



Speedster for Magento 1

Online Version

An online version of this user manual can be found [here](#).

Quick Links

This User Manual is structured in the following sections:

[Install + Set Up \(User Manual\)](#)

2

Install + Set Up (User Manual)

1. Please ensure you read and understand all instructions

If you do not understand something in the set-up instructions provided, do not install Fooman Speedster. Instead, contact your Magento developer and ask them to install the extension for you. If all steps are not followed correctly, the installation of Fooman Speedster could fail and generate errors on your site.

2. Install Fooman Speedster first on your test site, before installing it on your live site

If any errors do occur, it is far preferable to complete our troubleshooting instructions on your test site so your live site will remain unaffected.

Fooman Speedster is a free extension which has been tested with the default Magento themes and successfully installed on thousands of Magento stores. However, there is always a risk that minifying and combining Javascript files can produce Javascript errors.

We do not recommend using Fooman Speedster without thorough testing on a test site if:

- Significant Javascript customisation work has been done on your store and the added non-default Javascript files do not follow best practice Javascript (you can test with <http://jshint.com/>)
- Your store already contains minified files (including files minified by standard Magento). All Javascript on your site should be un-minified before installing Fooman Speedster - combining files that have been minified separately is bad news!
- Your store uses JQuery (or has extensions installed which use JQuery) - although a workaround is provided, almost 100% of reported errors with Fooman Speedster are from stores which also use JQuery
- You are not using Magento's default theme

3. Always test your site for existing Javascript errors before installing Fooman Speedster

Fooman Speedster relies on all Javascript on your store being completely error free. If your store contains existing Javascript errors, these will be magnified when installing Speedster and could cause issues with your store's frontend.

The Speedster self-test does not include testing for existing Javascript errors on your site. It is important to do this separately before installing Fooman Speedster (you can test with <http://jshint.com/>). If you identify any existing Javascript errors on your site, ensure these are completely fixed before installing Fooman Speedster.

4. Ensure your site meets all system requirements

Fooman Speedster works out of the box using an Apache platform run on Linux. In addition, system requirements are:

- Mod_rewrite (a standard Apache module installed by default on most Linux servers) must be enabled
- .htaccess must be supported
- CSS images must be available under the same domain as the store
- No existing Javascript errors on the store

Several developers have also successfully installed Fooman Speedster on different platforms by adapting the installation process. For further reading see the [ISAPI2](#) and [Nginx](#) threads.

Set up Speedster in the Magento backend

Go to **System > Configuration > Fooman Extensions > Fooman Speedster**.

Fooman Speedster



General settings

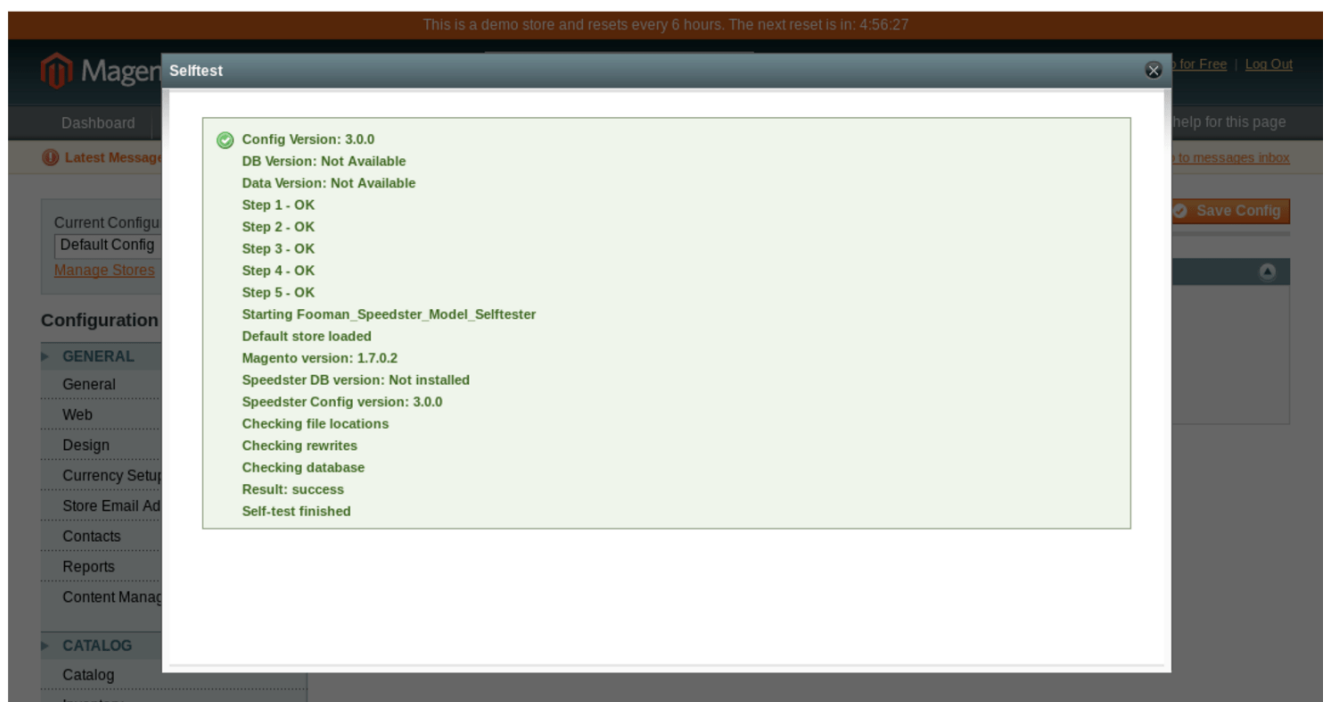
☒ Enabled

Enabled

To enable Fooman Speedster, select “Yes”.

This will automatically run the Speedster self-test, which includes verification steps 1-4 outlined in the next section of this User Manual.

Fooman Speedster will only be enabled if the self-test runs successfully:



The self-test will highlight the most common issues that people have when configuring Fooman Speedster. It does not highlight all possible issues because it is not possible to fully automate this process.

Once you have enabled Fooman Speedster, proceed to verification steps 5-7 to complete

these additional checks. Verification step 5 can be performed automatically via a further self-test (see next page), but steps 6 and 7 must be performed manually.

Other notes:

- Make sure that `/lib/minify/m.php` is executable (permissions like 755 on the file itself and the containing folder should work) and `/var/minifycache` is writeable
- Speedster has its own merging mechanism, please disable Magento's default Merging feature under System > Configuration > Developer > Javascript and CSS
- Minifying Javascript and CSS takes a while to compute. This only needs to be done once per Javascript/CSS combination and is then written to a cache. To prime the cache for a faster customer experience, browse each page type (homepage, product page, etc) in your store once
- Important: Don't disable the output of Fooman_Speedster under Configuration > Advanced - it will make your site unusable. To uninstall, either use Magento Connect or edit `/app/etc/modules/Fooman_Speedster.xml` (change true to false)

Step 1 - Verify Fooman Speedster Output is Enabled

Verify that the output of Fooman_Speedster under Configuration > Advanced is enabled.

Never disable this output - it will make your store unusable (this is a limitation of Magento's implementation of the disable feature).

TEST 2 - VERIFY PERMISSION SETTINGS

Step 2 - Verify Permission Settings

To verify that you have the correct permission settings to run Fooman Speedster, visit the url: <http://YOURMAGENTOWEBSITE/lib/minify/m.php> (be sure to adapt the url to include your own domain).

This test should return a blank page with only these words:

"HTTP/1.0 404 Not Found".

If you see this message, permission settings are working correctly and you can move to the next verification test.

If you do not see this message:

Internal Server Error Message

The most common error message you will see is the "Internal Server Error" message, which is caused by incorrect permission settings. Change your permission settings to allow the file

to be executed by the web server. Apply the same setting to the folder `/lib/minify` and the file `m.php`.

Note: Permission requirements differ from server to server, but most often permissions 755 or 775 will work. If in doubt, check which permissions are working for your main `index.php` in the Magento root folder.

Magento 404 Error Message (Fully Styled)

If redirected to Magento's fully styled, '404 Page Not Found' page, this indicates a permission issue. Please contact your Server Administrator or Hosting company to determine why the file is not accessible to you.

Other Error Messages

If you encounter any other error messages, consult your server's error log for clues about the error source and refer to the appropriate troubleshooting step in this Help Centre.

STEP 3 - VERIFY MINIFICATION

Step 3 - Verify Minification

To verify that minification is working correctly:

Test A

Visit the url:

<http://YOURMAGENTOWEBSITE/lib/minify/m.php?f=/skin/frontend/default/default/css/print.css> (be sure to include your domain name in the URL)

This test should return the `print.css` file in a condensed form in your browser.

Test B

Open `/var/minifocache` (via FTP) to inspect the file system. You should find new files starting with `minify_` which have been written to `/var/minifocache`.

If both tests return the expected result, minification is working correctly and you can move to the next verification test.

If either test does not return the expected result see the suggestions below:

Permission Settings (Test A and B)

Check that you have the correct permission settings for the cache directory. Check permissions on `/var/minifocache`. Change your permission settings to allow the web server to write files.

Note: Permission requirements differ from server to server, but most often permissions 755 or 775 will work. If in doubt, check which permissions are working for your main index.php in the Magento root folder.

Garbled Output (Test A only)

Check whether the output of Test A is garbled. Garbled output could mean that CSS files are being compressed twice. Most often, this can be eliminated by turning off zlib compression for this folder. Edit /lib/minify/.htaccess and follow the additional instructions given.

[STEP 4 - VERIFY URL WRITING](#)

Step 4 - Verify URL Writing

To verify that the URL is being rewritten, visit

<http://YOURMAGENTOWEBSITE/skin/m/1232681243/skin/frontend/default/default/css/print.css> (be sure to include your domain name in the URL)

This test should return the print.css file in a condensed form in your browser (the identical output from [verification step 3](#)).

If this test returns the expected result, URL rewriting is working correctly and you can move to the next verification test.

If this test does not return the expected result:

Apache Support

Ensure that your server supports Apache rewrite rules. Refer to the system requirements for installing Fooman Speedster.

Issues with the Rewrite Rule

Check that you have the correct file permissions for the skin/m/.htaccess file. Change your permission settings to allow the server to read and process the file. The easiest solution is to compare the permission settings to the .htaccess file in your main Magento folder, and make them the same.

Multi Store Setups

For multi store setups, refer to [these instructions](#).

[STEP 5 - VERIFY FULL PROCESS](#)

Step 5 - Verify Full Fooman Speedster Process

Step 5 can be performed manually via the version of the self-test accessed from **System > Configuration > Fooman Extensions > Support**. This self-test is available once the initial self-test has run successfully and the Fooman Speedster has been successfully enabled.

This step can also be performed manually.

To verify the full Fooman Speedster process, visit <http://YOURMAGENTOWEBSITE> (be sure to include your domain in the URL) and view the HTML source of the page.

Copy and paste the link addresses for the CSS and/or Javascript files (found in between <head> and </head> within the HTML source) into your browser address bar. Do this for a minimum of one CSS file and one Javascript file.

This test should return the requested files in a condensed form in your browser.

If this test does not return the expected result:

Permission Settings

If you can't see anything in between <head> and </head>, this usually signals a permission issue. Change permissions on /app/code/community/Fooman/Speedster and all containing files to enable the server to read the files. Most often permissions 755 or 775 will work.

Verify Fooman Speedster Output is Enabled

Check whether Fooman Speedster has been disabled under **System > Configuration > Advanced**. Check all three scopes: default, website and store. Refer to [verification step 1](#).

Disable Magento Compilation Mode

If you currently have Magento's compilation mode enabled, disable the Fooman Speedster extension (see instructions below), then disable compilation mode. Reinstall Fooman Speedster with Magento's compilation mode disabled.

[STEP 6 - VERIFY CSS](#)

Step 6 - Verify Your Store's CSS

This step must be performed manually after installation to verify that the extension is working correctly and to correct any troubleshooting errors.

Browse your store to verify that the styling hasn't changed.

One common issue is the use of @import directives in the original CSS files. During the minification process, the @import linked files end up being loaded first. If this is causing issues, remove the @import directives and instead load the files via an explicit layout xml instruction.

Step 7 - Verify Your Store's Javascript

Step 7 must be performed manually after installation to verify that the extension is working correctly and to correct any troubleshooting errors.

Browse your store to verify no new Javascript errors have appeared.

If nothing comes up, Fooman Speedster is working as expected.

Any new errors are most likely caused by either trying to minify already minified files or minifying javascript files that do not follow best practice coding style for Javascript.

To fix the double minification problem, replace the already minified file with the original non-minified javascript. A javascript file with .min.js in its file name is always a sure sign that this file has been minified.

To find the origin of remaining Javascript errors, we suggest testing all Javascript files which are not part of the Magento base/default theme on jslint.com for code quality.

[TROUBLESHOOTING ARTICLES](#)

Failed self-test result

If the self-test does not run successfully, Fooman Speedster cannot be enabled.

You will see an error message which tells you which verification and troubleshooting step/s failed.

❗ Selftest failed with: Step 1 - OK

Step 2 - OK

Step 3 - OK

Verify URL Rewriting ERROR: The minified result is larger than the original (Step 4)

[Please read the manual for details.](#)

Starting Fooman_Speedster_Model_Selftester

Default store loaded

Magento version: 1.7.0.2

Speedster DB version: Not installed

Speedster Config version: 3.0.0

Checking file locations

Checking rewrites

Checking database

Result: success

Self-test finished

Fooman Speedster
✔ Save Config

General settings
⬆

Enabled

No

[STORE VIEW]

The self-test will show you which steps (1-4) were passed or failed. You must correct every error to pass all tests 1-4 before you are able to enable Fooman Speedster.

Refer to the verification steps 1-4 (below) for directions on how to do this. If the self-test failed, you will see an error message which tells you which verification step(s) failed. Start the verification testing at this step.

The self-test will also highlight if you are running another extension which directly conflicts with Fooman Speedster. If a direct extension conflict is identified, you should either [disable Fooman Speedster](#) or the other conflicting extension. For JQuery Base by Mxperts and Canonical URLs by Yoast, refer to the identified workarounds in the [Fooman Support Centre](#).

HTTP Password Protection Issues with Self-Test

We've had an occasional report that the self-test can fail if you have HTTP password protection installed. Essentially the self-test tries to access your site as a client requesting the page. If you haven't made special provisions for the server itself to be excluded from your password challenge, it will not be able to access the self-test.

Workaround for Canonical URLs by Yoast (Yoast version 1.3 and below)

If you are using Yoast version 1.3 and below, you need to edit:

/app/code/community/Yoast/CanonicalUrl/Block/Head.php

Change

```
class Yoast_CanonicalUrl_Block_Head extends Mage_Page_Block_Html_Head
```

to

```
class Yoast_CanonicalUrl_Block_Head extends Fooman_Speedster_Block_Page_Html_Head
```

Workaround for JQuery Base by Mxperts

Edit

/app/code/local/Mxperts/Jquery/Block/Page/Html/Head.php

Change

```
class Mxperts_Jquery_Block_Page_Html_Head extends Mage_Page_Block_Html_Head
```

to

```
class Mxperts_Jquery_Block_Page_Html_Head extends Fooman_Speedster_Block_Page_Html_Head
```

Further advice on JQuery:

Jquery duplicates Magento's inbuilt Javascript library prototype which can add to page load times. If you are concerned about site speed, we recommend replacing extensions that use JQuery as a first step in optimising your site speed. Try finding a replacement solution which is based on the prototype library that comes with default Magento.

Disabling Fooman Speedster

Edit /app/etc/modules/Foومان_Speedster.xml (change true to false). Save and refresh the Magento cache.

Don't try to disable the output of Foومان_Speedster under Configuration > Advanced - it will make your site unusable! (This is a limitation of Magento's implementation of the disable feature).

Does Foومان Speedster add any settings or admin menus?

From version 3.0 onwards, Foومان Speedster adds a global on/off switch under **System > Configuration > Foومان Speedster**. Earlier extension versions do not add any settings or

admin menus to your Magento store.

Reporting Any Issues/Bugs

We are proud of our quality extension code - it's been widely tested and we stand by it 100%. If something does happen and you think you might be experiencing an issue or bug, please contact us via support@fooman.co.nz and we will help you out.