

WordPress Bits

Hacking WordPress. Keeping the bits together.

- [Home](#)
- [About](#)
- [Contact me](#)
- [FAQ](#)

-

• Categories

- [WordPress](#) (30)
 - [Database](#) (4)
 - [Documentation](#) (6)
 - [Editor](#) (1)
 - [Filters](#) (1)
 - [Login](#) (1)
 - [Menu](#) (3)
 - [Options](#) (6)
 - [Pages](#) (1)
 - [Plugins](#) (9)
 - [Themes](#) (6)
 - [Widgets](#) (6)
- [WordPress Tips](#) (9)

• Archives

- [December 2007](#)
- [October 2007](#)
- [September 2007](#)
- [August 2007](#)
- [June 2007](#)

• WP Designer Tutorials

- [How to Use Wordpress as a Membership Directory](#)
- [How to Create An Archives Page](#)
- [Do-It-Yourself WordPress Hover Menu](#)
- [How to Put Your Best Content Forward](#)
- [Part 3 - How to Code a WordPress Theme](#)
- [How to Slice a WordPress Theme](#)
- [How to Design a WordPress Theme](#)
- [How to manipulate category templates](#)
- [Style featured posts with WP-Sticky plugin](#)
- [How to place a login form in the sidebar](#)

• Meta

- [Log in](#)
- [Entries RSS](#)
- [Comments RSS](#)
- [WordPress.org](#)

• Subscribe
















- [Entries \(RSS\)](#)
- [Comments \(RSS\)](#)

• Top Posts

- [A look inside the WordPress database](#)
- [Making WordPress themes II : The Loop](#)
- [Having fun with WordPress login. Box or not.](#)

- [Making WordPress themes I : static basics](#)
- [Making WordPress themes III : template hierarchy](#)

• Recent Comments

-  [Webmasterdays BackOf...](#) on [Making WordPress themes III : ...](#)
-  Hadi on [Having fun with WordPress logi...](#)
-  [Talker](#) on [A look inside the WordPress...](#)
-  [WordPress source cod...](#) on [Navigating WordPress source...](#)
-  [atlantageek](#) on [A look inside the WordPress...](#)
-  [Wordpress Coding - W...](#) on [Widgetized web site. Beyond...](#)
-  [Leonid Mamchenkov](#) on [Tip #9 : Use trackback au...](#)
-  [Leonid Mamchenkov](#) on [Tip #9 : Use trackback au...](#)
-  [Michel](#) on [Tip #9 : Use trackback au...](#)
-  [Leonid Mamchenkov](#) on [Tip #9 : Use trackback au...](#)
-  [Michel](#) on [Tip #9 : Use trackback au...](#)
-  [Credit Card Finder](#) on [Tip #9 : Use trackback au...](#)
-  chrisber on [A look inside the WordPress...](#)
-  [Leonid Mamchenkov](#) on [A look inside the WordPress...](#)
-  chrisber on [A look inside the WordPress...](#)

• Coding links

- [DZone Snippets](#)
- [MySQL Manual](#)
- [PHP Manual](#)
- [W3 Schools](#)

• WordPress Links

- [WordPress Codex](#)
- [WordPress Hooks](#)
- [WordPress MU](#)
- [WordPress Source](#)
- [WordPress Widgets Blog](#)
- [WordPress.com](#)
- [WordPress.org](#)
- [WordPress Plugin Directory](#)

« [Quick access to WordPress options](#)
[Automating WordPress customizations - the plugin way](#) »

A look inside the WordPress database

Posted by [Leonid Mamchenkov](#) on August 8, 2007

Update: This article is also available in [Chinese](#) (thanks to [Denis](#)).

WordPress, as most other modern content management systems, is a very database-centric application. It keeps all information in the database - blog settings, posts, comments, links, users, etc. Therefore it is important to understand how the database is organized, what types of data is stored there, and how different things are linked to each other.

But before we dive into it, here are three points that I want to stress out:

1. WordPress API provides a number of ways to manipulate data without directly accessing the database. So, technically, you don't have to know how the database is organized. However, if you do, you'll probably find solutions to your problems faster, and they'll be easier to implement.
2. WordPress database scheme is small and simple. There is nothing to be scared of. It's only 10 tables, with just a few fields each. Overall, the scheme makes so much sense, that you will probably keep most of it in mind after a second look at it, even though you don't have to.
3. WordPress database scheme is [well documented](#). The only thing with that Codex page is that it explains tables in alphabetic order, which makes sense when you are looking up something. But for the first few times, it's better to have a different order, which is what this post is trying to achieve.

OK, now let's see what is there to see.

For the rest of the article, you'll need either another browser tab with previously mentioned [database description](#) at Codex, or a database tool, such as PHPMyAdmin or MySQL command line. To find a list of database tables in your WordPress installation run *"SHOW TABLES"* query. To get a specification of any table use *EXPLAIN* command, like so: *"EXPLAIN wp_posts"*.

wp_posts

WordPress is mostly used as a ~~blogging engine~~ content management system (CMS, for short). So, the "content" table, I think, is of primary importance. Pages, posts, and attachments are all content. And WordPress keeps them all in one table - *wp_posts*.

Actually, attachments are kept as they are - files on disk, but WordPress uses *wp_posts* table to keep attachment meta information, such as ID of user who uploaded it, ID of post to which the attachment is linked, the date it was uploaded, etc.

Pages, posts, and attachments can be very similar and they can be very different. For now, we want to look at things which are common between them. All three have a unique ID (usually generated by WordPress or MySQL automatically), an author (person who wrote the post or uploaded the file), posting date, title, and description (even if empty, there is still a way to add it and use it later). All three have a URL. So, for each post, page, and attachment, WordPress keeps a record in *wp_posts* table.

Now, there are a few ways that posts, pages, and attachments differ. To differentiate between them, WordPress uses a column *post_type* in *wp_posts* table. For posts, the value of *post_type* is 'post'. For pages - 'page'. For attachments - 'attachment'.

Then, there are a few columns which are used only when post is of some specific type. For example, column *post_mime_type* is used to keep attachments file types, such as 'image/jpeg' and 'application/pdf' (read about [MIME](#) at Wikipedia). Column *menu_order* is used to specify the order of pages. Column *post_parent* is used to organize pages with sub-pages, as well as to link attachments to posts.

To get a better picture of how and when each field is used, take a look at real records for each post type:

- *SELECT * FROM wp_posts WHERE post_type = 'post' LIMIT 1*
- *SELECT * FROM wp_posts WHERE post_type = 'page' LIMIT 1*
- *SELECT * FROM wp_posts WHERE post_type = 'attachment' LIMIT 1*

If you are using MySQL command line tool, then you will appreciate a "\G" at the end of the query ("SELECT * FROM wp_posts WHERE post_type='post' LIMIT 1 \G"). It will give you the results in a slightly more readable format.

Post status is also kept in this table. In most cases you'll see 'publish' for published posts, 'draft' for drafts, and 'inherit' for attachments. But there are a few others that you can learn from the table specification.

Here are a couple of ideas for you on what you could do with just *wp_posts* table:

- Generate some statistics - how many posts are there in total, how many posts of each type, how many posts each author published, what is the average number of posts published per day over a period of time, etc.
- Generate a list of all attachments
- Generate an image gallery for each post (hint: *"WHERE post_type = 'attachment' AND (post_mime_type = 'image/jpeg' OR post_mime_type = 'image/gif' OR post_mime_type = 'image/png')"*).

wp_postmeta

There are probably a million of things that you'd want to keep about some of your posts - the mood you were when wrote it, the song you were listening at the moment, the geographical location you were at, some list of related links, post-specific information for search engines (SEO), and so on, and so forth. All these things go into *wp_postmeta* table.

The structure of this table is very simple and flexible. It has only four columns: *meta_id*, *post_id*, *meta_key*, and *meta_value*. *meta_id* is automatically generated and I can't think of a scenario where you would want to change that - there are many other fun things to do. *post_id* is the ID of the post (as in record from the *wp_posts* table) to which current meta information is linked. *meta_key* is the description of the information that you want to attach to the post (such as 'mood' or 'song'). *meta_value* is the actual content of the meta information (such as 'Grumpy!' or 'The WordPress Song.mp3').

WordPress uses this table to store additional information about attachments, which does not fit into *wp_posts* table (file system path to the attachment,

thumbnail information, if any, etc). Also, this table is used to store custom field names and values - those that you see in the box under your post editor. And, of course, there are tonnes of plugins out there which utilize this table for their own needs. So, if you decide to use this table in your plugin, make sure that you come up with some unique names for your meta keys. Otherwise, your plugin might interfere with some other plugin and you'll get some really weird and totally unexpected results.

wp_users

WordPress uses this to store registered users. User ID, login, encrypted password, full name, registration date, and things like that are kept in *wp_users*. Think of it as essential information about the users.

wp_usermeta

The *wp_usermeta* table for users is exactly what *wp_postmeta* table is for posts. You can keep all sorts of things for each registered user here - personalized blog settings (WordPress already does so for visual editor preference, for example), birthdays, contact information, etc.

wp_comments

wp_comments

table keeps all comments for your site. These include all approved comments, comments awaiting moderation, SPAM comments, and even trackbacks and pingbacks which are sent to you from other web sites. You can probably guess most of the columns in this table without me telling you. But just in case, here are the ones you probably care about:

- `comment_ID` - unique ID of each comment, automatically generated by MySQL
- `comment_post_ID` - ID of the post for which the comment was made
- `comment_author` - name of the comment author
- `comment_author_email` - email of the comment author
- `comment_author_url` - URL of the comment author
- `comment_author_IP` - the IP that comment author came from
- `comment_date` - the data and time the comment was made
- `comment_content` - the actual comment
- `comment_approved` - whether the comment was moderated, and if yes, then how
- `comment_agent` - browser signature of the commenter (MSIE, Firefox, Safari, etc)
- `comment_type` - if it is a regular comment, or trackback, or something else
- `user_id` - if comment user is a registered user, you'll have his ID in here

There are a couple of other fields in there, but we'll leave them for now, as we did with other tables.

wp_options

As we saw in update to "[Quick access to WordPress options](#)" post, *wp_options* is the table that holds all the global options for your WordPress installation. The core of this table is very similar to the *wp_postmeta* and *wp_usermeta* tables. And it is understandable - *wp_options* stores information about your WordPress in much the same way, as *wp_postmeta* stores information about posts and *wp_usermeta* stores information about users. However, you will probably notice a few extra columns.

blog_id column is there, but it is not of much use on a standalone WordPress installations. It is used only by [WordPress MU](#). Regular WordPress and WordPress MU share most of the source code and database structure. So, WordPress MU uses *blog_id* column in *wp_options* table to differentiate between options for different blogs under one roof.

autoload

column controls if the option is always loaded from the database and cached for use by WordPress and plugins or if it is read only when asked for. Most options that I've seen, are autoloaded.

wp_categories

If you have any categories in your WordPress installation, *wp_categories* is the table that keeps those records. Category names and descriptions are stored here, as well the ID of each category's parent.

To work faster, WordPress often keeps aggregated values in the database, instead of recalculating them each time. For example, frequently requested counts of posts and links in each category are simply stored in the *wp_categories* table (WordPress uses the same set of categories for both links and posts). Every time you add post to a category, the post counter (column *category_counter*) increases. Every time you remove post from the category, the counter decreased. Same goes for links (column *link_count*). That's why you see those extra columns in the table.

wp_post2cat

Linking posts to categories is done via *wp_post2cat* table. This is a standard approach for many-to-many relationships in relational databases. The *wp_post2cat* table has only three fields: the unique record ID (automatically generated), the ID of the post, and the ID of the category to add the post to.

wp_links

Usually *wp_links* table is used to store the blogroll - web site's list of links. Check the blogroll administration interface carefully, and you'll see lots of options, which aren't used by most WordPress users (no statistical proof here though). All those links options can be stored in *wp_links* table. In fact, you won't need to write much code to turn WordPress into a link directory, like Yahoo! once was. It'll be only a matter of an appropriate theme.

One important bit to notice here, is that `link_owner` is kept for each link. So, you know who created the link.

wp_link2cat

`wp_link2cat` table is the exact copy of `wp_post2cat`, except it keeps the ID of the link, instead of the ID of the post.

That's all there is to WordPress database structure. If you see any other tables in your installation, then they were probably added by one of the plugins that you've installed.

Update: Those of you who prefer a bird's eye view, will appreciate [WordPress Database Structure](#) (PNG file, 167 KBytes, [from here](#)).


Update: [Ryan Boren](#) has posted an explanation of [database changes in WordPress 2.3 related to taxonomy](#) - a new way of tagging and categorizing posts and links.


Possibly related posts: (automatically generated)

- [The Top Ten WordPress Plugins](#)
- [Computer Self-Efficacy Data](#)
- [Adding attachments to Wordpress Blog Posts](#)


This entry was posted on August 8, 2007 at 5:33 pm and is filed under [Database](#), [WordPress](#). . You can follow any responses to this entry through the [RSS 2.0](#) feed. You can [leave a response](#), or [trackback](#) from your own site.

47 Responses to “A look inside the WordPress database”

-  [ia](#) Says:
[August 8, 2007 at 10:02 pm](#)

This is great. [This page](#) has some examples but is not up-to-date. It's worth noting how WP differentiates pages from posts in the database. I needed that info when I was updating to the latest version just now—all the pages became posts! :)
-  [sheida](#) Says:
[August 8, 2007 at 10:13 pm](#)


good job

it is really needed to know exactly what is going on
I figure out while playing with database, there were no tutorial for that
-  [Leonid Mamchenkov](#) Says:
[August 8, 2007 at 10:16 pm](#)

ia,

Thanks for the link. It has some good descriptions of what's going on, but it is, if I understand correctly, very outdated. It covers WordPress 2.0, while the most recent version of WordPress is 2.2 (OK, 2.2.2). And 2.3 is coming out shortly, from what I here.


It seems that before, WordPress used `post_status` to differentiate between posts and pages. In the example on that page, it mentions `post_status = 'static'`, and `post_type` is not set in all example records. This is not so in WordPress 2.2.2.

Anyway, it's good to have a few documents around, when trying to figure this stuff out. So thanks for the link once again.
-  [Leonid Mamchenkov](#) Says:
[August 8, 2007 at 10:28 pm](#)

Sheida,

thanks.

A lot of interesting stuff can be done with direct database access and with WordPress Database class ([wpdb](#)), so we'll sure be coming back to this subject more in the future.
- [Using Wordpress to Power Your Website - Adirondack Base Camp - Travel, Tourism, Outdoors, News and Information for the Adirondack Region of New York](#) Says:
[August 8, 2007 at 10:37 pm](#)

[...] A look inside the WordPress database - A great explanation - makes that codex article come alive. [...]
-  [munggur](#) Says:

[August 8, 2007 at 11:14 pm](#)

Agree with you. It's all about database. One interesting thing is that using the data, WP can make big 'collective knowledge', show stats and relate each post (with comments) to other post.

After all, each of user can even download and upload their data (posts and comments and etc.) in .xml format. Great job. WP!

7. [Top Posts « WordPress.com](#) Says:

[August 8, 2007 at 11:59 pm](#)

[...] A look inside the WordPress database WordPress, as most other modern content management systems, is a very database-centric application. It keeps all [...] [...]

8. [Ineation](#) Says:

[August 9, 2007 at 6:14 am](#)

Excellent Post (and blog).

Indeed I do really think that wp Db is very simple BUT very flexible.

One of the most exciting thing is in the post table. The fact that you can deal with other kind of objects (page, attachment, but you can easily imagine others) is very powerful if you want to customize and adapt the application for your need without adding a new table. The question is : should we integrate as much custom object in existing tables or would it be safer that plugins deals with their own tables ? (I guess the answer is "it depends !").

Bye alls.

9.  [Leonid Mamchenkov](#) Says:

[August 9, 2007 at 6:58 am](#)

Ineation,

indeed. The thing to remember here is that WordPress own things come with certain logic attached to them. Such as: pages can inherit each other, pages can be ordered, posts can be organized into categories, etc. If the things that you do from your plugin follow any of these logic, than you better use WP internal stuff. If not - than your own tables, and your own logic.

10. [Mikael](#) Says:

[August 9, 2007 at 10:35 am](#)

Great post and a great blog!

Found lots of interesting thoughts and facts here..

Keep up the good work Leonid!

11.  [Leonid Mamchenkov](#) Says:

[August 9, 2007 at 10:41 am](#)

Thanks Mikael. Will try to ;)

12. [DELETE YOURSELF! » Blog Archive » База данных WordPress изнутри](#) Says:

[August 9, 2007 at 4:37 pm](#)

[...] [Источник на английском] Очень вольный перевод - мой. [...]

13. [Nick Georgakis](#) Says:

[August 9, 2007 at 5:38 pm](#)

Thanks for this invaluable post, it complements/extends nicely the available literature on WordPress DB.

You have managed to create a very interesting and useful site for all "technically" inclined WordPress users who want to know "What is going under the hood" and to furtherly modify - enhance their WordPress experience.

Thanks for the great articles and wishes for the best.

As Michael said keep up the good work ;-)

WordPress Bits deservedly earned a place in my RSS feeds and took the liberty to add this post to StumbleUpon - Hope I didn't cause any problem with the additional traffic :-)

14. [www.ngtech.gr » Looking inside the WordPress DB](#) Says:

[August 9, 2007 at 5:44 pm](#)

[...] today I found A look inside the WordPress database ? at WordPress Bit through WP dashboard , an excellent complement to official WordPress DB Description in WordPress [...]

15.  [Leonid Mamchenkov](#) Says:

[August 9, 2007 at 5:55 pm](#)

Nick,

thanks for the kind words.

Feel free to promote the blog all you want - the traffic is not a problem, as the blog is hosted on WordPress.com . I bet it can easily handle Slashdot/Digg/Reddit and other effects of the modern web (hint, hint) ;)

16. [Weblog Ojm.be » Blog Archive » de Wordpress database](#) Says:
[August 9, 2007 at 10:09 pm](#)

[...] en totaal geen verstand hebt van databases (ook net als ik!!) dan is het misschien wel eens leuk om dit artikel van Wordpress Bits door te nemen. Simpele uitleg en hier en daar een goeie tip voor dare-devils onder [...]

17. [Famous last words of Marius » RSS feed update - 10 August 2007](#) Says:
[August 10, 2007 at 9:59 am](#)

[...] WordPress – A look inside the WordPress database [...]

18. [Murk](#) Says:
[August 10, 2007 at 11:26 am](#)

I'm really pleased to have found these pages at the beginning of the series on hacking WP. (I'm pretty computer literate, but from the days of pascal and Fortran - php and sql passed me by).

I'd love to be able to write a plugin which would be able to return (one at a time) the next and previous posts which share a category with the current post.

e.g.


and to do the same for dates (depending upon context). (Each post title would be visible)

I'm hoping to learn enough to do that... I'm not holding my breath though!

19. [Murk](#) Says:
[August 10, 2007 at 11:27 am](#)

The other plugin I want to do is 'remember my activated plugins'. So that as an upgrade takes place you can simply click 'remember' then 'disable'.


After the upgrade, one would click 'recall' and it'd activate the plugins one at a time, alerting the user to any problems. This seems tricky to me!

20.  [Leonid Mamchenkov](#) Says:
[August 10, 2007 at 1:27 pm](#)


Murk,

one of the best ways to learn is to look at other people's code and try to make your own. There are plenty of plugins available already, which do all sorts of things with pages, posts, and categories. Get your hand on some of the simple ones and work your way through. It isn't that difficult - the main issue is to get familiar with functions that WordPress has for manipulating this data.

Regarding the "remember plugins" plugin - I think this is or will shortly be done in WordPress itself. I remember I saw some code floating around.


21.  [Paveo](#) Says:
[August 10, 2007 at 6:56 pm](#)

Thanks, it helps me understand more.

22.  [Denis](#) Says:
[August 13, 2007 at 11:34 am](#)

Hi Leonid,

Can I translate your posts to Chinese?

23.  [Leonid Mamchenkov](#) Says:
[August 13, 2007 at 11:49 am](#)

Denis,

yes, of course, by all means, please do. :)

Send me the URL when you're done, so that I could link to the translation from here.

24. [links for 2007-08-09 en newdisco](#) Says:
August 16, 2007 at 2:15 am


[...] A look inside the WordPress database « WordPress Bits La base de datos de Wordpress mirada por dentro. (tags: wordpress database reverency) Archivado en: Links Diarios | Etiquetas: No Tags. [...]

25. [» Looking inside the WordPress DB Piperato: Just another WordPress weblog](#) Says:
August 24, 2007 at 11:31 am

[...] today I found A look inside the WordPress database at WordPress Bit through WP dashboard , an excellent complement to official WordPress DB Description in WordPress [...]

26. [Tip #5 : Taxonomy in WordPress 2.3 « WordPress Bits](#) Says:
August 27, 2007 at 10:33 am


[...] Recent Comments Famous last words of... on Tip #4 : WordPress cheat sheet...One Thousand and Twe... on More WordPress theming at WPDe...Bob on Tip #4 : WordPress cheat sheet...Vide08 on More WordPress theming at WPDe...Famous last words of... on Making WordPress themes III : ...Famous last words of... on Making WordPress themes II : T...Famous last words of... on Making WordPress themes II : T...Famous last words of... on Tip #3 : Under Construction...Leonid Mamchenkov on Making WordPress themes III : ...engtech on Making WordPress themes III : ...Cigar Inspector on Making WordPress themes III : ...Leonid Mamchenkov on Making WordPress themes III : ...Cigar Inspector on Making WordPress themes III : ...Leion on Tip #4 : WordPress cheat sheet... Looking inside the... on A look inside the WordPress da... [...]

27.  [Denis](#) Says:
September 5, 2007 at 5:05 pm

Hi Leonid,

Here is my Chinese Version of your post:

<http://fairfish.net/2007/08/11/wordpress-database-intro/>

28.  [Leonid Mamchenkov](#) Says:
September 5, 2007 at 6:24 pm

Denis,

Thanks a lot of translating it. I've updated the post and now there is a link to your translation at the top.

29. [WordPress 数据库探究 #1 — 介绍 | 我爱水煮鱼](#) Says:
September 13, 2007 at 12:25 pm


[...] 这个系列教程翻译自: A look inside the WordPress database。由于原文比较长, 我把它分成一系列教程, 今天第一篇是开篇介绍。 [...]

30. [links for 2007-09-18 | After The Dot Net](#) Says:
September 18, 2007 at 1:11 pm

[...] A look inside the WordPress database « WordPress Bits [...]


31. [出家如初, 成佛有余» WordPress Plugin 架构](#) Says:
October 2, 2007 at 10:57 pm

[...] <http://wpbits.wordpress.com/2007/08/08/a-look-inside-the-wordpress-database/> [...]

32.  [Pascal Zoghbi](#) Says:
October 20, 2007 at 7:13 pm

hello , i have a small problem. i was editing onw of my posts and i have deleted part of it and saved it by mistake, so now part of my post is missing. and i did not keep a copy on my computer. is there a way to find the old information of my post on the database of wp. and if yes, how?

thanks

33.  [Leonid Mamchenkov](#) Says:
October 20, 2007 at 8:31 pm

Pascal,

no. It's gone now.


34. [Tay - Super Blogging](#) Says:
[December 27, 2007 at 8:43 am](#)

I know this is a pretty old post, but I need some help. I was upgrading my blog to the latest Wordpress version, 2.3.1, and now I can't create or edit posts. I get this message:

```
WordPress database error: [Table 'superbt_wordpress.wp_post2cat' doesn't exist]
SELECT cat_ID AS ID, MAX(post_modified) AS last_mod FROM `wp_posts` p LEFT JOIN `wp_post2cat` pc ON p.ID = pc.post_id LEFT JOIN `wp_categories` c ON pc.category_id = c.cat_ID WHERE post_status = 'publish' GROUP BY cat_ID
```

Warning: Cannot modify header information - headers already sent by (output started at /home/superbt/public_html/wp-includes/wp-db.php:160) in /home/superbt/public_html/wp-includes/pluggable.php on line 391

My database doesn't have the post2cat or the categories tables anymore. I never messed around with my database, so I'm not sure how this happened. Is there any way I can recreate those? Please email me - thank you so much. :)

35.  [Leonid Mamchenkov](#) Says:
[December 27, 2007 at 11:52 am](#)

Tay,

it looks like either your upgrade didn't go through completely or some of your plugins are outdated. Try disabling all plugins and checking if you still get the error. If you still have it, something is wrong with your upgrade and you better redo it. Otherwise, enable plugins one by one to see which one causes the problem. Probably there is a newer version of it to which you can upgrade.


36. [Miyijura](#) Says:
[January 16, 2008 at 8:40 am](#)

Tay, thats because wordpress 2.3 change the categories, post2cat and link2cat for terms, term_taxonomy and term_relationships. Good luck fixing that!

37. [WTJ](#) Says:
[February 5, 2008 at 6:28 am](#)

i'm using wp 2.3 now. I upgraded from wp 2.2 to 2.3, and I encounter a big problem.

When I preset a post to be posted on 10th february, then i reedit the timestamp to post on 18th feb, however the post won't post on 18th feb, but still published on 10th feb (but in the public post, the date is written as 18th feb)


38.  [Leonid Mamchenkov](#) Says:
[February 5, 2008 at 11:24 am](#)

WTJ,

make sure that "Modify Timestamp" checkbox is checked when changing the date. If you still have the problem, you should probably ask in the Support forums or report a bug to WordPress developers.

39. [A look inside the WordPress database](#) Says:
[February 19, 2008 at 4:03 am](#)

[...] January 29th, 2008 · No Comments · 作者: admin 发表于:2008-01-29 19:09:51 最后更新于:2008-02-18 19:03:27 版权声明: 可以任意转载, 转载时请务必以超链接形式标明文章原始出处和作者信息。A look inside the WordPress database [...]

40.  [naturalhorsetrainer](#) Says:
[February 23, 2008 at 12:22 am](#)

Hi,

I'm trying find a way to export blogroll out of wp_links for one domain, and import into another domain with link categories intact from domain to domain.

I'm hitting a stumbling block on the FK constraint on the wp_term* structure on both sides and am now wondering why the wp_links category structure is wrapped up into the taxonomy for the rest of the WP installation instead of using the link_category field in the wp_link table?


My choices either seem to be to wipe multiple installations and start over with every link I ever want up front, or input blogroll links by hand on 14 installations... when a simple mysql export/import would do the job otherwise.

I'm using different categories in the blogroll for my sites, friends sites, business links, and keyworded text links... right now I have 48 links x 12

sites (576 manual edits on top of the original 48) I have to either input one link at a time, or edit one link at a time to change categories...

Is there a way to remove the foreign key links and work totally within the wp_links table? Why was it done this way? Has anyone come up with a nice way of working around this?


Thanks in advance :)

41.  *naturalhorsetrainer* Says:
[February 23, 2008 at 12:28 am](#)


I'm sorry, they are Rev 2.3.x installations.

42. [Cotralis » Blog Archive » A look inside the WordPress database](#) Says:
[February 26, 2008 at 8:56 pm](#)

[...] A look inside the WordPress database [...]


43.  *chrisber* Says:
[March 2, 2008 at 5:34 pm](#)

I am a relative newcomer to all of this, so please be kind if I'm way off base with something. I've created several static pages on my site, and it is my understanding that they appear in order based on the post ID in wp_posts. Is there a safe way to edit these ID's to alter the order of my pages? I assume that I would want to pick a new sequence of numbers starting 1 increment above the highest existing ID? Are there potential dangers that I am unaware of, or will my foolish actions bring about the end of the world as we know it?

44.  [Leonid Mamchenkov](#) Says:
[March 2, 2008 at 7:11 pm](#)

Chrisber,

when you edit the page, there is something called "page order". It's one of those boxes on the right. You can put a number there and then use this number to sort the pages. The actual way your pages appear (including the sorting) is controlled from the theme.

45.  *chrisber* Says:
[March 2, 2008 at 7:18 pm](#)

Thanks!

46. [atlantageek](#) Says:
[April 1, 2008 at 8:02 pm](#)

So is this totally out of date now. I am trying to write a plugin that will automatically generate categories and I see that my database contains wp_terms instead of wp_cat**. The other thing is there a set of public functions that I can call to generate a category instead of tweaking the database. Any help would be appreciated wince I am new to the WP world

47. [Talker](#) Says:
[April 11, 2008 at 9:16 pm](#)

Hi,

Am getting close to being comfortable with the crazy switch from ver 2.3.3 to ver 2.5. Now need to find out "how" to address "all" of a given blogs posts. I can access them through categories, but not all the actual posts "in" that blog show up for linear viewing, start to end. Am missing in setting up the address to access the complete blog! Hate being a "newbie" at anything, but thats what I am with WordPress.

Leave a Reply

Name (required)

Mail (will not be published) (required)

Website

XHTML:

You can use these tags: <abbr title=""> <acronym title=""> <blockquote cite=""> <cite> <code> <del datetime=""> <i> <q cite=""> <strike>

Submit Comment

« [Quick access to WordPress options](#)
[Automating WordPress customizations - the plugin way](#) »

[Blog at WordPress.com](#). | Theme: Andreas09 by [Andreas Viklund](#).