

How to install the extension

1. Backup your store's database and web directory.
2. Login to the SSH console of your server and navigate to the root directory of the Magento 2 store.
3. Copy the installation instructions from the page [My Downloadable Products](#) to the SSH console and press ENTER.
4. Run command `php -f bin/magento module:enable Mirasvit_Core Mirasvit_CacheWarmer Mirasvit_Report` to enable the extension.
5. Run command `php -f bin/magento setup:upgrade` to install the extension.
6. Run command `php -f bin/magento cache:clean` to clean the cache.
7. Deploy static view files

```
rm -rf pub/static/*; rm -rf var/view_preprocessed/*; php -f bin/magento setup:static-content:deploy
```

Page Cache Warmer settings

The settings of Cache Warmer are located at **System ? Page Cache Warmer ? Settings** and grouped into the following sections:

- [General Settings](#) - basic configuration of warming process.
- [Performance](#) - allows you to manage performance of Cache Warmer.
- [Extended Settings](#) - advanced settings to fine-tune server load and improve its responsiveness.
- [FPC hole punch](#) - settings for excluding individual blocks from caching.
- [Debug](#) - provides you with additional debug information about cache.

General Settings

- **Enabled** - enables/disables Page Cache Warmer.
- **Ignored Pages** - allows to set URL patterns for pages that should be ignored by Cache Warmer. Consists of records, where **Expression** contains regular expression for URL.

Example

- `/\?/` - ignore all pages, that have GET params (filtered category pages, search results pages, etc.)
- `/\?[\w=]*&[\w=]*&[\w=]*` - ignore pages with more than 2 GET params

[Read more](#) about regular expression.

- **Ignored User-agents** - allows to set patterns of user-agent headers, which will be ignored by cache warmer. For example, you can set `/*Googlebot.*` to ignore Google crawler. And pages visited by Google crawler will not be added to the queue of cache warmer.

Performance

This section will help you to tune-up Page Cache Warmer crawler. The crawler is launched regularly based on cron schedule and begins to traverse the list of available URLs of the store:

- If URL is already cached, it is skipped;
- If URL is not in the cache, crawler will visit it on behalf of external user so that the page is rendered and placed in the cache.

To adjust the performance of crawler you can use the following options:

- **Threads** - the number of concurrent (parallel) requests to the store.
- **Delay between requests** - delay between requests in milliseconds.
- **Maximum job run time** - maximum running time allowed for crawler job in seconds.
- **Job Schedule** - Cron expression for schedule warmer jobs. Read more about setting Cron expressions [here](#).

Extended Settings

- **Don't run warmer job if average system load more than (%)** - If average system load exceeds value in this field, warmer job will not start. Average system load graph can be seen at **SYSTEM->Page Cache Warmer -> Jobs**. Read more in [this FAQ section](#).
- **Crawler limit** - Crawler will stop crawling when quantity of URLs in cache reaches this value.
- **Forbid cache flushing** - completely disables flushing FPC cache. Read more in [this FAQ section](#).
- **Run warmer as web server user** - Enable this option only if cron user does not have enough permissions to warm pages.
- **Forcibly make pages cacheable** - Allows to override default behavior of Magento with regard to caching pages by changing content block attributes:
 - **Disable** - keeps Magento's default caching behavior.
 - **All pages** - extension will delete `cacheable="false"` for all blocks in home, category and product pages making them cacheable as a result.
 - **Configure** - allows to select page types which will be made cacheable, eg. Home page (**Cms · Index · Index**), category (**Catalog · Category · View**) and product pages (**Catalog · Product · View**)

Image not found or type unknown

- **Warm mobile pages separately** - Enable only if you want your store to create different cached pages for mobile and desktop users.
- **Use the same cache for new visitor** - Allows to avoid creation of several copies of cached pages for Magento installations with multiple storeviews. Disable it in case if your pages for different languages/currencies/storeviews are cached incorrectly.

Debug

- **Show cache status** - show block in the store frontend with page cache status.
- **Hole punch info** - show hint for 'hole punched' blocks. Read more in [FPC hole punch](#) section.
- **Show cache status and hole punch info only for** - ?omma separated IP addresses, for which info box should be visible.
- **Log tags for cache flushing** - logs all cache flushes to `/var/log/clean.log` file.
- **Log backtrace for cache flushing** - logs backtrace for cache flushes to the `/var/log/clean.log`. For more info on this and previous setting, read more in [this FAQ section](#).

Cache Warmer Rules

Go to **System > Page Cache Warmer > Rules**. You will see a list of warming rules. You can edit rules or add your own.

Warming Rules

Image not found or type unknown

The cache warmer will process rules one by one, sorted by their priority.

Each rule has following fields:

- **Name** - you can set any name
- **Is Active**
- **Priority** - rules with higher priority will be processed in first turn.
- **Page Conditions** - you can set any filters
- **Request Modifiers** - you can modify HTTP headers of request.
- **Vary Data Modifiers** - you can modify VaryData of request. For example, if you would like to

warm pages of logged-in customers, you may add the following:

```
customer_group: 2
```

Cache Warmer Pages

Go to **System > Page Cache Warmer > Pages**. You will see warmed pages.

- **Warm Cache** - add page to FPC cache.
- **Clean Cache** - delete page from FPC cache.
- **Delete** - delete url from pages list.

cache_warmer_pages.png

Cache Warmer Jobs

Go to **System > Page Cache Warmer > Jobs**. You will see Cache Warmer Jobs.

cache_warmer_jobs.png

Image not found or type unknown

Configuring "Hole Punch"

"Hole Punch" is a special feature of our Cache Warmer extension, which allows you to exclude from caching individual blocks.

This way most part of page can still be cached except of selected blocks (so-called *holes*).

It is extremely useful, when you have complex pages, that are constructed from multiple templates and multiple blocks, and can seriously optimize your store.

Note

WARNING: Hole Punch feature works primarily for standard Magento and CMS pages and blocks. It can not be applied to blocks, which are dynamically generated or loaded with AJAX, that is:

- Embedded product lists, such as Top Sells, or similar;
- Cart Widgets;
- AJAX-dependant dynamic blocks;
- Javascript-generated blocks;
- Complex blocks, added by third-party extensions.

How to Configure "Hole Punches"

Since "hole punch" is micro-level blocks management, it's configuration is a little tricky. Carefully follow these steps to configure them:

1. Run SSH console, proceed to your store's root, and check your store's mode with `bin/magento deploy:mode:show`.
2. If mode is `production` switch your store to developer mode with `bin/magento deploy:mode:set developer` command.
3. Navigate to **System -> Cache Management** and disable ALL cache.
4. Navigate to **FPC Hole Punch** section of **Stores -> Configuration -> Mirasvit Extensions -> Page Cache Warmer**, and you will see a configuration table. Press **Add** to configure new hole punch, and enter the following parameters:
 - **Template** - path to template, which contains excluded block.
 - **Block Class** - block class, which can be seen using path hints (see below).
 - **CMS Block ID** - if this block is custom, put there it's ID, that can be seen at **Content -> Blocks**.
 - **Store** - storeview, for which block excluding should be applied.
5. If you do not know templates, pick them using hints. Navigate to **Store -> Configuration -> Advanced -> Developer -> Debug** section, and enable the following options:
 - **Enabled Template Path Hints for Storefront** - it will display paths for templates, which consist your frontend page.
 - **Add Block Names to Hints** - it will display block names.
6. Open frontend page, where you need to have *hole punches*. You should see something like shown below:

Image not found or type unknown

White-colored labels should be put to **Template**, blue-colored to **Block Class** options.

Example

- **Template:** `../../../../src/vendor/magento/module-catalog/view/frontend/templates/product/view/form.phtml`
- **Block Class:** `Magento\Catalog\Block\Product\View\Interceptor`

7. To check, whether hole punch actually work, revert settings from step 5 (if you did them), then navigate to **Debug** section **Stores -> Configuration -> Mirasvit Extensions -> Page Cache Warmer**, and set the following options:

- **Show cache status** - it will display cache info block on frontend pages.
- **Hole punch info** - it will add to info block hole punch information.
- **Show debug hints only for** - if you wish info box to display only for your developers, put there

IP of their workstation.

8. Clear cache, and open target frontend page to make it go to the cache (you will see this time **Page miss** in info block). Then refresh page, and you should see simultaneously **Page Hit** message, but **Miss** at specific block - that one, which you had excluded.

cache-miss.png

Image not found or type unknown

9. Revert options, enabled in steps 7. Switch store back to to production mode with `bin/magento deploy:mode:set production`, and purge ALL cache again to apply changes in this mode.

FPC Reports

Note

This functionality will work only for Built-in mode (will not work for Varnish).

Go to **System > Page Cache Warmer > FPC Reports**. You will see FPC Reports.

Average Response Time - it is average time for Time To First Byte with cache and without cache.

Hit average response time - it is average time for Time To First Byte for pages which in cache.

Miss average response times - it is average time for Time To First Byte for pages without cache.

fpc_report_avg.png

Image not found or type unknown

Hit Count - quantity for visited cached pages

Miss Count - quantity for visited pages which is out of cache

Visit Count - quantity for visited pages (in cache and out of cache)

fpc_report_count.png

Image not found or type unknown

Time To First Byte - it is time which you can see on following image:

tffb.png

Image not found or type unknown

More about Time To First Byte you can read here https://en.wikipedia.org/wiki/Time_To_First_Byte

Command Line Interface

Usage: `php -f bin/magento [options]`

- `mirasvit:cache-warmer --warm` - warm the pages which you can see in SYSTEM->Page Cache Warmer depending on Performance configurations
- `mirasvit:cache-warmer --remove-all-pages` - remove all pages from the module
- `mirasvit:cache-warmer:crawl` - crawl all store pages
- `mirasvit:cache-warmer:crawl --cycle_limit 5` - set cycle limit for crawl command, where default value is 100 (from version 1.0.53)
- `mirasvit:cache-warmer:crawl --unlock` - unlock if was finished incorrectly (from version 1.0.53)

- `mirasvit:cache-warmer:crawl --all` - crawl all store pages (from version 1.0.42)
- `mirasvit:cache-warmer:crawl --store-id 2` - crawl all pages for store with ID 2
- `mirasvit:cache-warmer:crawl --base-url http://example.com` - crawl all pages for store with base url <http://example.com>
- `mirasvit:cache-warmer:crawl --ignore-query` - crawl all pages except pages with query params (?)
- `mirasvit:cache-warmer:test` - for purpose of a module test

Frequently asked questions

- [What Page Cache warmer do?](#)
- [What can I do if all pages have "Pending" status?](#)
- [How check if my pages are cacheable?](#)
- [How can I make pages cacheable?](#)
- [How can I stop cache flushing?](#)
- [How can I control average system load?](#)

What Page Cache warmer does?

Page Cache Warmer is a crawler (powered by Magento Cron job) which visits uncached pages like user and adds those pages to the cache. More pages will be in cache, more faster will be store in a whole.

It can be used either by Magento standard Cache or Varnish. It also works with other cache types (like Redis) without any problem.

Note

If you use Varnish, please, make sure, that it is configured correctly. Refer [here](#) for more info.

What can I do if all pages have "Pending" status?

Our extension will work with your store **only** if Varnish is configured correctly or if you had switched caching mode to Built-in. This mode is governed by **STORES -> Configuration -> ADVANCED -> System -> Full Page Cache -> Caching Application** option, which can have two cases:

Case 1: Varnish:

1. If you had set caching mode to **Varnish Cache (Recommended)**, make sure, that Varnish is installed on server. Use this command for that: `varnishd -V`.

If result is correct you should see varnish version, as shown below.

varnishd_v.png

Image not found or type unknown

2. Switch caching mode to **Built-in mode** and set **Show cache status** to **Yes** at **STORES -> Configuration -> Mirasvit Extensions -> Page Cache Warmer -> Debug** section. It will show, whether your pages are actually cached. If you visit frontend Category or Product page, you should see **Page Hit** message, as shown below:

hit.png

Image not found or type unknown

If the page is not cached, you will see **Page Miss**, and a list of not cacheable blocks:

miss_blocks.png

Image not found or type unknown

In this case refer to [How can I make pages cacheable?](#) section for further actions.

3. If you see **Page Hit** on most pages, then most likely Varnish is not configured to work with magento. You can check using [this way](#).

Case 2: Built-in Cache:

1. If you had set caching mode to **Built-in Cache**, then set **Show cache status** to **Yes** at **STORES -> Configuration -> Mirasvit Extensions -> Page Cache Warmer -> Debug** section.
2. Visit frontend Category or Product pages of your store to check, whether they're actually cached.
3. If you see there **Page Miss**, you need to fix caching settings. Refer to [How can I make pages cacheable?](#) for further steps.

How to check if my pages are cacheable?

1. Switch store to **Built-in Cache** at **STORES -> Configuration -> ADVANCED -> System -> Full Page Cache -> Caching Application**

, and set **Show cache status** to **Yes**, as in previous case.

2. Visit frontend Category or Product pages of your store to check, whether they're actually cached. If the page is cached, you will see **Page Hit** message, as shown below:

hit.png

Image not found or type unknown

If you see **Page Miss** message, with list of not cached blocks, as shown below

miss_blocks.png

Image not found or type unknown

Then the pages are not actually cached. Refer to [How can I make pages cacheable?](#) section for further steps.

How can I make pages cacheable?

Variant 1 (from version 1.1.21)

1. Make sure, that page is not cached, using **Show cache status** option, set to **Yes**, and special infobox, which appears after it:

miss_blocks.png

Image not found or type unknown

2. Set **Forcibly make pages cacheable** option to **Yes**, at **Stores -> Configuration -> Mirasvit Extensions -> Page Cache Warmer -> Extended Settings**.
3. Purge **ALL** in Cache Management and revisit page with **Show cache status** still enabled. You should see **Page Hit** message:

hit.png

Image not found or type unknown

Variant 2

1. Make sure, that page is not cached, using **Show cache status** option, set to **Yes**, and special infobox, which appears after it:

miss_blocks.png

Image not found or type unknown

2. Connect to your server via SSH, and run the following command:

```
grep -r "[your_block_name].*cacheable" * --include="*.xml"
```

where [your_block_name] is block, which you can pick in section **Not cacheable blocks**. It will give you the list of layout files, that govern that block.

Example

Typical SSH session with layout files listing shall look like below. Let the block name is `authorization-link-login`

```
cd [magento_root_folder]/app/code
grep -r "authorization-link-login.*cacheable" * --include="*.xml"
cd [magento_root_folder]/app/design
grep -r "authorization-link-login.*cacheable" * --include="*.xml"
```

Output should look like this:

```
[custom_theme]/Magento_Customer/layout/default.xml <block class="Magento\
```

3. Make sure, that layout does not forbid caching explicitly.

Open files, which you had obtained in previous step, and search for block record, which should have the following expression in its header:

```
name="BLOCK_NAME"
```

BLOCK_NAME is the same name, as in previous step. Remove attribute `cacheable="false"` from that record, if it exists.

4. Purge **ALL** in Cache Management and revisit page with **Show cache status** still enabled. You should see **Page Hit** message:

hit.png

Image not found or type unknown

How can I stop cache flushing?

Our Cache Warmer extension do not have means for cache flushing.

However, you can see, how often your store's cache is flushed using Fill Rate graph, available at **System -> Page Cache Warmer -> Pages**.

fill_rate.png

Image not found or type unknown

If flushing is performed too often, you can use logs to see its reasons. To enable the, use options below at **Stores -> Configuration -> Mirasvit Extensions -> Page Cache Warmer -> Debug** section:

- **Log tags for cache flushing** - puts to the log each flush schedule.
- **Log backtrace for cache flushing** - puts to the log modules, which start flushing.

Result will be recorded to the file `/var/log/clean.log`.

The most origin of flushing are reindex (FPC also flushes cache on reindexing), or re-saving products (FPC also flushes frontend pages, which are associated with product).

Products also can be manipulated using external WebAPI services, which also save them, and thus - trigger cache flushing.

In the latter and similar cases you need to configure your system to not save products too often and trigger

reindex not more than once per day.

If it is impossible for some reason, you can disable flushing completely by setting **Forbid cache flushing** option to **Yes**, at **Stores ->Configuration -> Mirasvit Extensions -> Page Cache Warmer -> Extended Settings**.

Note

If you had enabled **Forbid cache flushing**, changes at Category and Product pages won't apply unless you manually purge cache from **System -> Cache Management**, or cache will expiry (depends from TTL set in Magento settings).

How can I control average system load?

Our extension do not have means for system load balancing.

But at **SYSTEM->Page Cache Warmer->Jobs** we have graph **Average system load**, which will help you to discover bottlenecks.

avg_system_load.png

Image not found or type unknown

You can also reduce system load by setting threshold limit for Cache Warmer jobs with option **Don't run warmer job if average system load more than (%)** at **Stores ->Configuration -> Mirasvit Extensions -> Page Cache Warmer -> Extended Settings** section.

If this option is set, whenever warming jobs are started, our extension will check current load level, and if will be equal or greater than entered load percentage - jobs won't start.

Recommended values for this option - 70 or 80.

How to upgrade extension

To upgrade the extension follow these steps:

1. Backup your store's database and web directory.
2. Login to the SSH console of your server and navigate to the root directory of the Magento 2 store.
3. Run command `composer require mirasvit/module-cache-warmer:* --update-with-dependencies` to update current extension with all dependencies.

Note

In some cases the command above is not applicable, it's not possible to update just current module, or you just need to upgrade all Mirasvit modules in a bundle. In this case command above will have no effect.

Run instead `composer update mirasvit/*` command. It will update all Mirasvit modules, installed on your store.

4. Run command `php -f bin/magento setup:upgrade` to install updates.
5. Run command `php -f bin/magento cache:clean` to clean the cache.
6. Deploy static view files

```
rm -rf pub/static/*; rm -rf var/view_preprocessed/*; php -f bin/magento setup:static-content:deploy
```

Disabling the Extension

Temporarily Disable

To temporarily disable the extension please follow these steps:

1. Login to the SSH console of your server and navigate to the root directory of the Magento 2 store.
2. Run command `php -f bin/magento module:disable Mirasvit_CacheWarmer` to disable the extension.
3. Login in to the Magento back-end and refresh the store cache (if enabled).

Removing the Extension

To uninstall the extension please follow these steps:

1. Login to the SSH console of your server and navigate to the root directory of the Magento 2 store.
2. Run the command `composer remove mirasvit/module-cache-warmer` to remove the extension.
3. Login in to the Magento back-end and refresh the store cache (if enabled).

Change Log

1.2.8

(2018-12-05)

Fixed

- Error during log cleanup (affects only 1.2.7, 1.2.6)
 - In debug mode, warmer adds debug html to non-html pages (like /robots.txt)
-

1.2.7

(2018-12-04)

Improvements

- Support of Magento 2.3

Fixed

- In some cases, extension can broke ajax calls of 3rd party extensions
 - Warming rules for customer groups
 - Improved clean of trace logs
-

1.2.5

(2018-11-29)

Fixed

- Issue with rendering admin for M 2.1
 - Magento 2.1, error Class Magento\Framework\Serialize\Serializer\Json does not exist (affects from 1.2.0)
-

1.2.4

(2018-11-28)

Improvements

- Changed condition for remove old logs

Fixed

- Issue with adding pages to list
-

1.2.3

(2018-11-23)

Fixed

- Error if I run crawl via SSH. Affects only 1.2.2
-

1.2.2

(2018-11-22)

Fixed

- When I try to warm cache via admin, I receive error: "Argument 2 passed to Mirasvit\CacheWarmer\Service\WarmerService::warmCollection(..". Affects only 1.2.0.
-

1.2.0

(2018-11-09)

Features

- Cache clean trace
- Warm Rules (Priority System)

Improvements

- Show crawler error if customer has http authorization

Fixed

- Incorrect calculation of cache hits rate
 - M2.1 some pages are not warmed
 - If our toolbar is enabled and file robots.txt is missing in the customer's store, we incorrectly added some html tags to request by URI /robots.txt.
 - 301 as error in log
-

1.1.62

(2018-10-29)

Fixed

- Issue with stored serialized vary_data

Warmer can't correctly warm URLs which were added to the queue.

1.1.61

(2018-10-26)

Improvements

- CURL for pages
-

1.1.60

(2018-10-25)

Improvements

- CURL information to trace
 - Skip warming, if page cache is disabled
-

1.1.59

(2018-10-24)

Improvements

- Added ability use with HTTP auth

Fixed

- Issue with compilation
-

1.1.58

(2018-10-24)

Improvements

- Bolt FPC compatibility
-

1.1.57

(2018-10-24)

Improvements

- Remove page only after 3 unsuccessful warm attempts
-

1.1.56

(2018-10-22)

Fixed

- Unserialize issue
-

1.1.55

(2018-10-17)

Improvements

- Charts
 - Cache clean logger
-

1.1.53

(2018-10-11)

Improvements

- Refactoring
- Trace

Fixed

- Time issue in trace
 - Crawler does not follows redirects
-

1.1.52

(2018-09-24)

Fixed

- Fixed out of memory crashes

Improvements

- Added option "Ignored User-agents"
 - Stop job execution after 3 errors
-

1.1.51

(2018-08-31)

Improvements

- Clear old logs by cron (30 days)
-

1.1.50

(2018-08-30)

Fixed

- Slightly improved TTFB
-

1.1.49

(2018-08-30)

Improvements

- Use the same cache for new visitor
-

1.1.48

(2018-08-27)

Fixed

- Fixed and error: "Read timed out after 60 seconds..." (for some stores)
-

1.1.47

(2018-08-27)

Improvements

- Coverage rate

Fixed

- Fixed "Invalid URI supplied" error (for urls more than 255 symbols)
 - Fixed rate limit depending from "Crawler limit"
 - Fixed and error: "Read timed out after 60 seconds {"exception":"[object] (Zend_Http_Client_Adapter_Exception(code: 1000): Read timed out after 60 seconds at .../vendor/magento/zendframework1/library/Zend/Http/Client/Adapter/Socket.php" (for some stores)
-

1.1.46

(2018-08-20)

Features

- Ability show cache status for Varnish
- Fixed popularity for Varnish

Fixed

- Info about extensions which can broke Magento Page Cache
-

1.1.45

(2018-08-13)

Improvements

- Info about extensions which can broke Magento Page Cache
-

1.1.44

(2018-08-03)

Fixed

- Fixed an error: "Unable to Connect to ssl:..."
 - Fixed an error if FPC Hole Punch is enabled for widget
-

1.1.43

(2018-07-26)

Fixed

- Fixed an error during compilation
-

1.1.42

(2018-07-26)

Fixed

- Fixed an issue with "Flush Magento Cache" from admin panel if "Forbid cache flushing" set to "Yes"

Improvements

- Ability set protocol in cli warmer
 - Forbid return from cache an empty page and add in cache an empty page
-

1.1.41

(2018-07-24)

Fixed

- Fixed cli warmer if option "Warm mobile pages separately" is enabled
-

1.1.40

(2018-07-13)

Fixed

- Fixed minor bug (for some stores)
-

1.1.39

(2018-07-13)

Fixed

- Fixed issue with page type disappearing
-

1.1.38

(2018-07-12)

Fixed

- Fixed crawler if "Add Store Code to Urls" set to "Yes"
-

1.1.37

(2018-06-28)

Improvements

- Make pages cacheable by page type
-

1.1.36

(2018-06-23)

Improvements

- Warm pages by page type order
-

1.1.35

(2018-06-23)

Fixed

- Fixed crawling for urls without domains
-

1.1.34

(2018-06-15)

Improvements

- Use empty vary data for mobile pages
-

1.1.33

(2018-06-14)

Features

- Ability warm mobile pages separately
-

1.1.32

(2018-06-13)

Improvements

- Ability prewarm stores depending from currency
-

1.1.31

(2018-06-11)

Fixed

- Delete old(not used) cli options
-

1.1.30

(2018-06-11)

Fixed

- Delete old(not used) cli options

Improvements

- TTL info
-

1.1.29

(2018-05-18)

Fixed

- Small css fix

1.1.28

(2018-05-08)

Fixed

- Ability clear cache by url for products and categories in Pages listing if Varnish installed
-

1.1.27

(2018-05-07)

Fixed

- bug: Fixed large jobs list with status "Scheduled"
-

1.1.26

(2018-05-03)

Fixed

- Fixed ability run a lot of warm processes
-

1.1.25

(2018-05-02)

Fixed

- Fixed parse error
-

1.1.24

(2018-05-02)

Fixed

- Fixed an error: "Undefined property in .../vendor/mirasvit/module-cache-warmer/src/CacheWarmer/Service/WarmerService.php on line 128"
-

1.1.23

(2018-04-26)

Fixed

- Fixed "Forbid Cache Flushing" option for Varnish
-

1.1.22

(2018-04-25)

Fixed

- Fixed presence not cacheable pages in list
-

1.1.21

(2018-04-20)

Features

- Forcibly make pages cacheable
-

1.1.20

(2018-04-20)

Fixed

- bug: Fixed an error: "Warning: strpos(): Empty needle in .../app/code/Mirasvit/CacheWarmer/Service/BlockMarkService.php on line 108" if "FPC hole punch" enabled without "Template" or "Block class" fields
- Fixed compilation error "Errors during compilation: Mirasvit\CacheWarmer\Service\DebugService..."
- Use also secure base url in cli warmer

Improvements

- The same version for Magento 2.1 and Magento 2.2
-

1.1.19

(2018-04-11)

Fixed

- Fixed multi store crawling (from cli)
-

1.1.18

(2018-04-05)

Fixed

- Fixed error while feed generation
-

1.1.17

(2018-04-04)

Improvements

- Ability crawl incorrect html content from command line

Fixed

- Fixed re-crawling links
-

1.1.16

(2018-03-23)

Improvements

- Run warmer as web server user
-

1.1.15

(2018-03-22)

Improvements

- Crawl speed improvement
-

1.1.14

(2018-03-07)

Fixed

- Fixed incorrect X-Magento-Vary
-

1.1.13

(2018-03-05)

Fixed

- Cannot instantiate abstract class
-

1.1.12

(2018-03-02)

Improvements

- Automatically using "Don't verify peer" function

Fixed

- Fixed compatibility with varnish and fastly
 - Ability run setup:di:compile without database
-

1.1.11

(2018-02-22)

Fixed

- Fixed an error "PHP Fatal error: Class 'Mirasvit\Report\Model\Query\Column\Date\Range' not found in .../CacheWarmer/Reports/Query/Column/Date/Range.php on line 19"
-

1.1.10

(2018-02-22)

Improvements

- Switched to new module-report version
-

1.1.9

(2018-02-12)

Fixed

- Fixed an error "Notice: unserialize(): Error at offset 255 of 255 bytes in .../app/code/Mirasvit/CacheWarmer/Model/Job.php on line 66"
-

1.1.8

(2018-02-02)

Fixed

- Fixed an error "sh: sysctl: command not found"
-

1.1.7

(2018-01-29)

Fixed

- Fixed notice (Notice: Undefined property: Mirasvit\CacheWarmer\Plugin\Debug\OnMissPlugin::\$request in ...vendor/mirasvit/module-cache-warmer/src/CacheWarmer/Plugin/Debug/OnMissPlugin.php on line 96)
-

1.1.6

(2018-01-23)

Fixed

- Fixed Magento error "(InvalidArgumentException): Unable to serialize value." when incorrect content and json_encode return false
-

1.1.5

(2018-01-19)

Fixed

- Fixed an issue with block excluding when template have module class different from block
-

1.1.4

(2018-01-18)

Fixed

- Fixed minor bug in text
 - Fixed the error appeared while using command "unlock" in cli [Exception] Warning: unlink(cache-warmer.cli.crawl.lock): No such file or directory in .../app/code/Mirasvit/CacheWarmer/Console/Command/CrawlCommand.php on line 217
-

1.1.3

(2018-01-05)

Fixed

bug: Fixed incorrect message (for magento 2.2.*)

1.1.2

(2018-01-04)

Improvements

improve: Message about incorrect version (if extension installed without composer)

1.1.1

(2017-12-08)

Fixed

- Fixed cli error if sysctl command not found
-

1.1.0

(2017-12-07)

Fixed

- Fixed cli command error

Documentation

- Documentation improvement
-

1.0.63

(2018-04-05)

Fixed

- Fixed error while feed generation
 - Fixed re-crawling links
-

1.0.62

(2018-03-29)

Improvements

- Switched to new module-report version
-

1.0.61

(2018-03-07)

Fixed

- Fixed incorrect X-Magento-Vary
-

1.0.60

(2018-03-02)

Improvements

- Automatically using "Don't verify peer" function

Fixed

- Ability run setup:di:compile without database
-

1.0.59

(2018-02-12)

Fixed

- Fixed an error "Notice: unserialize(): Error at offset 255 of 255 bytes in .../app/code/Mirasvit/CacheWarmer/Model/Job.php on line 66"
-

1.0.58

(2018-02-02)

Fixed

- Fixed an error "sh: sysctl: command not found"
-

1.0.57

(2018-02-02)

Fixed

- Fixed an error "sh: sysctl: command not found"
-

1.0.56

(2018-01-19)

Fixed

- Fixed an issue with block excluding when template have module class different from block
-

1.0.55

(2018-01-18)

Fixed

- Fixed minor bug in text
- Fixed the error appeared while using command "unlock" in cli [Exception] Warning: unlink(cache-

warmer.cli.crawl.lock): No such file or directory in .../app/code/Mirasvit/CacheWarmer/Console/Command/CrawlCommand.php on line 217

1.0.54

(2017-12-08)

Fixed

- Fixed cli error if sysctl command not found
-

1.0.53

(2017-12-07)

Fixed

- Fixed cli command error

Documentation

- Documentation improvement
-

1.0.52

(2017-11-29)

Fixed

- Fixed cli error
-

1.0.51

(2017-11-29)

Fixed

- Fixed cli error
-

1.0.50

(2017-11-29)

Fixed

- Magento 2.2 compatibility
-

1.0.49

(2017-11-28)

Fixed

- Magento 2.1 compatibility
-

1.0.48

(2017-11-28)

Improvements

- LiteMage compatibility
-

1.0.47

(2017-11-27)

Fixed

- Fixed admin load time issue.
-

1.0.46

(2017-11-24)

Fixed

- Extended config recurring update
-

1.0.45

(2017-11-21)

Improvements

- Refactoring

Fixed

- Recurring json update
-

1.0.44

(2017-11-17)

Fixed

- Fix an error in console if use setup:install
 - Fixed an error when get cpu count
-

1.0.43

(2017-11-07)

Fixed

- Minor stability fix
-

1.0.42

(2017-11-03)

Improvements

- Stability improvement

Fixed

- Fixed an error "Cache frontend 'default' is not recognized." (for some stores)

Documentation

- Documentation update
-

1.0.41

(2017-10-18)

Fixed

- Magento 2.2 compatibility
-

1.0.40

(2017-10-17)

Features

- Hole punching for blocks
-

1.0.39

(2017-09-28)

Fixed

- Magento 2.2 compatibility

Documentation

- Documentation improvement
-

1.0.38

(2017-08-09)

Fixed

- Fixed an error with comman warm in command line
-

1.0.37

(2017-08-04)

Documentation

- Documentation update
-

1.0.36

(2017-08-04)

Fixed

- Minor compatibility adjustments introduced

Documentation

- Documentation update
-

1.0.35

(2017-07-26)

Fixed

- Minor compatibility adjustments introduced
-

1.0.34

(2017-07-24)

Improvements

- Menu improvement

Fixed

- Fixed an error
-

1.0.33

(2017-07-24)

Improvements

- Refactoring
-

1.0.32

(2017-07-18)

Fixed

- Fixed an error
-

1.0.31

(2017-07-17)

Fixed

- Fixed an error "Area code is already set"
-

1.0.30

(2017-07-10)

Fixed

- Fixed an error
-

1.0.29

(2017-07-10)

Fixed

- Fixed an error
-

1.0.28

(2017-07-07)

Improvements

- Ability crawl particular store

Fixed

- Fixed issue with maximum job run time
 - Compatibility with TemplateMonster
-

1.0.27

(2017-06-27)

Fixed

- Fixed popularity calculation for internal requests
-

1.0.26

(2017-05-15)

Documentation

- Documentation update
-

1.0.25

(2017-05-15)

Improvements

- Report period filter
-

1.0.24

(2017-04-25)

Fixed

- Fixed ability enable report for custom user roles
-

1.0.23

(2017-04-24)

Features

- Reports
-

1.0.22

(2017-04-18)

Improvements

- Ability to run/remove warmer jobs manually

Fixed

- Issue with vary string
-

1.0.21

(2017-04-13)

Improvements

- Added lock file, that not allows to run parallel warmer processes
 - Replaced product/category observer to plugins
-

1.0.20

(2017-04-12)

Improvements

- Speed up popularity logging
-

1.0.19

(2017-03-24)

Fixed

- Fixed an issue with compilation
-

1.0.18

(2017-03-24)

Improvements

- Performance of fill rate feature
-

1.0.17

(2017-03-23)

Improvements

- Crontab & User Interface

Fixed

- Fixed an issue with fill report
-

1.0.16

(2017-03-21)

Improvements

- Added mass actions to grid with pages

Fixed

- Fixed an issue with warming by page type
-

1.0.15

(2017-03-15)

Improvements

- Changed default configuration
- Automatically removing not valid pages from warmer list

Fixed

- Fixed an issue with checking cache status for Varnish
-

1.0.14

(2017-02-22)

Improvements

- Added console command for test warming features
-

1.0.13

(2017-02-01)

Fixed

- Fixed compilation issue
-

1.0.12

(2017-01-27)

Fixed

- Added compatibility with old SEO version
-

1.0.11

(2017-01-26)

Fixed

- Fixed security issue
-

1.0.10

(2017-01-25)

Fixed

- Fixed an issue with observer event

Improvements

- Warm jobs
-

1.0.8

(2017-01-10)

Improvements

- Disable information toolbar for ajax requests (JSON output)
-

1.0.7

(2016-12-29)

Fixed

- Fixed an error if Cache-Control object not exist
-

1.0.6

(2016-12-28)

Documentation

- Improvement
-

1.0.5

(2016-12-28)

Improvements

- Compatibility with M2.2

Documentation

- Added new docs
-

1.0.4

(2016-12-16)

Fixed

- Fixed an issue with 404 pages error during crawling (CLI)

Documentation

- Settings
-

1.0.3

(2016-12-13)

Improvements

- Updated information and appearance of info block [Screenshot](#)
-

1.0.2

(2016-11-24)

Improvements

- Compatibility with SEO version 1.0.34
-

1.0.1

(2016-11-07)

Features

- Info block which help check if page in FPC cache
-

Getting Started

Welcome to the **Page Cache Warmer** documentation. Whether you are a new or an advanced user, you can find some useful information here.

First of all we recommend to check the following link:

- [How to install extension](#)
- [Page Cache Warmer configuration](#)