

Phorum Support Forums / Phorum Development

Best way to search thread metadata?

Skye N. / February 22, 2015 04:00AM

[Best way to search thread metadata?](#)

Hello, I'm writing a tagging module to allow users to tag threads with a simple array of strings. I'm using v5.2

The way I've done this in other modules is to store the array in the first post's metadata (parent_id = 0). Easy enough.

But, the challenge here is how to filter the thread list by those meta tags. I think I have two options:

1. Add tag support to search.php
2. Filter the topc list via list.php

Option #1 - Adding a new parameter to the search is easy enough, and there is already a hook "search_action" that allows a module to bypass the built-in phorum_db_search(). However, that would require writing a new phorum_db_search() as the existing DB layer functions don't have hooks. I don't need to combine tag search with the other search params, so that makes it fairly easy. However, I would have to somehow parse/search the "meta" column within the search SQL itself - very hacky/fragile

Option #2 - Modify the thread list to filter by tag(s). Looking at list.php, there doesn't seem to be a standard hook to bypass the built-in phorum_db_get_thread_list() - but it would be easy enough to add a new phorum_hook call. But, then I'm in the same position of writing a new phorum_db_get_thread_list() to SQL search the serialized meta data just like before.

Any tips or tricks or hints or thoughts on this? Do any other modules implement SQL meta data search that I can look at?

Thanks,
Skye

Skye Nott
[Corvus Digital](#)

Edited 1 time(s). Last edit at 02/22/2015 04:01AM by Skye N..

Skye N. / February 22, 2015 06:18AM

[Re: Best way to search thread metadata?](#)

I guess in the end, there is no good way to search the meta column so it's a non starter. The performance would be terrible and error prone

[\[stackoverflow.com\]](#)

So, I should just normalize the db and create a separate thread_id -> tag table from the start.

Skye Nott
[Corvus Digital](#)

Thomas Seifert / February 22, 2015 10:41AM

[Re: Best way to search thread metadata?](#)

Quote

Skye N.

I guess in the end, there is no good way to search the meta column so it's a non starter. The performance would be terrible and error prone

[stackoverflow.com]

So, I should just normalize the db and create a separate thread_id -> tag table from the start.

I was about to answer exactly that. The metadata is not really searchable, you should store it in a better searchable format.

Thomas Seifert

Phorum Development Team / Mysnip-Solutions.de

[Custom Phorum and general software development](#)
[worry-free Phorum Hosting](#)

Skye N. / February 27, 2015 12:56AM

[Re: Best way to search thread metadata?](#)

OK, the module is done!

I ended up adding a new hook "list_action" in list.php, but that's the only modification I needed to the base code.

You can check it out here, with the new topic list filter drop-down: [www.triumphexp.com]

I'm going to run it on that site for a while to iron out any bugs, before trying it on a bigger site.

If there's any interest, I could package it up and make the code public. But it does require some coding skills to install the new hook and quite a lot of template elements, and I don't have any time to provide support for the module.

Skye

Skye Nott

[Corvus Digital](#)

Skye N. / February 27, 2015 08:16PM

[Re: Best way to search thread metadata?](#)

It's too bad all the paging URL generation in list.php doesn't include all the passed \$PHORUM["args"], I had to redo all the paging in my hook because there was no way to extend or override the default list url templates. That's the only really messy part in the tags module....

Skye Nott

[Corvus Digital](#)

Orinic / April 20, 2016 09:00AM

[Re: Best way to search thread metadata?](#)

I have been trying to decide how to record metadata for a project I am working on. I had already decided to use a structure to keep my data together. I think the structure will work well for storing general and specific metadata.
