What is a taxonomy?

Taxonomy is one of those words that most people never hear or use. Basically, a taxonomy is a way to group things together.

For example, I might have a bunch of different types of animals. I can group them together according to various characteristics and then assign those groups names. This is something most people encounter in biology classes, and it is known as the Linnaean Taxonomy.

In WordPress, a "taxonomy" is a grouping mechanism for some posts (or links or custom post types).

The names for the different groupings in a taxonomy are called *terms*. Using groupings of animals as an example, we might call one group "birds", and another group "fish". "Fish" and "birds" are terms in our taxonomy. As an example from WordPress, a category or tag (see next section) is a term.

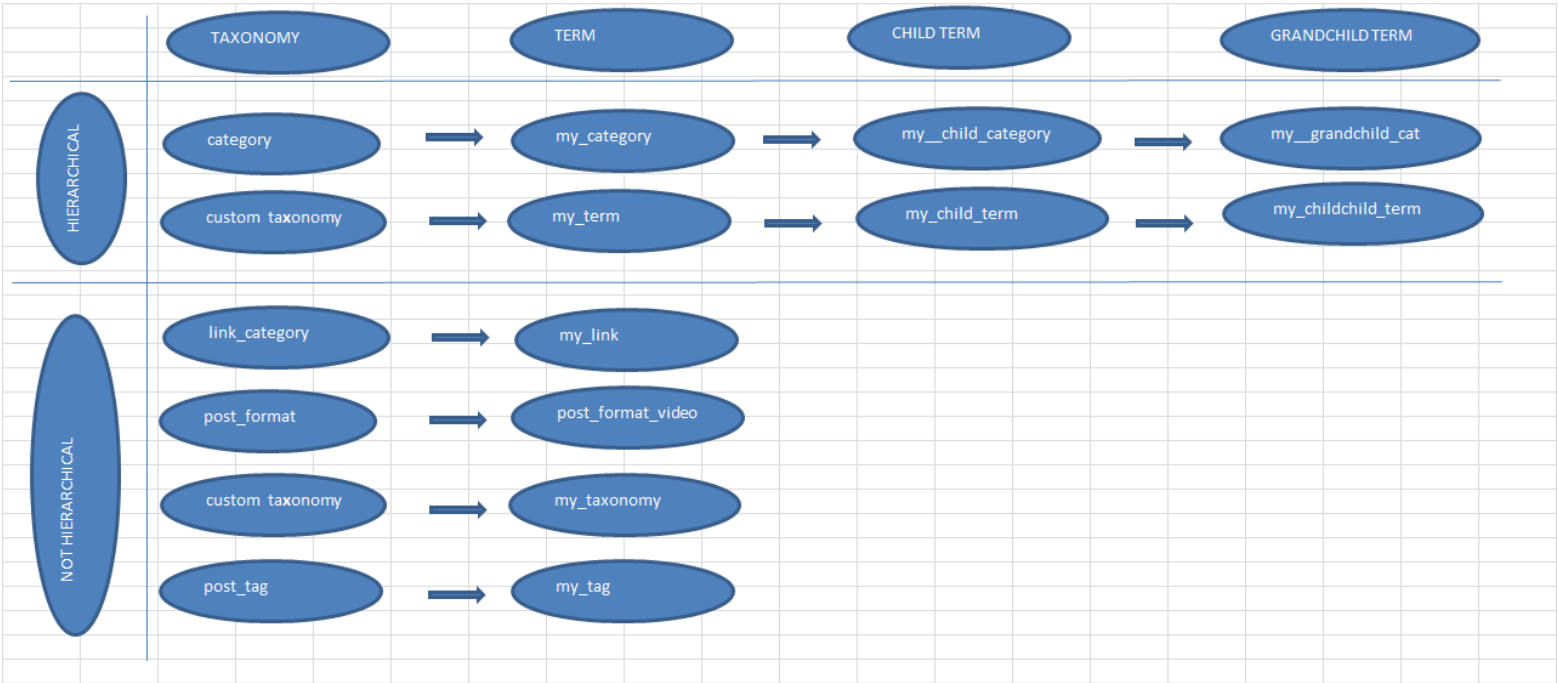# Basic diagram on taxonomies and their relationships in WordPress

Diagram added with no restrictions with compliments by Pieter Goosen

Default Taxonomies

WordPress has four built in taxonomies that you've probably used already.

Category

The 'category' taxonomy lets you group posts together by sorting them into various categories. These categories can then be seen on the site by using '/category/name' types of URLs. Categories tend to be pre-defined and broad ranging.

Tag

The 'post_tag' taxonomy is similar to categories, but more free form. Tags can be made up on the fly, by simply typing them in. They can be seen on the site in the '/tag/name' types of URLs. Posts tend to have numerous tags, and they are generally displayed near posts or in the form of tag clouds.

Link Category

The 'link_category' taxonomy lets you categorize your links. These tend to be used only internally, for organizational reasons, and are not usually exposed on the site itself. They are handy for defining groups of links to be displayed in sidebars and the like.

Post Formats

The 'post_format' taxonomy was introduced in WordPress 3.1 and it is a piece of meta information that can be used by a theme to customize its presentation of a post. New post formats can't be created or added to the default existing ones.

Custom Taxonomies

Since WordPress 2.3, you've been able to create your own custom taxonomies, but these have been a rarely used feature of WordPress until Version 2.9. In truth, they are an extremely powerful way to group various items in all sorts of ways.

Example

The plugin Matt's Community Tags uses taxonomies to define "people" as a taxonomy for attachments. He uses it to allow people to mark the names of people in pictures, and using that his site can display pictures of people under the '/person/name' URL.

Registering a taxonomy

To register a taxonomy, you use the register_taxonomy() function.

Here's an example of registering a "people" taxonomy:

```
function people_init() {
        // create a new taxonomy
        register_taxonomy(
                'people',
                'post',
                array(
                        'label' => __( 'People' ),
                        'rewrite' => array( 'slug' => 'person' ),
                        'capabilities' => array(
                                'assign_terms' => 'edit_guides',
                                'edit_terms' => 'publish_guides'
                        )
                )
        );
}
add_action( 'init', 'people_init' );
```

Here, the "people" taxonomy is defined. It's defined to work for posts, and a rewrite slug is defined to make the url into '/person/' instead of '/people/'. The capabilities line is optional. Without it, WordPress will default capabilities to the same users as posts. As shown above, this will allow any user with the custom "edit_guides" capability to assign the taxonomy to a post and any user with the custom "publish_guides" capability to create new taxonomy items.

Taxonomy capabilities include assign_terms, edit_terms, manage_terms (displays the taxonomy in the admin navigation) and delete_terms.

Using that taxonomy

Once you've added a taxonomy, you'll find that WordPress creates a new meta box on posts for you. This new meta box looks almost exactly like the Tags box and will let you add tags to those posts.

If you're not attaching your taxonomy to posts, then you may not get the interface created for you. Taxonomies are generic, after all, you could create one for any sort of object. To add terms to an object using your taxonomy, you'll need to use the wp_set_object_terms() function. Here's an example of adding the term "Bob" to post ID number 123 in the "person" taxonomy:

```
wp_set_object_terms( 123, 'Bob', 'person' );
```

As you can see, it's simple to do. The second parameter can also be an array of terms to add all at once, if you need to do so.

Clouds

The wp_tag_cloud() function can also accept a "taxonomy" parameter, if you want to display a cloud of terms for your custom taxonomy.

Listing the terms

If you want to have a custom list in your theme, then you can pass the taxonomy name into the the_terms() function in the Loop, like so:

```
the_terms( $post->ID, 'people', 'People: ', ', ', ' ' );
```

That displays the list of People attached to each post.

Querying by taxonomy

Creating a taxonomy generally automatically creates a special query variable using WP_Query class, which we can use to retrieve posts based on. For example, to pull a list of posts that have "Bob" as a "person" taxonomy in them, we will use:

```
$query = new WP_Query( array( 'person' => 'bob' ) );
```

or, for more complex argument:

```
$args = array(
        'tax_query' => array(
                array(
                        'taxonomy' => 'person',
                        'field' => 'slug',
                        'terms' => 'bob'
                )
        )
);
$query = new WP_Query( $args );
```

404 Error

If your site uses custom permalinks, you will need to flush your permalink structure after making changes to your taxonomies, or else you may see a "Page Not Found" error. Your permalink structure is automatically flushed when you visit Settings > Permalinks in your WordPress dashboard.

Related

- Article: **Categories**
- Function: the_category()
- Function: the_category_rss()
- Function: single_cat_title()
- Function: category_description()
- Function: wp_dropdown_categories()
- Function: wp_list_categories()
- Function: get_the_category()
- Function: get_the_category_by_ID()
- Function: get_category_by_slug()
- Function: get_the_category_list()
- Function: get_category_parents()
- Function: get_category_link()
- Function: is_category()
- Function: in_category()