Contents

Setup:Installation Guide/Advanced

A quality version of this page, approved on *21 December 2021*, was based off this revision.

Special:PrefixIndex/Setup:Installation Guide/Advanced/

Setup:Installation Guide/Advanced/Backup

A quality version of this page, approved on *3 December 2020*, was based off this revision.

Backup of the database

To back up your database, use the native program "mysqldump" of MySQL/MariaDB: Open a console, switch to the directory `<installpath-bluespice>` and backup your database with the following command:

```
mysqldump -u <username> -p --lock-tables <datenbank> > database.sql
```

For `<username>` , use the administrative database user (usually "root"), and for `<datenbank>` , use the database of your MediaWiki/BlueSpice installation.

> **Note:** In Windows, make sure that your environment variables are set correctly, so that you can access the command "mysqldump".

Backup of the file system

Back up the complete directory `<installpath-bluespice>` , which now also includes the database image, to a different location on your server.

For security reasons, immediately delete the file `<installpath-bluespice>/database.sql`, since it can be accessed from the browser.

For Advanced Users: Configuration Folder settings.d

A quality version of this page, approved on *3 September 2020*, was based off this revision.

Contents

Tip for theDocument

- Please, read this manual completely and work through the single installation steps one after another.
- For editing the configuration files with a text editor, the files must be saved in UTF-8 coding without BOM (Byte Order Mark).
- The placeholder `<installpath-bluespice>` stands for the path to your BlueSpice installation, e.g `C:\inetpub\wwwroot\bluespice` (Windows) or `/var/www/bluespice` (Linux).

This document provides background information for advanced users. We recommend not to make any changes on BlueSpice, if you are not familiar with the following information.

Explanation For the Folder settings.d

Starting with BlueSpice version 2.27.1, all default settings as well as the modules of BlueSpice are outsourced. All configurations can now be found in separate files in the /settings.d folder.

The files in this folder integrate the standard MediaWiki extensions required for BlueSpice, perform recommended default configurations, and also integrate all BlueSpice-related extensions (free or pro).

The files in the /settings.d folder are automatically integrated in alphabetical order and are therefore prefixed with numbers.
If you want to add additional configurations yourself, you can save your own files here. For this, it is recommended to use the prefix 001-.

Additional configurations, which are not packed per default but can be used, can be found at auf github.com.

Update-safe changing of default configuration files

If you want to change the files delivered by default in the settings.d folder, we strongly recommend that you choose the following method so as not to lose this configuration after an update:

- Create a copy of the respective configuration file with the extension `.local.php` - e.g. *020-VisualEditor. local.php*
- Only edit this file

Files with this file extension are generally preferred in the loading process.

ExtendedSearch configuration

A quality version of this page, approved on *2 December 2020*, was based off this revision.

If the ElasticSearch service is up and running, you can enable BlueSpice Extended Search.

- Go to your BlueSpice install path. For example:

```
cd /var/www/bluespice
```

- Go to the folder *settings.d*:

```
cd settings.d
```

- Create a file in this folder and name it *020-BlueSpiceExtendedSearch.local.php*.
- Put the following content in the newly created file.

```
<?php
wfLoadExtension( 'BlueSpiceExtendedSearch' );
$GLOBALS['wgSearchType'] = 'BS\\ExtendedSearch\\MediaWiki\\Backend\\BlueSpiceSearch';
```

- Go to your BlueSpice install path again. For example:

```
cd /var/www/bluespice
```

- Run the following scripts to build the search index.

```
php extensions/BlueSpiceExtendedSearch/maintenance/initBackends.php --quick
php extensions/BlueSpiceExtendedSearch/maintenance/rebuildIndex.php --quick
php maintenance/runJobs.php
```

Done! The extended search functionality is now active.

Setup:Installation Guide/Advanced/Performance Optimization

A quality version of this page, approved on *29 November 2019*, was based off this revision.

- PHP
- MySQL (MariaDB)
- ManualRecache of the LanguageCache

Performance Optimization: ManualRecache of LanguageCache

A quality version of this page, approved on *29 November 2019*, was based off this revision.

To prevent Rebuilding LocalizationCache with every page call, apply the following configuration:

> ⚠️ When following these instructions, make sure that the $wgCacheDirectory variable is set in advance. This is already the case in the BlueSpice standard delivery in the file settings.d/005-Directories.php.

> ⚠️ This performance optimization achieves the best performance by enabling and configuring opcache in PHP.

In the `settings.d` folder of your codebase, create the file 006-ManualRecache.php with the following contents:

```php
<?php
$wgLocalisationCacheConf = [
        'class' => LocalisationCache::class,
        'store' => 'array',
        'storeClass' => false,
        'storeDirectory' => $wgCacheDirectory,
        'manualRecache' => true,
];
```

Then delete all existing files in the `cache` folder of your codebase. Then manually re-create the LocalizationCache by entering the following on your console:

```
php /pfad/zur/installation/maintenance/rebuildLocalisationCache.php --force
```

Finally, make sure that this is done regularly in the background via cronjob ("Task Scheduler" under Windows). We recommend this twice a day, for example at 6am and 6pm.

Performance Optimization: MySQL (MariaDB)

A quality version of this page, approved on *21 July 2020*, was based off this revision.

Match MySQL or MariaDB directly to your memory and CPU cores. To do this, apply the following configuration in your my.ini:

```
[mysqld]

; with 16GB RAM
innodb_buffer_pool_size=4096M
tmp-table-size=1024M
max-heap-table-size=1024M
query_cache_size=1024M

; with 12 CPU cores
innodb-buffer-pool-instances=12
max_connections=12000

; Only use "127.0.0.1", not "localhost" when accessing MySQL
; Remember to also make these changes in the $wgDBserver variable in LocalSettings.php.
skip-name-resolve
```

Performance Optimization: PHP

A quality version of this page, approved on *29 November 2019*, was based off this revision.

Contents

opcache

Enable the Zend extension opcache in PHP. In the php.ini, we recommend the following configuration:

```
opcache.enable=1
opcache.memory_consumption=512
opcache.max_accelerated_files=100000
opcache.validate_timestamps=1
opcache.revalidate_freq=2
opcache.optimization_level=0x7FFF9FFF
```

In addition, you should include the BlueSpice configuration files (`extensions /BlueSpiceFoundation/config/*`) in the Opcache blacklist (`opcache.blacklist_filename`). You can find more information in the official PHP documentationfor configuration.php.

zlib

Activate the extension zlib. In the php.ini we recommend the following configuration:

```
zlib.output_compression = On
zlib.output_compression_level = 9
```

You can find more information in the official PHP documentation for zlib.

Rebuilding the search index

A quality version of this page, approved on *20 September 2019*, was based off this revision.

Contents

In some cases it is needed to rebuild the search index. This document will show you the steps to do this.

Tips for this Document

- Please, read this manual completely and work through the single installation steps one after another.
- The placeholder `<installpath-bluespice>` stands for the path to your BlueSpice installation, e.g `C:\inetpub\wwwroot\bluespice` (Windows) or `/var/www/bluespice` (Linux).

Rebuilding the search index

Please open a command line on your system and change to `<installpath-bluespice>`.
Execute the following command:

```
php extensions/BlueSpiceExtensions/ExtendedSearch/maintenance/searchUpdate.php (Linux)
php extensions\BlueSpiceExtensions\ExtendedSearch\maintenance\searchUpdate.php (Windows
))
```

With Windows, please take care that the environment variables are set correctly to access the php command.

Depending on the size of your contents and uploads this can take a while.

VisualEditor configuration

A quality version of this page, approved on *21 October 2020*, was based off this revision.

Parsoid is the rendering service for VisualEditor. Make sure to install and configure Parsoid before configuring VisualEditor. See the instructions for Windows and Linux.
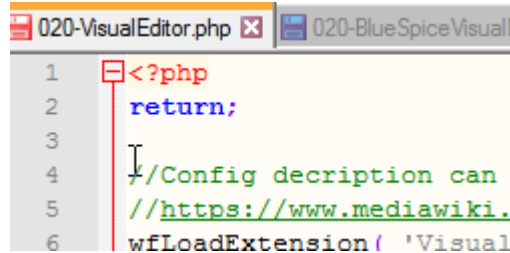
Contents

Server settings

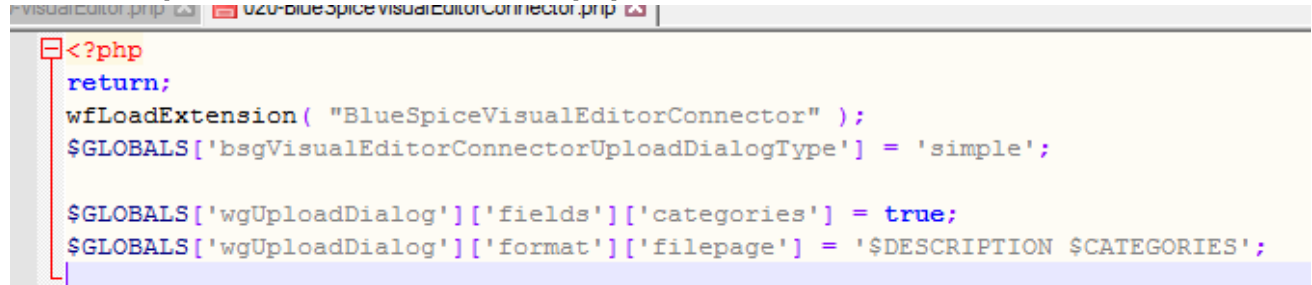Before you can use VisualEditor in BlueSpice, you have to activate it in your installation:

In the BlueSpice installation directory, locate the following files inside the folder „var\www\settings.d":

020-VisualEditor.php:

020-BlueSpiceVisualEditorConnector.php:

1. Remove the line containing the „return;" command in both files and save them.
2. Restart your web server (recommended).

Checking the Configuration manager

In case the VisualEditor is still not appearing, check if VisualEditor is shown as activated in the configuration manager of your wiki:

1. Go to *Global actions > Config manager*
2. Make sure the checkbox "Enable VisualEditor for BlueSpice extensions" is activated:
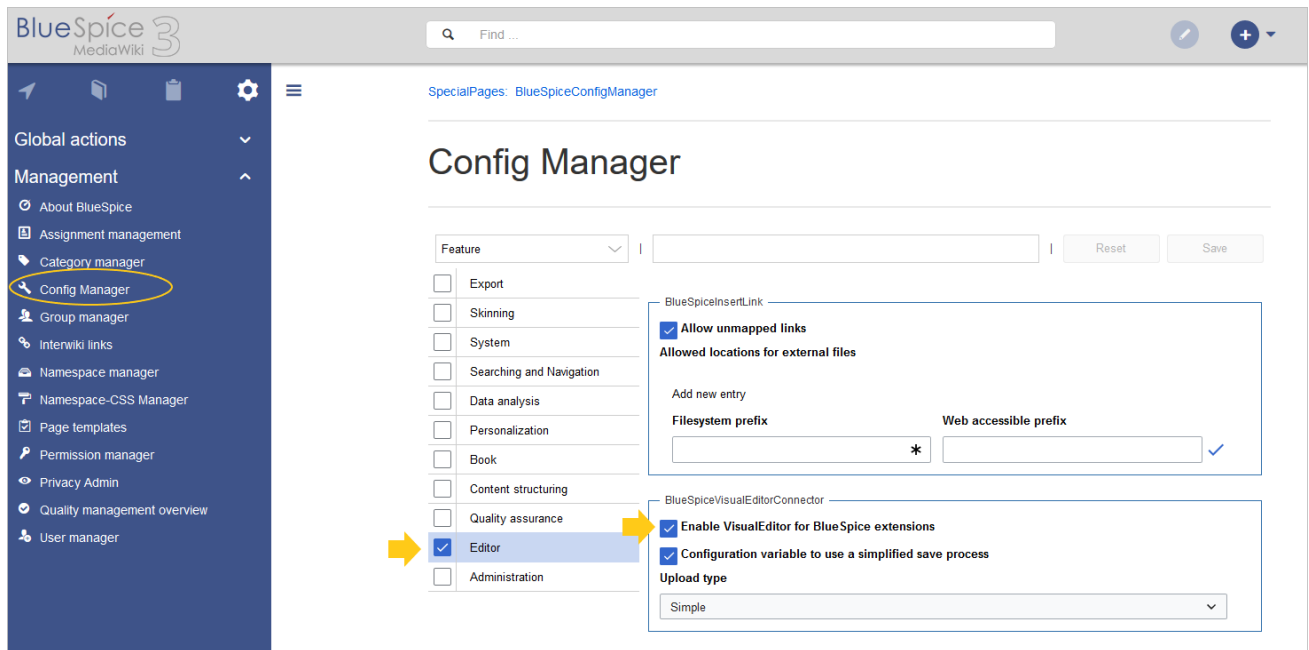
By default, VisualEditor is only activated in the main (Pages) and the User namespaces.

Activating VisualEditor for a namespace

1. Go to Global actions > Namespace manager In the column "VisualEditor", you can see which namespaces have VisualEditor activated. They have a green check mark.
2. Click the wrench icon in the "Actions" column. The wrench appears, when you hover over the row of the namespace:
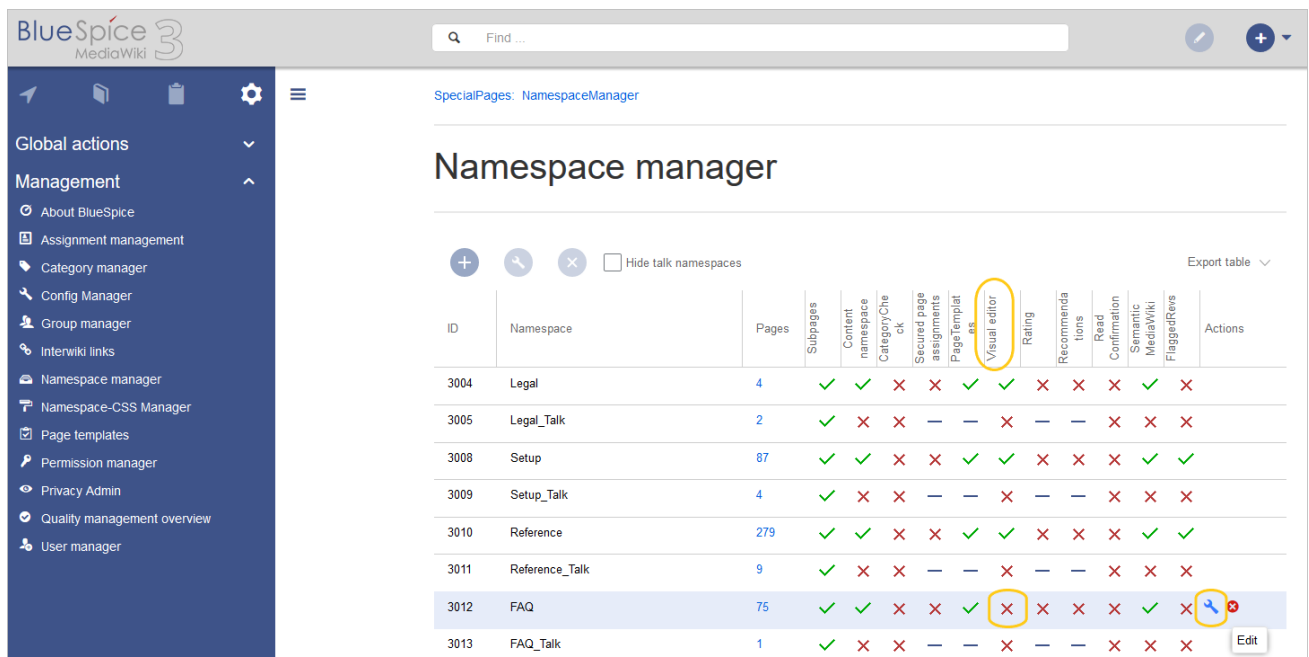
1. Activate the checkbox for VisualEditor in the dialog window and click "Done":
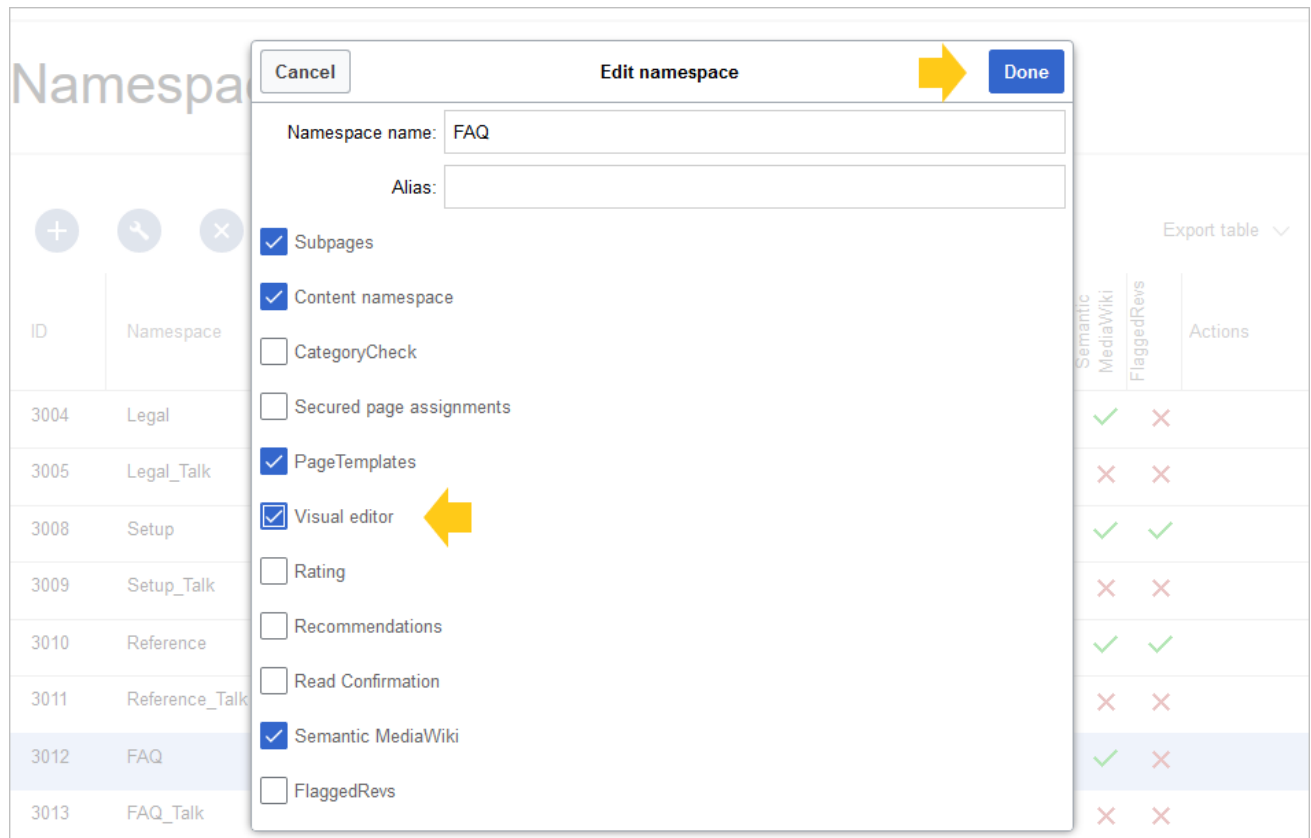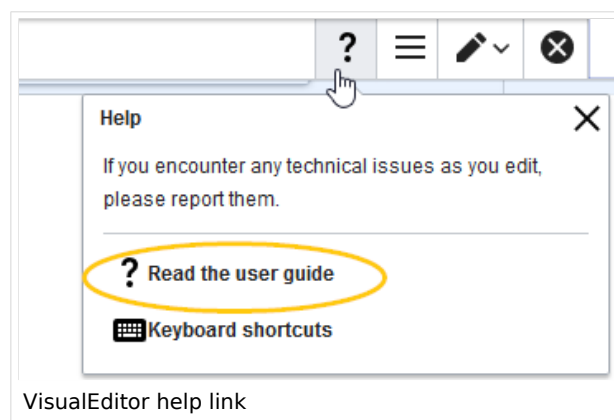
1. Refresh the Namespace manager page (F5) to confirm the change.

VisualEditor should now be working correctly in all namespaces that actively use it.

Customizing the link to the help pages

By default, the help link for VisualEditor links to the help page for VisualEditor on the BlueSpice helpdesk.

VisualEditor help link

The link can be changed to go to a custom URL:

To configure the url, add:

```
`$GLOBALS['bsgVisualEditorConnectorHelpUrl'] = 'https://your_url';`
```

to 'LocalSettings.php' or the appropriate config file.

Setup:Installation Guide/Advanced/Windows Folder Structure

A quality version of this page, approved on *2 August 2021*, was based off this revision.

Contents

```
`$GLOBALS['bsgVisualEditorConnectorHelpUrl'] = 'https://your_url';`
```

Preface

Microsoft Windows distributes program installations and configurations, the web root under IIS, etc., to various folders in the file system. Since BlueSpice is a PHP-based web application, it is not always easy to use the Windows default system paths in the application.
For this reason, we recommend creating your own folder structure for BlueSpice, for all related data and for additional program installations.
In the following we describe and recommend a proven folder structure.

Folder structure

As root directory for the installation, "\bluespice\" is specified in the root directory of the drive (for example, C:\bluespice or D:\bluespice).

The folder structure based on Unix operating systems is:

```
bluespice/

├── backup/            # for backups

├── bin/               # for additional programs and custom scripts

├── etc/               # for configuration files

├── opt/
│   └── bluespice/
│       ├── cache/         # BlueSpice-specific cache
│       ├── images/        # file uploads
│       └── widgets/       # rendered widgets

├── tmp/               # temporary files

└── var/
    ├── log/           # log files of the web server
    └── www/           # The actual BlueSpice application
```

File names for installing programs

When installing programs in \bluespice\ bin, please be sure to install these programs in subfolders that contain no special characters and/or spaces.