

## Manticore Search

Manticore Search is an Open Source database that was created in 2017 as a continuation of Sphinx Search engine (Started in 2001). It's a modern, fast, light-weight and full-featured database with outstanding full-text search capabilities. <https://manticoresearch.com/>  
You can check out benchmarks at: <https://db-benchmarks.com/>

It is a [Unified Index](#) option starting in [Tiki25](#) (experimental) and will be production ready in [Tiki26](#).  
[Manticore Search is very appreciated by its community](#).

As of 2022-11-16, the results between the various engines are not identical in all cases. We will write a script to compare the results of MariaDB/MySQL vs Elasticsearch vs Manticore Search, and review all the discrepancies. Also, different engines have different features. If Manticore doesn't work for you, you can revert to your previous engine.

### Installation

See instructions on how to install it server side:

- <https://manticoresearch.com/install/>
- <https://wikisuite.org/Manticore-Search>

How to indicate to Tiki to use Manticore Search

Once Manticore is installed and running:

1. Pick Manticore as **Unified search engine** at `tiki-admin.php?page=search`
2. Rebuild index
  - Small sites, via GUI
  - Medium and large sites, via CLI `php console.php index:rebuild --log`
3. Enjoy!

Please see development information: [Manticore Search](#) and <https://wikisuite.org/Manticore-Search>

It has many features including:

- [Faceted search](#)
- [Stored Search](#)

See also

- [Manticore Buddy](#)

Dev notes from Victor

In short:

1. Introducing a new setting in manticore search settings - you can enumerate the fields you want to be always indexed - e.g. important fields or fields that search too slowly with regex. Default here is title and contents - you will always want contents be in the full-text index for the whole site search.
2. Keep the search setting of the tracker fields as it works until now - work in combination with the preference to exclude non-searchable fields from the index.
3. When building the index, we try to fit all text fields as full-text but if this is more than 255, we prefer the preferenced ones and then start with the rest until we fill the limit. If we go over the limit, the rest are indexed as string columns and allow regex search.

4. Changed multivalue attributes to MVA native manticore implementation to spare some fulltext fields.

So, most of the sites shouldn't notice this change. Bigger sites will probably need some configuration to allow faster lookups on big trackers - ensure the searched fields end up in the preference.

Mac OS&nbsp;Notes

The only way I could get this to run locally on macOS (on Apple Silicon) was by using Docker, and i needed to add these config items, pasted here for reference)

### **Environment**

PATH

/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin

GOSU\_VERSION

1.11

MCL\_URL

[https://repo.manticoresearch.com/repository/manticoresearch\\_focal/dists/focal/main/binary-amd64/manticore-columnar-lib\\_1.15.4-220522-2fef34e\\_amd64.deb](https://repo.manticoresearch.com/repository/manticoresearch_focal/dists/focal/main/binary-amd64/manticore-columnar-lib_1.15.4-220522-2fef34e_amd64.deb)

DAEMON\_URL

[https://repo.manticoresearch.com/repository/manticoresearch\\_focal/dists/manticore\\_5.0.2-220530-348514c86\\_amd64.tgz](https://repo.manticoresearch.com/repository/manticoresearch_focal/dists/manticore_5.0.2-220530-348514c86_amd64.tgz)

LANG

C.UTF-8

LC\_ALL

C.UTF-8

### **Mounts**

/tmp

/tmp

/var/lib/manticore/

Users/username/data

### **Ports**

9306/tcp

0.0.0.0:9306

9308/tcp

0.0.0.0:9308

9312/tcp

0.0.0.0:9312