

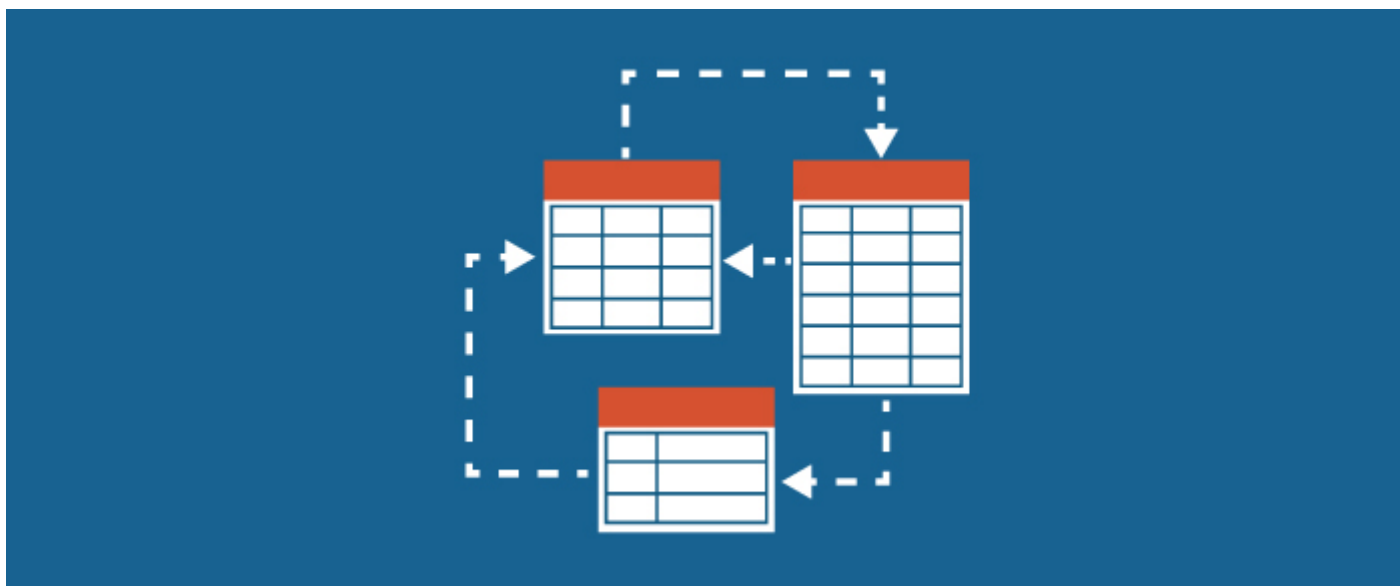
## Come filtrare gli articoli di WordPress per tassonomia personalizzata

*lunedì, 16 Aprile 2018*

Tra le impostazioni predefinite di WordPress c'è la possibilità di filtrare l'elenco degli articoli nel backend tramite la tassonomia di default, "Categorie", oppure tramite data, con un semplice menu a tendina. Ma ti è mai capitato di aver registrato una tassonomia personalizzata (Custom taxonomy)? Chi lo ha già fatto, si sarà reso conto che la nuova tassonomia non è presente tra le possibilità di filtro nella pagina di backend relativa all'elenco articoli.

Ma non sarebbe comodo poter filtrare i tuoi articoli anche con questa nuova tassonomia, creando magari filtri combinati anche con quello delle categorie presente di default?

Bello sicuramente, ma fortunatamente assolutamente semplice!



### Sommario

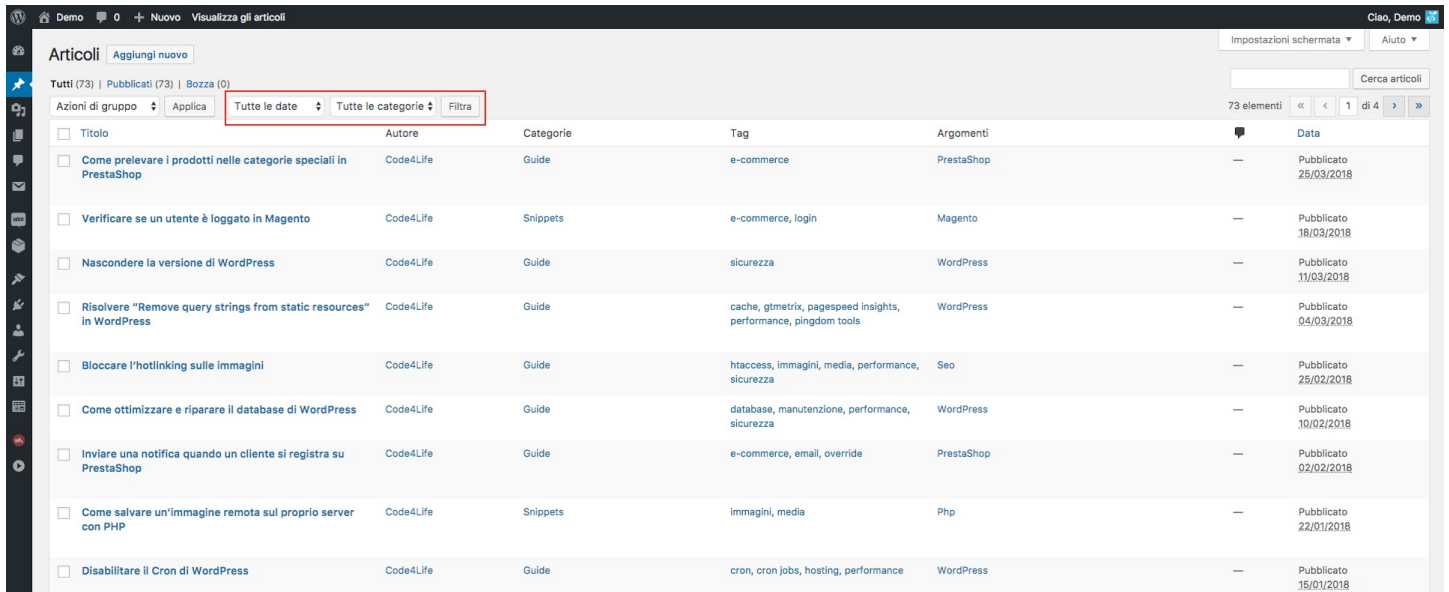
Il comportamento predefinito

Filtrare gli articoli tramite tassonomie personalizzate

Considerazioni finali

## Il comportamento predefinito

Come anticipato, WordPress dispone di default solamente del filtro per categoria e per data nella pagina di elenco articoli:



The screenshot shows the WordPress admin interface for the 'Articoli' (Articles) section. The top navigation bar includes 'Demo', '0', '+ Nuovo', and 'Visualizza gli articoli'. The main content area shows a list of articles with columns for 'Titolo', 'Autore', 'Categorie', 'Tag', 'Argomenti', and 'Data'. The 'Tutte le date' and 'Tutte le categorie' filters are highlighted with a red box. The 'Data' column shows the publication date for each article.

Titolo	Autore	Categorie	Tag	Argomenti	Data
<input type="checkbox"/> Come prelevare i prodotti nelle categorie speciali in PrestaShop	Code4Life	Guide	e-commerce	PrestaShop	— Pubblicato 25/03/2018
<input type="checkbox"/> Verificare se un utente è loggato in Magento	Code4Life	Snippets	e-commerce, login	Magento	— Pubblicato 18/03/2018
<input type="checkbox"/> Nascondere la versione di WordPress	Code4Life	Guide	sicurezza	WordPress	— Pubblicato 11/03/2018
<input type="checkbox"/> Risolvere "Remove query strings from static resources" in WordPress	Code4Life	Guide	cache, gtmatrix, pagespeed insights, performance, pingdom tools	WordPress	— Pubblicato 04/03/2018
<input type="checkbox"/> Bloccare l'hotlinking sulle immagini	Code4Life	Guide	htaccess, immagini, media, performance, sicurezza	Seo	— Pubblicato 25/02/2018
<input type="checkbox"/> Come ottimizzare e riparare il database di WordPress	Code4Life	Guide	database, manutenzione, performance, sicurezza	WordPress	— Pubblicato 10/02/2018
<input type="checkbox"/> Inviare una notifica quando un cliente si registra su PrestaShop	Code4Life	Guide	e-commerce, email, override	PrestaShop	— Pubblicato 02/02/2018
<input type="checkbox"/> Come salvare un'immagine remota sul proprio server con PHP	Code4Life	Snippets	immagini, media	Php	— Pubblicato 22/01/2018
<input type="checkbox"/> Disabilitare il Cron di WordPress	Code4Life	Guide	cron, cron jobs, hosting, performance	WordPress	— Pubblicato 15/01/2018

Naturalmente, combinando questi due filtri possiamo visualizzare solamente gli articoli che soddisfano i criteri di scelta.

## Filtrare gli articoli tramite tassonomie personalizzate

Ma ipotizza di aver creato una **Custom taxonomy** per gli articoli del tuo sito WordPress e di voler creare un ulteriore filtro che faccia riferimento alla nuova tassonomia, come fare?

È presto detto! A scopo esemplificativo, ipotizza di avere creato una nuova tassonomia "topic" e per crearne il filtro ti basta inserire nel file:

*wp-content/themes/{nome\_del\_tema}/functions.php*

che trovi nello spazio FTP del tuo sito ed inserire il codice:

```
// Aggiungo il filtro per argomenti agli articoli
add_action( 'restrict_manage_posts', function( $post_type, $which ) {
    // Applico solamente agli articoli
    if ( 'post' !== $post_type ) { return; }

    // Recupero i termini della tassonomia
    $terms = get_terms( 'topic' );

    // Visualizzo il filtro
    echo '<select name="topic" id="topic" class="postform">';
    echo '<option value=""> . esc_html__( 'All topics', 'code4life' ) . '</option>';

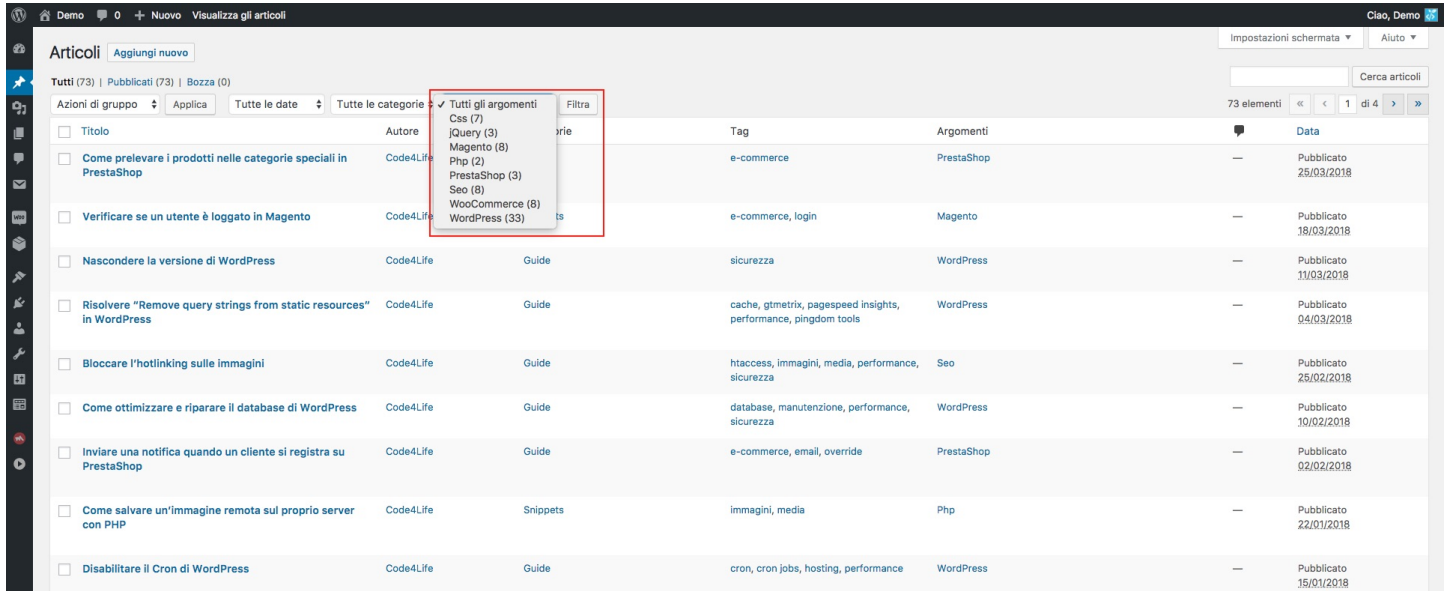
    // Visualizzo le opzioni
    foreach ( $terms as $term ) {
        printf(
            '<option value="%1$s" %2$s>%3$s (%4$s)</option>',
            $term->slug,
            ( ( isset( $_GET['topic'] ) && ( $_GET['topic'] == $term->slug ) ) ? '
selected="selected"' : '' ),
            $term->name,
            $term->count
        );
    }
    echo '</select>';
}, 10, 2 );
```

La funzione appena inserita è molto semplice, non fa altro che:

- **Riga 4:** controllare se il tipo di post su cui la si sta cercando di applicare è effettivamente un articolo (post)
- **Riga 7:** recuperare tutti i termini della tassonomia, da inserire come opzioni nel menu a tendina del filtro

- **Righe 10-23:** costruire il markup per la visualizzazione del menu a tendina che costituisce il filtro

Ed è tutto fatto! Ecco il risultato finale:



## Considerazioni finali

Naturalmente, quello preso in esame nell'articolo è solamente una casistica particolare, utilizzata a scopo esemplificativo. Partendo dall'esempio, è possibile aggiungere più di un filtro personalizzato e modificando il controllo sul **post type** (riga 4) è ovviamente possibile creare filtri personalizzati per qualsiasi **post type**, anche di tipo **custom**.

Non ti resta che dare libero sfogo alla tua fantasia!