

# Unified Index

This applies to the [search](#) capabilities in Tiki, such as those used by [PluginList](#) or the search at [tiki-searchindex.php](#) which rely on a *search index*.

## Table of contents

- [Unified Index](#)
  - [Fields](#)
  - [Tracker Fields](#)
    - [Multilingual fields are indexed as multiple fields](#)
    - [Rating and related fields store as multiple fields](#)
    - [Items List and Item Link fields](#)
    - [Language of the tracker item](#)
  - [Rebuild search index](#)
    - [From the Tiki interface](#)
    - [From the command line](#)
      - [Basic command](#)
      - [Multitiki sites](#)
      - [Successful rebuild](#)
      - [Troubleshooting](#)
        - [MySQL limits for very big numbers of tracker fields](#)
          - [How to detect](#)
          - [Technical explanation \(thanks Victor\)](#)
          - [Solutions](#)
  - [Unified Index storage in the database \(MySQL full-text search\)](#)
    - [Duplicate unified-index table](#)
  - [Related](#)
  - [Developer Notes](#)

Tiki can support multiple search engines internally. Each of those will have different capabilities and limitations. The default engine should provide capabilities good enough for small and medium sites. Larger sites may need additional infrastructure to get the most performance. Please see: [Unified Index Comparison](#)

## Fields

Below is a matrix between the fields and the object types.

Legend:

X - Available

/ - Static value

? - Depends on the data

Field	Type	Tokenized*	Sortable	wiki page	forum post	blog post	article	file	trackeritem	sheet	comment	user	Available in Tiki version
-------	------	------------	----------	--------------	---------------	--------------	---------	------	-------------	-------	---------	------	---------------------------------

Field	Type	Tokenized*	Sortable	wiki page	forum post	blog post	article	file	trackeritem	sheet	comment	user	Available in Tiki version
<b>object_type</b>	Generic		X	X	X	X	X	X	X	X	X	X	7
<b>object_id</b>	Generic		X	X	X	X	X	X	X	X	X	X	7
<b>title</b>	Generic	X	X	X	X	X	X	X	X	X	?	X	7
<b>title_initial</b>	Generic			X	X	X	X	X	X	?	?		?
<b>title_firstword</b>	Generic			X	X	X	X	X	X	?	?		?
<b>language</b>	Generic		X	X			X		?				7
<b>creation_date</b>	Generic		X	X	X	X	X	X	X		X	X	7/15
<b>modification_date</b>	Generic		X	X	X	X	X	X	X	X	X		7
<b>contributors</b>	Generic			X	X	X	X	X	X	X	X		7
<b>description</b>	Generic	X		X			X	X		X			7
<b>contents</b>	Generic	X		X	X	X	X	X	X	X	X	X	7
<b>wiki_content</b>	Specific			X								X	7
<b>wiki_uptodateness</b>	Specific		X	X									7
<b>wiki_approval_state</b>	Specific			X									11
<b>post_content</b>	Specific				X								7
<b>post_snippet</b>	Specific				X								14
<b>parent_thread_id</b> <i>(not to be confused with parent_object_id)</i>	Specific				X								8
<b>root_thread_id</b>	Specific				X								14
<b>parent_contributors</b>	Specific				X								14
<b>blog_id</b>	Specific		X			X							7
<b>blog_excerpt</b>	Specific					X							7
<b>blog_content</b>	Specific					X							7
<b>topic_id</b>	Specific		X				X						7
<b>article_content</b>	Specific						X						7
<b>article_toplevel</b>	Specific						X						7
<b>article_subtitle</b>	Specific						X						7
<b>article_author</b>	Specific						X						9
<b>article_type</b>	Specific						X						9
<b>article_heading</b> <i>available as description</i>	Specific						X						9
<b>published</b>	Specific						X						13
<b>sitetitle</b>	Specific						X						13
<b>siteurl</b>	Specific						X						13
<b>gallery_id</b>	Specific		X					X					7
<b>filename_id</b>	Specific		X					X					7
<b>filetype</b>	Specific	X	X					X					7
<b>filesize</b>	Specific	X	X					X					15
<b>file_comment</b>	Specific							X					7
<b>file_content</b>	Specific							X					7
<b>tracker_id</b>	Specific		X						X				7
<b>tracker_status</b>	Specific		X						X				7

Field	Type	Tokenized*	Sortable	wiki page	forum post	blog post	article	file	trackeritem	sheet	comment	user	Available in Tiki version
<b>tracker_field_PERMNAME/ID</b> <i>(see below for more details)</i>	Specific	X	?						X				7
<b>sheet_content</b>	Specific									X			7
<b>comment_content</b>	Specific	X									X		7
<b>user_country</b>	Specific	X	X									X	10
<b>groups</b>	Specific	X	X									X	?
<b>hits</b>	Specific				X								15
<b>lastpost_title</b>	Specific				X								15
<b>lastpost_modification_date</b>	Specific				X								15
<b>lastpost_contributors</b>	Specific				X								15
<b>lastpost_post_content</b>	Specific				X								15
<b>lastpost_post_snippet</b>	Specific				X								15
<b>lastpost_hits</b>	Specific				X								15
<b>lastpost_thread_id</b>	Specific				X								15
<b>view_permission</b>	Internal			X			X			X			7
<b>parent_object_type</b>	Internal		X		X	X	X	X	X		X		7
<b>parent_object_id</b>	Internal		X		X	X	X	X	X		X		7
<b>parent_view_permission</b>	Internal				X	X	X	X	X		X		7
<b>global_view_permission</b>	Internal										X		7
<b>hash</b>	Internal			X									7
<b>url</b>	Internal			X							X		7
<b>categories</b>	Global			X	X	X	X	X	X	X	X		7
<b>deep_categories</b>	Global			X	X	X	X	X	X	X	X		7
<b>allowed_groups</b>	Global			X	X	X	X	X	X	X	X		7
<b>freetags</b>	Global			X	X	X	X	X	X	X	X		7
<b>freetags_text</b>	Global			X	X	X	X	X	X	X	X		7
<b>adv_rating_ID</b>	Global		X	X	X	X	X	X	X	X	X	X	7
<b>comment_count</b>	Global		X	X	X	X	X	X	X	X	X		8
<b>relations</b>	Global			X	X	X	X	X	X	X	X	X	8
<b>attachments</b>	Global			X	X	X	X	X	X	X	X		7
<b>attachment_contents</b>	Global	X		X	X	X	X	X	X	X	X		7
<b>geo_located</b>	Global			X	X	X	X	X	X	X	X	X	9
<b>geo_location</b>	Global			X	X	X	X	X	X	X	X	X	9
<b>visits</b>	Global		X	X				X					9.2

\*) Tokenized - as in decomposed in words for full text search

## Tracker Fields

In general, tracker fields are indexed as **tracker\_field\_PERMNAME/ID**. PERMNAME/ID is your tracker field permanent name or ID. However, many tracker field types have additional useful variants of the main field (see below) that are indexed for each field.

The indexing for tracker fields will vary depending on the field type. As a general rule, `tracker_field_PERMNAME/ID` will be used as the field and will be sortable. However, there are a few exceptions:

- Image and File fields are not indexed
- TextArea is not sortable

## Multilingual fields are indexed as multiple fields

- The main one (`tracker_field_PERMNAME/ID`) contains all languages
- `tracker_field_PERMNAME/ID_lang` contains one language only (`tracker_field_12_fr` for example)

## Rating and related fields store as multiple fields

- `tracker_field_PERMNAME/ID` contains the average
- `tracker_field_PERMNAME/ID_sum` contains the vote totals
- `tracker_field_PERMNAME/ID_count` contains the number of votes

## Items List and Item Link fields

- `tracker_field_PERMNAME/ID_text` contains the text instead of the IDs of the linked/listed items

## Language of the tracker item

- If a language field is set for the tracker item, that language is indexed as the item language, i.e. the **language** field.

**Some used in `buildQuery/tiki-searchindex.php` (need explanation on whether these are real fields or just helpers):**

**type:** refers to **object\_type**

**deep:** if this is set, **categories** will be considered **deep-categories**

**autocomplete:** Will search for items with title starting with this

## Rebuild search index

The index is stored in `temp/unified-index/`. While the rebuild is occurring, a directory `temp/unified-index-new/` will appear (This is to permit the existing index to be used until the new one is ready). If `temp/unified-index-new/` doesn't disappear after the indexing, something must have gone wrong. You can delete it and try the re-indexing again. You may want to run `sh setup.sh` to make sure the permissions are OK.

## From the Tiki interface

You can visit this url: **`tiki-admin.php?page=search&rebuild=now`** and search index will be rebuilt if the site is small. For medium to high load sites, you can do that from the command line.

# From the command line

The search index can be rebuilt from the command line, and since [Tiki9](#) can be run using a [Cron](#) job where the server runs the command automatically - see [Cron Job to Rebuild Search Index](#).

Below are the commands that may be used to rebuild the index.

You can also rebuild it using the unified console.php command, with the appropriate parameters. For example:

## Basic command

```
php console.php index:rebuild
```

<-- or -->

```
php console.php i:r
```

## Multitiki sites

For multitiki sites, you can rebuild with commands like:

```
php console.php index:rebuild --site=site1.example.com  
php console.php index:rebuild --site=site2.example.com  
...
```

## Successful rebuild

If the rebuild is successful a message like the following will be produced (for cron jobs, this can usually be sent to you via email as part of automatically running the command):

```
Started rebuilding index...  
Indexation  
wiki page: 150  
forum post: 67  
blog post: 412  
article: 61  
file: 1294  
trackeritem: 196  
comment: 0  
Rebuilding index done  
X-Powered-By: PHP/5.5.8  
Content-type: text/html; charset=utf-8
```

## Troubleshooting

If the rebuild is unsuccessful, instead of the above message you may get a message that indicates there has been an internal server error, or it may say "Rebuild in progress." This may be because the rebuild process uses more memory or takes more time than allowed by the server's php settings. Such settings can be changed on the fly as part of the rebuild command - examples of how to do this are shown below.

## Increase memory limit

One way to increase memory is to change the `memory_limit` php setting as follows (this example changes the memory limit to 4 gigabytes while the rebuild process is running):

```
php -dmemory_limit=4G console.php i:r --log
```

You could also direct php to use a specific `php.ini` file, where there may be a higher memory limit setting or no limit. In this case you would use the `-c` parameter followed by the path to the `php.ini` file, as in the example below:

```
php -c /etc/php5/cli/php.ini console.php i:r --log
```

## Increase maximum execution time

Getting an internal server error may indicate the rebuild process takes longer than the `max_execution_time` php setting. That can be increased as part of the command as shown below where the max execution time is set to 300 seconds, or 5 minutes. (This command is also increasing the memory limit as described above):

```
php -dmemory_limit=4G -dmax_execution_time=300 console.php i:r --log
```

## Force rebuild

When the rebuild is unsuccessful with a "Rebuild in progress" message, this usually means that the rebuild failed previously in the middle of the process, leaving a temporary folder called `temp/unified-index-new` on the server. When a new rebuild is started and the program sees this folder, it thinks there is a rebuild already in progress and will stop. You can either delete this folder before rebuilding again or include the `--force` parameter in the rebuild command as follows:

```
php -dmemory_limit=4G -dmax_execution_time=300 console.php i:r --force --log
```

## MySQL limits for very big numbers of tracker fields

How to detect

If *Unified Search* is configured with the *MySQL Full Text Search* engine, you might encounter a failure to reindex with the following symptoms:

- `php console index:rebuild` stops without displaying the list of indexing statistics:

```
php console index:rebuild
[14 mai 2020 09:26 EDT] Started rebuilding index...
Unified search engine: MySQL, version 10.1.45-MariaDB
(it takes a very long time and in the end nothing shows up)
```

If you investigate with producing a log it ends this way:

```
php console.php index:rebuild --log
[14 mai 2020 16:10 EDT] Started rebuilding index... logging to file:
temp/Search_Indexer_mysql_database_name_console.log
Unified search engine: MySQL, version 10.1.45-MariaDB
```

(it takes a very long time and in the end nothing shows up)

```
tail temp/Search_Indexer_mysql_database_name_console.log
...
2020-05-14T16:21:47-04:00 ERR (3): Indexing failed while processing "2512" (type
trackeritem) with the error "Could not perform index modification: Too many columns"
2020-05-14T16:21:47-04:00 ERR (3): WARNING: PDO::query(): SQLSTATE[HY000]: General error:
1117 Too many columns
{"code":null,"file":"/path/to/tikiroot/lib/core/TikiDb/Pdo.php","line":104}
2020-05-14T16:21:47-04:00 INFO (6): addDocument trackeritem 2513 {"memoryUsage":"46.5 MiB"}
2020-05-14T16:21:47-04:00 ERR (3): Indexing failed while processing "2513" (type
trackeritem) with the error "Could not perform index modification: Too many columns"
2020-05-14T16:21:47-04:00 ERR (3): WARNING: PDO::query(): SQLSTATE[HY000]: General error:
1117 Too many columns
{"code":null,"file":"/path/to/tikiroot/lib/core/TikiDb/Pdo.php","line":104}
2020-05-14T16:21:47-04:00 INFO (6): addDocument trackeritem 2514 {"memoryUsage":"46.5 MiB"}
```

Technical explanation (thanks Victor)

[+]

Solutions

The following options in [Control panels](#) → [Search](#) help for this situation:

- **MySQL use short field names**

*Due to frm file constraints, number of search fields that one index can hold is usually limited to about 1500. This can be exceeded if you have numerous tracker fields. Enabling this option will try to shorten the field names internally that should allow you to use 300-500 more fields. Switching this option requires full index rebuild.*

Summary: This makes no difference in how you use Tiki.

- **Don't index non searchable fields**

*Indexing will skip adding all tracker fields that are not marked as "searchable". This will free index space but also make it impossible to use those fields in search index queries.*

Summary: You need to review the *Searchable* property of you tracker fields. A full index rebuild will be necessary after changes in the tracker fields *Searchable* properties.

Hint: If the first option is enough for your site, you may ignore the second one.

## Unified Index storage in the database (MySQL full-text search)

By design the unified-index store data in the MySQL database using MyISAM no matter if your Tiki uses InnoDB or MyISAM (not all versions of InnoDB offer FULLTEXT).

## Duplicate unified-index table

For some reasons you may found that your database has several index (tables like index\_...alphanumeric...).

This happen when they were issues rebuilding the index or when moving and upgrading the database (Tiki upgrade).

You can delete those tables (always consider saving a backup of your data prior any work) and rebuild the unified-index. Tiki will recreate and relink everything properly. You can find information about the unified-index in use at the Search control panel, General settings under Unified search index.

## Related

- [Cron Job to Rebuild Search Index](#)
- [PluginList](#)
- [PluginCustomSearch](#)
- [Unified Search](#)

## Developer Notes

See [https://dev.tiki.org/Unified-Index#Developer\\_Notes](https://dev.tiki.org/Unified-Index#Developer_Notes).

alias names for this page

---

[Unified Search](#) | [UnifiedSearch](#) | [Enterprise search](#) | [Search Index](#) | [SearchIndex](#) | [UnifiedIndex](#) | [IndexRebuild](#) | [Index Rebuild](#)