

# HelpMiscellaneous

## Other [MoinMoin](#) help topics

### Subtopics

- [/FrequentlyAskedQuestions](#)
- [/ExperimentalFeatures](#)

### Contents

- [Other MoinMoin help topics](#)
  - [Adding an icon to your wiki setup](#)
  - [Enabling a publically editable intermap](#)
  - [Apache tricks](#)
    - [Securing access to a wiki via Apache means](#)
    - [Serving the wiki from a domain root](#)
  - [Special features for wiki farms and intranets](#)
    - [Centralized settings](#)
    - [Remapping URLs](#)
  - [TODO](#)

## Adding an icon to your wiki setup

You maybe want to add some icon to your wiki site, commonly known as `favicon.ico`. To get those little icons besides bookmarks and with tabbed browsing, add this to your `moin_config.py`:

```
html_head = '''
<link rel="icon" href="/images/icon.png" type="image/png">
<!-- ... more header stuff ... -->
'''
```

The image size *must* be 16x16 and simply putting it as `/favicon.ico` (into your DocumentRoot) does *not* work with all browsers (e.g. does not work with Mozilla versions 1.0 - 1.3).

## Enabling a publically editable intermap

If you want your users to easily add new [InterWiki](#) monikers, change your `moin_config.py` to contain an entry like this:

```
shared_intermap = ['/usr/local/moin/mywiki/data/text/intermap_2etxt']
```

Then create a page named `intermap.txt` (see [LinuxWiki:intermap.txt](#) for an example) with the following content:

```
#format text/plain
## Keep the first line and add text only after two hash marks (##)
##
## This is a publically editable extension of standard InterWiki
## definitions. Any addition here is immediately reflected in
## the wiki.
##
## To enable this in your own wiki, either add the page file of
## this page to config.shared_intermap (can be a list of files
## since version 1.1), or add an appropriate symbolic link.
##
## The format of an entry is interwiki moniker followed by the URL
## $PAGE can be used to embed the URI following the moniker in
## the resultant URL

MyNewMoniker http://example.com/somewiki/$PAGE.html
```

Then check whether your new monikers appear on the [InterWiki](#) list.

## Apache tricks

### Securing access to a wiki via Apache means

See [MoinMoin:PasswordProtectedEditing](#)

## Serving the wiki from a domain root

httpd.conf modification for [MoinMoin](#) installation at "/:

```
# global section:
LoadModule rewrite_module /usr/lib/apache/1.3/mod_rewrite.so
# virtualhost section:
RewriteEngine on
RewriteRule ^/wiki/ - [last]
RewriteRule ^/robots.txt - [last]
RewriteRule ^/favicon.ico - [last]
RewriteRule ^/?(.*) /path/to/cgi-bin/moin.cgi/$1 [last,type=application/x-httpd-cgi]
```

This way you can use short and clean URLs like <http://LinuxWiki.de/FrontPage>.

## Special features for wiki farms and intranets

The following shows how certain features can help you to optimize your configuration when you run several wikis on one host, or when you use [MoinMoin](#) in intranet environments.

### Centralized settings

- farm\_config.py
- shared intermap.txt
- shared code base and icons

### Remapping URLs

The configuration variable "url\_mappings" can be used to remap absolute URLs to relative ones, or change certain URL prefixes to totally different ones. This is especially useful in intranets, when whole trees of documents hosted outside of the wiki move around.

"url\_mappings" is a lookup table in the form of a dict, the key is the prefix to be replaced, the value is the replacement. A typical example is `url_mappings = {'http://my.server.net/': '/'}`, which removes the scheme from local URLs, and thus makes links to your own server work for both `http` and `https`.

## TODO

⚠ TODO: explain the following things in detail, and add more detail to the above sections

Needing more explanation:

- event logging
- Page trails (user option)
- [UserPreferences](#): checkboxes for double-click edit, page trail, fancy links, emoticons, jump to last page visited, and some other yes/no options
- "Remove trailing whitespace from each line" option in the editor
- Page templates (create a new page using a template page)
- Only "normal" URLs (outside of brackets) are converted to inline images ==> document exact rules
- Highlighting of search results, and of bad words when spellchecking
- Support for "file:" URLs
- email notification features
- "RSS" icon displayed by [RecentChanges](#) macro
- commentary field of editor, recent changes and page info
- Page caching, for now limited to XSLT processing (where it's absolutely needed); new code & API to add the "RefreshCache" link
- inline search fields (at the bottom of each page)
- the interwiki name "Self" now always points to the own wiki
- page info action

Admin stuff:

- describe "moin.cgi?test" tool in more detail
- WikiFarm support:
  - `<datadir>/plugin/macro` and `<datadir>/plugin/action` can be used to store macros and actions local to a specific wiki instance
  - `config.shared_intermap` can contain a pathname to a shared "intermap.txt" file (i.e. one stored outside the datadir)
- `$PAGE` placeholder in interwiki map

Command-line tools:

- "moin-dump" command line tool to create a static copy of the wiki content
- though not extensively tested, the standalone server now does POST requests, i.e. you can save pages; there are still problems with persistent global variables! It only works for Python >= 2.0.

Compatibility:

- Statistical features are NOT designed to work with Python 1.5.2 and require Python 2.0 or higher. Overall, [MoinMoin](#) 0.11 is not explicitly tested for 1.5.2 compatibility. The lowest *recommended* Python version is 2.0!
- RSS features require a recent PyXML (CVS or 0.7) due to bugs in the namespace handling of xml.sax.saxutils in earlier versions. This is (hopefully) automatically detected on every installation.