

Custom Post Type: come strutturare al meglio i nostri contenuti?

Grazie all'introduzione dei *Custom Post Type* in Wordpress 2.9, il nostro motore di blogging preferito ha fatto l'ultimo passo che lo separava dal diventare un vero e proprio CMS. In questa serie di articoli vedremo come sfruttare le potenzialità di questa funzionalità per strutturare al meglio i nostri contenuti.

Cosa sono i *Custom Post Type*?

Un *Post Type* è una tipologia di contenuto gestibile da Wordpress (infatti a mio avviso il nome *Post Type* è fuorviante, sarebbe più corretto parlare di *Content Type*). Un'installazione standard di Wordpress contiene 5 *Post Type* predefiniti:

- Post
- Page
- Attachment
- Revisions
- Nav Menu

Per avere una visione più chiara, prendiamo come esempio i post e le pagine, che sono le tipologie con cui abbiamo più familiarità.

Un post è un tipo di contenuto che ha un titolo, un testo, un autore. Ad un post possono essere associate delle categorie, oppure dei tag. Un post, inoltre, può avere un riassunto.

Una pagina, allo stesso modo, ha un titolo, un testo ed un autore. Non puoi però associare (di default) delle categorie o dei tag ad una pagina, mentre puoi definire una gerarchia di pagine, cosa che non è possibile con i post.

Quindi post e pagine rappresentano due tipi di contenuto diversi, con caratteristiche diverse. **Un *Custom Post Type* (d'ora in poi *CPT*) non è altro che un tipo di contenuto personalizzato**, che puoi aggiungere a quelli predefiniti da Wordpress.

Fino alla versione attuale di Wordpress (3.4.2) purtroppo non esiste un'interfaccia grafica nel pannello di amministrazione che permetta di creare un nuovo custom post type, ma vedrai che basteranno poche righe di codice per migliorare la struttura del contenuto del tuo sito.

Organizziamo i nostri progetti

In molti siti web personali si sente l'esigenza di creare una sezione per mostrare al mondo i propri progetti personali. Alcune soluzioni prevedono di inserire i progetti come post, che poi verranno

rimossi manualmente dal [loop di Wordpress](#) (nella pagina blog, ad esempio) in modo da mostrare solo gli articoli. Invece noi creeremo un nuovo *CPT* "Progetto" in modo da tenere separati i nostri lavori dagli articoli standard. Ad ogni progetto saranno poi associate le tecnologie utilizzate per realizzarlo.

Creiamo il nostro primo *Custom Post Type*

Per creare un nuovo *CPT* possiamo utilizzare due approcci:

1. Inserire il codice nel file `functions.php` del tuo tema
2. Creare un plugin apposito.

Non c'è una regola precisa per scegliere un approccio piuttosto che l'altro. Se stai sviluppando un tema per un cliente e la categoria è strettamente correlata al tema, magari può essere preferibile inserire il *CPT* in `functions.php`. D'altro canto, se la tipologia di contenuto che stai creando è generica, un plugin ti garantisce una soluzione flessibile (e puoi anche dividerlo più facilmente con la comunità!).

Noi definiremo il nostro *CPT Progetto* attraverso un plugin: crea quindi una cartella chiamata `yiw` in `wp-content/plugins`, e al suo interno crea un file chiamato `yiw_custom_types.php`. Come avviene per la [creazione di ogni plugin](#), inserisci l'intestazione che Wordpress utilizzerà per mostrare le informazioni all'interno del pannello di amministrazione:

Fatto questo, vai nella sezione **Plugin** della tua installazione di Wordpress: nella lista dei plugin installati dovrebbe essere presente il nostro plugin: non resta che attivarlo. Al momento, però, il nostro plugin non effettua alcuna azione: dobbiamo quindi aggiungere il codice necessario per la creazione del nostro *CPT*.

La funzione che permette di creare un nuovo *CPT* in Wordpress è `register_post_type()`. Questa funzione accetta due parametri:

- una stringa che rappresenta il nome del nuovo Post Type: questo sarà l'identificatore del Post Type,
- un array di [argomenti](#)

Per creare quindi la nuova tipologia di contenuto aggiungi il seguente codice al nostro plugin:

Questo è tutto il codice necessario per creare un nuovo tipo di contenuto in Wordpress. Andando nel pannello di amministrazione di Wordpress vedrai che nella colonna a sinistra, sotto *Commenti* ci sarà una nuova voce: **Articoli**. Wordpress ha già realizzato tutto il necessario per consentirti di

creare, aggiornare, rimuovere e leggere il nuovo *CPT*.

Per testare che gli articoli scritti in questa tipologia sono completamente diversi dai normali post, prova a crearne uno: vedrai che non verrà mostrato nell'elenco dei post del tuo sito.

In realtà, ti accorgerai che non c'è alcuna sezione in cui è possibile visualizzare questi contenuti. Questo perché il tuo tema Wordpress non è predisposto per la visualizzazione di "progetti", ma porremo presto rimedio.

Inoltre avrai notato che i nostri "progetti" vengono riferiti come generici "Articoli", e che nella sezione editor vengono visualizzati solo i campi titolo e testo. Per personalizzare al meglio il nostro *CPT* dobbiamo aggiungere altri (dei tanti) parametri a `register_post_type()`. Per far questo, sostituisci il codice scritto precedentemente con il seguente:

Anche se sono molti parametri, la loro funzione è molto semplice. Analizziamo il codice riga per riga:

- [1] Aggiungiamo la funzione `crea_progetti` all'hook `init` di Wordpress: questo garantisce che all'inizializzazione Wordpress eseguirà la funzione passata come secondo parametro.
- [2] L'indice `labels` dell'array degli argomenti accetta un array contenente tutte le etichette che verranno mostrate nei vari link del pannello di amministrazione.
- [3] L'indice `public` definisce se deve essere aggiunta l'interfaccia grafica per la gestione del *CPT*. Può essere necessario disabilitarlo, ad esempio, nella creazione automatica di contenuti dove non è necessario l'intervento degli utenti.
- [4] `rewrite` definisce alcune regole per la creazione dei permalink del *CPT*. In questo caso vogliamo che il nome del progetto nell'URL sia preceduto dalla stringa 'progetti' e che la pagina archivi relativa a questo *Post Type* sia presente all'URL `http://tua_installazione_wordpress/progetti`.
- [5] `has_archive` specifica se deve essere generata una pagina contenente un elenco dei nuovi contenuti. Nel nostro caso è necessaria, quindi lo impostiamo a `true`.
- [6] L'indice `hierarchical` definisce se è possibile creare una gerarchia tra gli elementi del nuovo content type (una relazione tipo quella tra pagine e sottopagine). Anche se è

ammissibile l'esistenza di sottoprogetti, per semplicità manteniamo le cose lineari.

- [7] `menu_position` stabilisce la posizione che il nostro *Post Type* occuperà nella barra laterale del pannello di amministrazione. In questo caso è impostato a 5, che lo posiziona appena sotto i post (per altri valori consultare [il Codex](#)).
- [8] `supports` definisce quali tra i campi predefiniti da Wordpress devono essere associati al nuovo *Post Type*. I valori disponibili sono:
 - `title`: mostra il campo per l'inserimento del titolo
 - `editor` (contenuto): mostra la textarea principale, per l'inserimento del testo
 - `author`: mostra un menu a tendina per la scelta dell'autore
 - `thumbnail`: definisce se è possibile associare un'immagine tramite la funzionalità *Featured image* di Wordpress
 - `excerpt`: mostra il campo di testo per l'inserimento di un riassunto
 - `trackbacks`: se il nuovo *Post Type* supporta i trackback
 - `custom-fields`: mostra l'area per l'inserimento di *custom fields*
 - `comments`: abilita i commenti su questo *Post Type*
 - `revisions`: abilita le revisioni
 - `page-attributes` (`menu_order`, `hierarchical` must be true to show Parent option): mostra la sezione delle proprietà delle pagine (posizione nel menu, gerarchia). L'attributo `hierarchical` deve essere posto a true
 - `post-formats`: abilita i [Post formats](#), una funzionalità introdotta in Wordpress 3.1

Per il nostro esempio utilizzeremo solo i campi `title`, `editor` e `thumbnail`.

- [9] Infine registriamo il nostro *CPT*, passando l'identificatore e l'array di parametri costruito precedentemente.

Una pagina per tutti i nostri progetti

Ora non ci resta che realizzare una pagina per elencare tutti i nostri progetti. In realtà, questa pagina è già stata creata da Wordpress per noi: infatti l'archivio serve proprio a questo scopo. Nonostante questo, non è ancora possibile inserire questa pagina nei menu di Wordpress tramite il pannello di amministrazione. Solo a scopo di test, scriveremo a mano l'URL dell'archivio per accertarci del corretto funzionamento e, nel prossimo articolo, realizzeremo una pagina ad hoc per elencare i nostri progetti.

Aggiungiamo quindi la pagina al menu principale di Wordpress: questo passo varierà a seconda del tema che hai installato, ma la procedura dovrebbe essere simile. Io mi rifarò al tema predefinito di Wordpress, *Twenty Eleven*.

Vai nella sezione *Aspetto=>Menu* del pannello di amministrazione di Wordpress e aggiungi un nuovo menu (se non ne hai uno esistente). Nel riquadro *Link personalizzati* aggiungi i seguenti valori:

- nel campo URL: http://tua_installazione_wordpress/progetti
- nel campo Etichetta: "Progetti"

e clicca su *Aggiungi al menu* e poi *Salva menu*. Ricaricando la pagina principale dovresti vedere il nuovo link, che porta all'elenco dei nostri progetti.

Your Inspiration Web

Web Design Community, ispirazione, tutorial, guide e risorse gratuite.

Come accennato in precedenza, però, questa non è una procedura ottimale: infatti non è mai una buona idea inserire un valore manualmente quando può essere ricavato automaticamente. Nel prossimo articolo vedremo come ovviare a questo problema con un template di pagina. Inoltre aggiungeremo una tassonomia specializzata al nostro *CPT*.

Conclusioni e considerazioni

La strutturazione e classificazione dei contenuti è il punto di forza di ogni CMS. Se nel passato il fulcro centrale di Wordpress era il *blogging*, grazie ai *Custom Post Type* le possibilità d'uso si moltiplicano, consentendo di realizzare facilmente anche applicazioni complesse.

Molti sviluppatori esperti potrebbero storcere il naso di fronte a questo utilizzo di un semplice motore di blogging e consigliare uno dei più blasonati Drupal o Joomla. Però nell'adozione di un prodotto bisogna tener conto anche dell'esperienza: iniziare un lavoro con Drupal senza averlo mai utilizzato è ben diverso dall'imparare come sfruttare i *CPT* in Wordpress se quest'ultimo è la piattaforma che usiamo quotidianamente.

Insomma, come in ogni scelta, non esiste il bianco e il nero: bisogna essere consapevoli delle capacità di ogni strumento e saper discernere quando utilizzarne uno piuttosto che l'altro. E mai smettere di imparare.