, inculte et étonnamment stupide, occupe le premier plan jusqu'à l'entrée en scène de Monsieur B. Dès lors que cet aristocrate autrichien s'intéresse à la partie livrée entre le champion et les passagers amateurs, la direction du texte bascule.' The text is formatted with bold, italic, and list markers. The bottom of the main area shows 'Nombre de mots : 164' and 'Dernière modification par cj, le 20 septembre 2018 à 19 h 25 min'.

Right Sidebar: Contains metadata fields. 'Publier' is set to 'Publier'. 'Genres' includes 'Roman' (checked) and a link to '+ Ajouter une nouvelle catégorie'. 'Auteurs' includes 'Stefan Zweig' (checked) and a link to '+ Ajouter une nouvelle catégorie'. 'Langues d'origine' includes 'Allemand' (checked) and 'Français' (unchecked), with a link to '+ Ajouter une nouvelle catégorie'. 'Image de couverture' shows a book cover for 'Stefan Zweig Le Joueur d'échecs'.

TPP: ajout de champs personnalisés, cas pratique

9. Ajouter des champs personnalisés, avec l'interface d'administration; ces champs sont des données supplémentaires d'un post, et sont stockés dans la table `postmeta` :

Valider au préalable l'affichage des champs personnalisés sur la page d'ajout ou de modification d'un post, ici le post personnalisé `n41_book` :



TPP: ajout de champs personnalisés, cas pratique

9. Ajout de champs personnalisés, avec l'interface d'administration; ces champs sont des données supplémentaires d'un post et sont stockés dans la table [postmeta](#) :

Dans la zone de gestion des champs personnalisés d'un post vous pouvez :

Champs personnalisés

Nom	Valeur
date publication	2006-03-01
<input type="button" value="Supprimer"/>	
<input type="button" value="Mettre à jour"/>	

Ajouter un nouveau champ personnalisé :

Nom	Valeur
<input type="text" value="— Sélectionner —"/>	
Saisissez-en un nouveau	
<input type="button" value="Ajouter un champ personnalisé"/>	

Les champs personnalisés peuvent être utilisés pour ajouter des métadonnées supplémentaires à un article, que vous pouvez ensuite [utiliser dans votre thème](#).

- ① Modifier la valeur d'un champ personnalisé déjà affecté à ce post. Ici [date publication](#) du TPP [n41_book](#).
- ② Affecter à ce post un champ personnalisé déjà créé et lui donner une valeur.
- ③ Saisir un nouveau champ personnalisé et l'affecter à ce post avec une valeur.

Les afficher dans les templates du thème avec [the_meta\(\)](#) ou [get_post_meta\(\)](#)