Contents

1. Setup:Installation Guide 2
2. Setup:Installation Guide/Advanced/Backup 4
3. Setup:Installation Guide/Advanced/Configuration Folder settings.d 5
4. Setup:Installation Guide/Advanced/ExtendedSearch Configuration 6
5. Setup:Installation Guide/Advanced/Folder structure under Windows 7
6. Setup:Installation Guide/Advanced/Maintenance scripts 8
7. Setup:Installation Guide/Advanced/Maintenance scripts/batchStabilize 9
8. Setup:Installation Guide/Advanced/Maintenance scripts/dumpPDF 11
9. Setup:Installation Guide/Advanced/Maintenance scripts/executeTransfer 11
10. Setup:Installation Guide/Advanced/Maintenance scripts/export 14
11. Setup:Installation Guide/Advanced/Performance Optimization 17
12. Setup:Installation Guide/Advanced/Performance Optimization/ManualRecache 17
13. Setup:Installation Guide/Advanced/Performance Optimization/MySQL 18
14. Setup:Installation Guide/Advanced/Performance Optimization/PHP 19
15. Setup:Installation Guide/Advanced/VisualEditor Configuration 20
16. Setup:Installation Guide/Advanced/Windows Folder Structure 22
17. Setup:Installation Guide/Docker/Docker Hub 23
18. Setup:Installation Guide/Installation BlueSpice WikiFarm 25
19. Setup:Installation Guide/Migration from MediaWiki to BlueSpice 28
20. Setup:Installation Guide/Optimization/Caching 33
21. Setup:Installation Guide/Optimization/Cronjobs 36
22. Setup:Installation Guide/Optimization/Time Zone 38
23. Setup:Installation Guide/Patch Update 39
24. Setup:Installation Guide/Security Settings/File System Permissions 44
25. Setup:Installation Guide/Security Settings/Save Directories 48
26. Setup:Installation Guide/System Preparation/Linux 49
27. Setup:Installation Guide/System Preparation/Windows 49
28. Setup:Installation Guide/Upgrade 50
29. Setup:Installation Guide/Webservices/Drawio 54
30. Setup:Installation Guide/Webservices/Mathoid 54
31. Setup:Installation Guide/Webservices/PDF-Export 55
32. Setup:Installation Guide/With Installer 59
33. Setup:System requirements 65

Setup:Installation Guide

Note: For general questions regarding the installation, maintenance, and usage of BlueSpice free, go to our community forum.

Contents

1 Introduction 3
2 Check the system requirements 3
3 Prepare the server environment 3
4 Install the BlueSpice application 3
5 Upgrade and patch updates 3
6 Migration from MediaWiki to BlueSpice 3
7 Optimize the configuration 3
7.1 Webservices for Apache Tomcat 3
7.2 Additional settings and optimizations 4
7.3 Security settings 4
7.4 Compendium 4

Introduction

We are happy that you decided to install the latest version of BlueSpice 4.

Please follow these steps:

1. Check the system requirements
2. Prepare the server environment
3. Install the BlueSpice application
4. Optimize the configuration

Check the system requirements

- System requirements

Prepare the server environment

- Linux server environment
- Windows server environment

Install the BlueSpice application

Please select which installation type you need:

- Full BlueSpice installation
  - Extended Functions: VisualEditor, ExtendedSearch
- BlueSpice WikiFarm installation
- Docker image

Upgrade and patch updates

- Patch update from BlueSpice 4.2.x to a higher version 4.2.x+n
- Upgrade from BlueSpice 3.2.x to BlueSpice 4.2.x

Migration from MediaWiki to BlueSpice

- Migration from MediaWiki to BlueSpice

Optimize the configuration

If you don't need to set up a server environment "from scratch", you can directly refer to the setup instructions for individual system components. Just make sure that you really have everything configured as needed:

Webservices for Apache Tomcat

- Drawio
- Mathoid
- PDF-Export

Additional settings and optimizations

- Caching
- Cronjobs
- Time Zone

Security settings

- File System Permissions
- Save Directories

Compendium

- Backup
- Configuration Folder settings.d
- ExtendedSearch Configuration
- Folder structure under Windows
- Maintenance scripts
- Maintenance scripts/batchStabilize
- Maintenance scripts/dumpPDF
- Maintenance scripts/executeTransfer
- Maintenance scripts/export
- Performance Optimization
- Performance Optimization/ManualRecache
- Performance Optimization/MySQL
- Performance Optimization/PHP
- VisualEditor Configuration
- Windows Folder Structure

Setup:Installation Guide/Advanced/Backup

Contents

1 Manual backup 5
1.1 Backup of the database 5
1.2 Backup of the file system 5
2 Using the mediawiki-adm tool 5

Manual backup

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 <database> > database.sql
```

For `<username>` , use the administrative database user, and for `<database>` , use the database of your MediaWiki/BlueSpice installation.

> On 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.

Using the `mediawiki-adm` tool

To eas backup and restore one can also use the `mediawiki-adm` tool. It can be downloaded here: https://github.com/hallowelt/misc-mediawiki-adm

Example usage:

```
mediawiki-adm wiki-backup \
    --mediawiki-root <installpath-bluespice> \
    --dest /mnt/backup/
```

The tool will leave out all unnecessary data (like temporary files and cache database tables)and produce much smaller backup files that the manual method described above. It also allows to create versioned backups.

Setup:Installation Guide/Advanced/Configuration Folder settings.d

> **Hinweis:** 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.

- 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).

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

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.

Folder structure under Windows

Contents

1 Preface 8
2 Folder structure 8
3 Folder names for program installations 8

Preface

Microsoft Windows distributes program installations and configurations, the Webroot under IIS etc. to various folders in the file system. Since BlueSpice is a web application based on PHP, 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, all related data and additional program installations.

In the following we describe a proven folder structure. This is recommended by us.

Folder structure

In the following, "\bluespice\" is defined as the base directory for the installation in the root directory of the drive (e.g. C:\bluespice or D:\bluespice).

The folder structure is based on Unix operating systems:

```
bluespice/
├─ backup/           # for backups
├─ bin/              # for additional programs and own scripts
├─ etc/              # for configuration files
├─ opt/
│  └─ bluespice/
│     ├─ cache/         # BlueSpice cache
│     ├─ images/        # File uploads
│     └─ widgets/       # Rendered widgets
├─ tmp/              # Temporary files
└─ var/
   ├─ log/           # LOG files of the web server
   └─ www/           # Actual BlueSpice application
```

Folder names for program installations

When installing programs in \bluespice\, please make sure to install these programs in subfolders that do not contain special characters and/or spaces.

Setup:Installation Guide/Advanced/Maintenance scripts

- createAndPromote.php:
  creates a new user or modifies an existing user. For the corresponding user, the script can then grant additional rights (e.g. sysop, bureaucrat or bot rights) and change the user's password.

- dumpBackup.php:
  creates an XML dump which contains the content of the wiki (wiki pages with all their revisions), without the site-related data.

- dumpPDF.php:
  exports pages from all content namespaces to PDF.

- importDump.php:
  imports XML dump files produced from*Special:Export* or *dumpBackup.php* and saves them into the current wiki.

- importImages.php:
  uploads images to MediaWiki from the same computer where the wiki is configured. It does a remote upload if configuration parameter$wgForeignFileRepos is configured correctly.

- executeTransfer.php:
  transfers files from one wiki to another. Dependent on the extensionContentTransfer

- export.php:
  exports a defined list of pages to PDF.

- contentStabilization v4.3+ :
  changes the page status of pages.

Setup:Installation Guide/Advanced/Maintenance scripts /batchStabilize

Contents

1 Mass approval using a script 10
1.1 Options 10
1.2 Examples 10
1.2.1 Whole namespace 10
1.2.2 List of page names 10

Mass approval using a script

The script `ContentStabilization/maintenance/batchStabilize.php` can be used to approve all drafts in the wiki starting with BlueSpice 4.3+. This affects both first drafts, drafts of pages with already approved versions and drafts for embedded resources.

This script allows the approval of all pages in a namespace or, alternatively, a list of pages defined in a text file.

Options

Option	Description
`--namespace`	Id of namespace to flag entirely
`--user`	Required. The user name of an existing user that is shown as page approver
`--comment`	Adds a stabilization comment to the revision history of the page
`--verbose`	Runs the script in verbose mode to show exceptions

Examples

Whole namespace

```
php extensions/ContentStabilization/maintenance/batchStabilize.php --user WikiSysop --
namespace 3000
```

List of page names

Starting with BlueSpice 4.3.2, a file with a newline-separate list of pages (as an arg or as stdin) can be used to define the pages that need to be approved:

```
php extensions/ContentStabilization/maintenance/batchStabilize.php --verbose /tmp/page-
titles-to-review.txt
```

```
php extensions/ContentStabilization/maintenance/batchStabilize.php --verbose < /tmp
/page-titles-to-review.txt
```



In case the page titles contain special characters (like ä, ö, ü and so on), make sure to save the file in UTF-8 encoding. Otherwise the script may not be able to find the according page names inside the database!

Setup:Installation Guide/Advanced/Maintenance scripts /dumpPDF

all maintenance scripts

Extension

BlueSpice Extension: BlueSpiceUEModuleBookPDF

Usage

This script allows to export the wiki pages **from all content namespaces** to PDF.

To run the script, execute:

```
php {MW_ROOT}/extensions/BlueSpiceUEModuleBookPDF/maintenance/dumpPDF.php --file:
<localpath> --limit:200 --mail:admin@mymail.com
```

Options

- `--file:` Defines the output path and the base filename.
  This name will be appended with the namespace number and, if `--limit` is set, with the split group.
  Expamle: `--file=/tmp/test.pdf` will result in `/tmp/test_0.pdf` or `/tmp/test_0_ABC.pdf`,
- `--limit:` Sets a limit for wikipages in a namespace included in one pdf. If the limit is exceed, the namspace will be split into pdfs depending on the first character of the page name. If parameter `--limit` is not set, all pages of a namespace will be exported to one pdf file.
- `--verbose`: Runs the script in verbose mode.
- `--mail:` E-mail address for a notification email.

You can see the list of possible options with descriptions by using the following command:

```
php {MW_ROOT}/extensions/BlueSpiceUEModuleBookPDF/maintenance/dumpPDF.php --help
```

Setup:Installation Guide/Advanced/Maintenance scripts /executeTransfer

all maintenance scripts

Contents

1 Reference 13
2 Usage 13
3 Configuration 13
4 Options 14

Reference

BlueSpice Extension: ContentTransfer

Script location: `extensions/ContentTransfer/maintenance/executeTransfer.php`

Usage

There are three ways to specify pages which pages to transfer:

- **`--category`**: This option lets you transfer based on category:

```
php extensions/ContentTransfer/maintenance/executeTransfer.php --category=GlobalPage
 --targets=Target1,Target2,Target3
```

- **`--namespace`**: This option lets you transfer based on namespace ID:

```
php extensions/ContentTransfer/maintenance/executeTransfer.php --namespace=10 --targ
ets=Target1,Target2,Target3
```

- **`--pages`** This option lets you explicitly specify which pages to transfer:

```
php extensions/ContentTransfer/maintenance/executeTransfer.php --pages=Page1,Demo:Pa
ge2,Page3 --targets=Target1,Target2,Target3
```

Configuration

The maintenance script can consume a JSON file with the necessary configuration. A sample JSON file is located at this path: `extensions/ContentTransfer/docs/transfer-config.json.example`. This file can be used to create a JSON configuration file somewhere, so that it is not necessary to type all options in the CLI each time.

For that, `--json-config` can be used. There can be a specified path to the file with the JSON configuration.

If you need to create your own JSON configuration file, copy the sample JSON file somewhere and edit it to set the necessary configuration.

For example, you have the file `/home/some_user/config/transfer-config.json`. Then it can be used like that:

```
php extensions/ContentTransfer/maintenance/executeTransfer.php --json-config=/home
/some_user/config/transfer-config.json
```

You can specify either an absolute file path from the root or a relative path from the present working directory.

Note!

It is also possible to use a JSON file and additionally pass options from the CLI. In that case, the options from CLI will have greater priority and **will override** the same options from JSON file.

Example:

```
php extensions/ContentTransfer/maintenance/executeTransfer.php --json-config=/home
/some_user/config/transfer-config.json --pages=Page1,Demo:Page2,Page3 --
targets=Target1,Target2,Target3
```

Here, the script will read the configuration from the specified JSON file , but the **“pages”** and **“targets”** options **will be overridden** from the CLI (if they are presented in JSON). The same is true for the other options.

Options

- `--user`: The context user of the "sending" wiki who will be used to transfer wiki pages. This is just about recording the transfer in the DB. Default: **“MediaWiki default”**
- `--only-modified`: If the page should be transferred only if it was modified since the last transfer. **It's just a flag, it can be passed or not.**
- `--modified-since`:Transfer the page only if it was modified since the specified date. The date must be specified in the format **"DD.MM.YYYY"**. Default: **“”**
- `--include-related`:If all related wiki pages should also be transferred (templates, files, links, and so on, which are used on the page to be transferred. **It's just a flag, it can be passed or not.**
- `--force`:Transfer the page even if it is protected on the receiving wiki. **It's just a flag, it can be passed or not.**

Also there is a `--dry` option (it's a flag too), which may be useful if a user just wants to take a look at transferring titles, but without actual changes in “receiving” wikis.

You can see a complete list of possible options with descriptions by the following command:

```
php extensions/ContentTransfer/maintenance/executeTransfer.php --help
```

If there is a need to set a specific user for creating pages in the target wiki , it can be done that way:

```
php extensions/ContentTransfer/maintenance/executeTransfer.php ... --
targets=Target1=User1,Target2=User2,Target3...
```

Setup:Installation Guide/Advanced/Maintenance scripts/export

Contents

1 Running the script 16
2 Creating a page list for the export 16
3 Setting up a CronJob 17

Running the script

The script to export wiki pages to PDF is executed from:

```
php {MW_ROOT}/extensions/BlueSpiceUniversalExport/maintenance/export.php --
specification-file=spec.json
```

The specification file `spec.json` can have the following content:

```
{
    "module": "pdf",
    "attachments": 1,
    "recursive": 1,
    "title": "MyPDF",
    "target": "localfilesystem",
    "target-file-name": "All.pdf",
    "target-file-path": "/path/to/target-file"
}
```

Key	Value	Description
module	pdf	Output type
attachments	1 0	Attachments will be included in the export
recursive	1 0	Linked pages (one level deep) will be included.
title	string	Title of the wiki page that contains the list of pages to be exported. **The pages need to be links.**
target	localfilesystem	The file will be exported to a local file system.
target-file-name		Name of the exported PDF file
target-file-path		Target file location

Creating a page list for the export

The export list is defined in the wiki page that is shown as the `title` in the json-file. (In the example above, this is the page `MyPDF` .)

The page `MyPDF` now needs to list all pages that will be exported to PDF.

This list can be created with an SMW inline query (BlueSpice free) or with a DPL3 list (BlueSpice free). Example of an SMW query to create an export for all pages in the categories `Process` and `Work instruction`:

You would add the following to your `MyPDF` page in the wiki:

```
==All processes==
{{#ask:[[Process:+]]|format=ul|limit=99999}}

==All work instructions==
{{#ask:[[Work instruction:+]]|format=ul|limit=99999}}
```

The same list can be created with the following DPL3 syntax:

```
==All processes==
{{#dpl:category=Process}}

==All work instructions==
{{#dpl:category=Work instruction}}
```

Setting up a CronJob

To automate the export, the following CronJob is available:

```
php {MW_ROOT}/extensions/BlueSpiceUniversalExport/maintenance/export.php --
specification-file={MW_ROOT}/extensions/BlueSpiceFoundation/data/spec.json
```

Setup:Installation Guide/Advanced/Performance Optimization

- Installation Guide/Advanced/Performance Optimization/ManualRecache
- Installation Guide/Advanced/Performance Optimization/MySQL
- Installation Guide/Advanced/Performance Optimization/PHP

Performance Optimization: ManualRecache of LanguageCache

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

Important! 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.

Note: 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
$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)

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

Contents

1 opcache 20
2 zlib 20

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
```

Up to BlueSpice 4.2.x only -v4.2.x : 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.

This config-file is no longer available from version 4.3.

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.

VisualEditor configuration

Contents

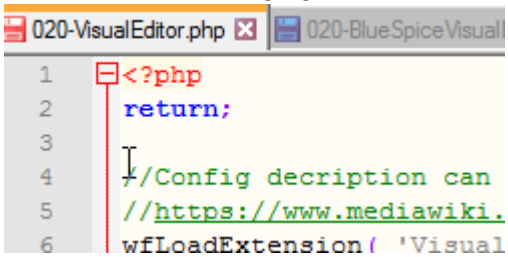
1 Server settings 21
2 Checking the Configuration manager 21
3 Activating VisualEditor for a namespace 21
4 Customizing the link to the help pages 22

Server settings

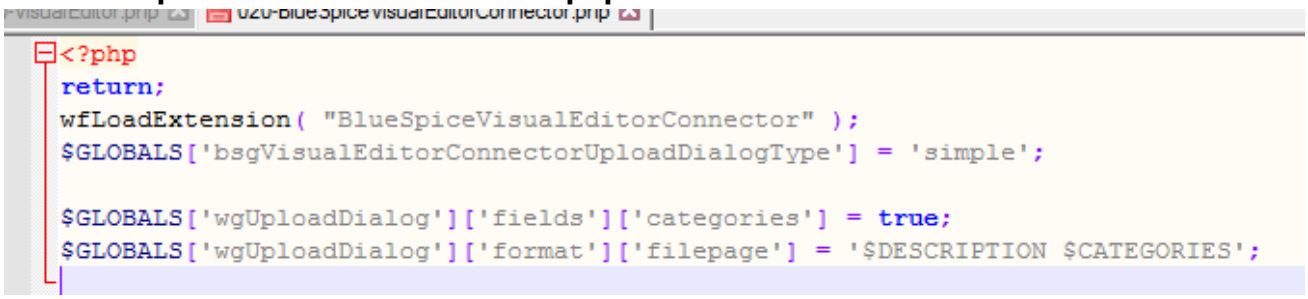
Before you can use VisualEditor in BlueSpice, you have to activate it in your installation:

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:

File:ve-config.png

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:

File:ve-config-namespace.png

1. Activate the checkbox for VisualEditor in the dialog window and click "Done":

File:ve-config-namespace-edit.png

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.

File:Setup:installation-ve-helplink.png
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.

Windows Folder Structure

Contents

1 Introduction 23
2 Folder structure 23
3 Folder names for program installations 23

Introduction

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

Folder structure

In the following, "\bluespice\" is specified as the base directory for the installation in the root directory of the drive (e.g. C:\bluespice or D:\bluespice).

The folder structure based on Unix operating systems is:

```
bluespice/
│
├─ backup/           # für Backups
│
├─ bin/              # für Zusätzliche Programme sowie eigene Scripte
│
├─ etc/              # für Konfigurationsdateien
│
├─ opt/
│  └─ bluespice/
│     ├─ cache/         # BlueSpice-eigener Cache
│     ├─ images/        # Datei-Uploads
│     └─ widgets/       # Gerenderte Widgets
│
├─ tmp/              # Temporäre Dateien
│
└─ var/
   ├─ log/           # LOG-Dateien des Webservers
   └─ www/           # Die eigentliche BlueSpice-Applikation
```

Folder names for program installations

When installing programs in \bluespice\bin, please always ensure that these programs are installed in subfolders that do not contain any special characters and/or spaces.

Installing BlueSpice free with Docker

The easiest way to run BlueSpice free is to install it with an all-in-one Docker image. All required services are preconfigured.

You can find all necessary information directly on Docker Hub.

How to use the Docker Hub image

Basic usage

Example for quick start. Blue Spice will be accessible only in localhost.

```
docker run -d -p 80:80 bluespice/bluespice-free
```

Keep your data outside of the docker

```
docker run -d -p 80:80 -v {/my/data/folder}:/data bluespice/bluespice-free
```

Setting BlueSpice language and URL

```
docker run -d -p 80:80 -v {/my/data/folder}:/data -e "bs_lang=en" -e "bs_url=http://www
.domain.com" bluespice/bluespice-free
```

Activating SSL

Using SSL inside the Blue Spice docker image, the `data` directory should be outside of the docker. Create a folder named `cert` inside your data folder. Inside this folder, certificates must be named like:

- `ssl.cert` (SSL certificate. *mandatory*)
- `ssl.key` (Private key of `ssl.cert`. *mandatory*)
- `ssl.ca` (3rd party CA certs for `ssl.cert`. *optional*) If everything is ready for the first run, just run the following command:

```
docker run -d -p 80:80 -p 443:443 -v {/my/data/folder}:/data -e "bs_lang=en" -e "bs_url
=https://www.domain.com" bluespice/bluespice-free
```

Note: Port 443 includes the command and also `$bs_url` schema changed to `https`

Login to BlueSpice

```
username: WikiSysop
password: PleaseChangeMe
```

Which services are runnning?

- Apache
- PHP-FPM
- Jetty9
- Elasticsearch

- MySQL/MariaDB
- Parsoid
- crond
- memcached

Setup:Installation Guide/Installation BlueSpice WikiFarm

Contents

1 PHP prerequisites 26
2 Ensuring the correct path in the file system 26
3 Creating the necessary folders for BlueSpiceWikiFarm 26
4 Creating the LocalSettingsAppend.php 26
5 Including the extension "BlueSpiceWikiFarm" 26
6 Configuring the web server 27
7 Adding new permissions in MySql 27
8 Important for Windows farming 27
9 cronjobs 27
10 Completion 27

An existing BlueSpice pro installation can easily be extended with BlueSpiceWikiFarm. The existing installation assumes the role of the main wiki (=farm management). The contents remain there and can be maintained as usually.

> **Note:** WikiFarm is not a standard feature of BlueSpice pro and needs to be acquired separately.

PHP prerequisites

Please ensure that the PHP extension ZIP is archived, which is required for BlueSpiceWikFarm.

Ensuring the correct path in the file system

Important for the operation of BlueSpiceWikiFarm is that BlueSpice is located in the directory `/w` of the DocumentRoot of your web server or VirtualHost.

If this is not the case, you have to move the entire codebase there.

After that, the variable `$wgScriptPath` in the `LocalSettings.php` contains the path `/w`.

```
$ wgScriptPath = "/w";
```

Creating the necessary folders for BlueSpiceWikiFarm

BlueSpiceWikiFarm needs two additional folders in the folder `/w` at the same level as the `LocalSettings.php`. You have to create these two folders:

- `_sf_instances` - all data and configurations for the respective instances are stored here
- `_sf_archive` - deleted farm instances including your database dump are stored here as ZIP archive

Make sure that these folders are writable by the web server. See also our (notes in the helpdesk) for this purpose.

Creating the LocalSettingsAppend.php

Including the extension "BlueSpiceWikiFarm" first requires an additional configuration file with the name `LocalSettingsAppend.php`. Create this file on the same level as the `LocalSettings.php` and add the following content:

```
<?php
require_once "$IP/LocalSettings.BlueSpice.php";
```

Including the extension "BlueSpiceWikiFarm"

In the `LocalSettings.php` replace the line

```
require_once "$IP/LocalSettings.BlueSpice.php";
```

with

```
require_once "$IP/extensions/BlueSpiceWikiFarm/BlueSpiceWikiFarm.php";
```

Configuring the web server

In the folder `extensions/BlueSpiceWikiFarm/SimpleFarmer/doc` you will find example files for RewriteRules, which are necessary for BlueSpiceWikiFarm:

- `htaccess.template` - RewriteRules for Apache
- `web.config.template` - RewriteRules for IIS

Configure these RewriteRules for the DocumentRoot level of your Web server or VirtualHost.



If you experience issues with spaces in page names: Add `RewriteRule ^([^\ ]*)\ (.*)$ $1_$2 [E=rspace:yes,N]` if it is not already included in your rewrite rules.

Adding new permissions in MySql

```
GRANT ALL ON ´sfr\_%´.* to ´bluespice´@´127.0.0.1´ identified by ´password´;
```

Important for Windows farming

https://www.mediawiki.org/wiki/Manual:$wgPhpCli

cronjobs

Has to be executed as Apache user:

```
*/5 * * * * php /var/www/bluespice/w/extensions/BlueSpiceWikiFarm/SimpleFarmer
/maintenance/RunForAll.php --script=maintenance/runJobs.php
*/1 * * * * php /var/www/bluespice/w/extensions/BlueSpiceWikiFarm/SimpleFarmer
/maintenance/FarmProcessRunner.php
```

Completion

The installation of BlueSpiceWikiFarm is now complete. From now on, you can reach the farm administration in the main wiki under the special page "Special:SimpleFarmer".

Important! If localhost is not available, download the IIS extension "URL Rewrite". https://www.iis.net/downloads/microsoft/url-rewrite

Setup:Installation Guide/Migration from MediaWiki to BlueSpice

Contents

1 Before you start 29
2 Backup 29
3 Preparing the migration 29
4 Importing the data 30
5 Configuration settings 30
6 Migrating the system 33

Before you start

It is important to check the compatibility of your current MediaWiki installation and the BlueSpice version to which you want to migrate:

- BlueSpice Version 3.x is based on MediaWiki 1.31.x
- BlueSpice version 4.x is based on MediaWiki 1.35.x

Backup

Create a full backup of your current installation. Commands may differ depending on your operating system.

1. Create a directory to store the backup:

```
mkdir -p /opt/mediawiki-migration-backup/{db,web}
```

2. Create the database dump:

```
mysqldump -u DB_USER_OR_ROOT -p MEDIAWIKI_DATABASE_NAME > /opt/mediawiki-migration-backup/db/database.sql
```

3. Copy everything in the web root to the backup folder:

```
cp -Rvf /PATH/TO/MEDIAWIKI/FOLDER/* /opt/mediawiki-migration-backup/web/
```

We will use this backup later for the migration process as well.

Preparing the migration

You can create a separate user account and database for BlueSpice. This step is optional but recommended. The following steps are based on such a separately created user and database.

If you choose not to create a separate user account and database, replace the following commands accordingly.

1. Create a MySQL/MariaDB user via MySQL CLI:

```
CREATE USER 'bluespice'@'localhost' IDENFITIED BY 'PleaseChooseAComplexPassword';
```

2. Create a database for BlueSpice via MySQL CLI:

```
CREATE DATABASE bluespice;
```

3. Grant the required privileges for the bluespice user to the bluespice database via MySQL CLI:

```
GRANT ALL PRIVILEGES ON bluespice.* TO 'bluespice'@'localhost';
FLUSH PRIVILEGES;
```

Now we have an empty SQL database and are ready to import.

Importing the data

1. Import the SQL dump from the backup you created earlier:

```
mysql -u bluespice -p bluespice < /opt/mediawiki-migration-backup/db/database.sql
```

After this command, enter the password you chose for the `bluespice` database user.
2. Next, clean up the old installation folder and create an empty one.

```
rm -Rf /PATH/TO/MEDIAWIKI/FOLDER/
```

and

```
mkdir -p /PATH/TO/MEDIAWIKI/FOLDER/
```

3. Next, extract the BlueSpice package and place it into the /PATH/TO/MEDIAWIKI/FOLDER/

```
cd /PATH/TO/MEDIAWIKI/FOLDER/
```

4. Copy the *images* folder from the backup to the /PATH/TO/MEDIAWIKI/FOLDER/ :

```
cp -Rf /opt/mediawiki-migration-backup/images .
```

5. Set the correct permissions based on your operating system to the folder /PATH/TO/MEDIAWIKI/FOLDER/ .

Configuration settings

1. In /PATH/TO/MEDIAWIKI/FOLDER/, create a new file named *LocalSettings.php* and copy the following content into the file:

```
<?php
# This file was automatically generated by the MediaWiki 1.35.3
# installer. If you make manual changes, please keep track in case you
# need to recreate them later.
#
# See includes/DefaultSettings.php for all configurable settings
# and their default values, but don't forget to make changes in _this_
# file, not there.
#
# Further documentation for configuration settings may be found at:
# https://www.mediawiki.org/wiki/Manual:Configuration_settings
```

```
# Protect against web entry
if ( !defined( 'MEDIAWIKI' ) ) {
        exit;
}


## Uncomment this to disable output compression
# $wgDisableOutputCompression = true;

$wgSitename = "YOUR_WIKI_NAME";

## The URL base path to the directory containing the wiki;
## defaults for all runtime URL paths are based off of this.
## For more information on customizing the URLs
## (like /w/index.php/Page_title to /wiki/Page_title) please see:
## https://www.mediawiki.org/wiki/Manual:Short_URL
$wgScriptPath = "/w"; # <--- PLEASE CHECK YOUR DOCUMENTROOT

## The protocol and server name to use in fully-qualified URLs
$wgServer = "http(s)://your-domain.ltd";

## The URL path to static resources (images, scripts, etc.)
$wgResourceBasePath = $wgScriptPath;

## The URL paths to the logo.  Make sure you change this from the default,
## or else you'll overwrite your logo when you upgrade!
$wgLogos = [ '1x' => "$wgResourceBasePath/resources/assets/wiki.png" ];

## UPO means: this is also a user preference option

$wgEnableEmail = true;
$wgEnableUserEmail = true; # UPO

$wgEmergencyContact = "nomail@localhost.localdomain";
$wgPasswordSender = "nomail@localhost.localdomain";

$wgEnotifUserTalk = false; # UPO
$wgEnotifWatchlist = false; # UPO
$wgEmailAuthentication = true;

## Database settings
$wgDBtype = "mysql";
$wgDBserver = "YOUR_DATABASE_SERVER";
$wgDBname = "bluespice";
$wgDBuser = "bluespice";
$wgDBpassword = "PleaseChooseAComplexPassword";

# MySQL specific settings
$wgDBprefix = "";

# MySQL table options to use during installation or update
$wgDBTableOptions = "ENGINE=InnoDB, DEFAULT CHARSET=binary";

# Shared database table
# This has no effect unless $wgSharedDB is also set.
$wgSharedTables[] = "actor";

## Shared memory settings
$wgMainCacheType = CACHE_NONE;
$wgMemCachedServers = [];

## To enable image uploads, make sure the 'images' directory
## is writable, then set this to true:
$wgEnableUploads = false;
$wgUseImageMagick = true;
$wgImageMagickConvertCommand = "/usr/bin/convert";

# InstantCommons allows wiki to use images from https://commons.wikimedia.org
$wgUseInstantCommons = false;

# Periodically send a pingback to https://www.mediawiki.org/ with basic data
```

```
# about this MediaWiki instance. The Wikimedia Foundation shares this data
# with MediaWiki developers to help guide future development efforts.
$wgPingback = false;

## If you use ImageMagick (or any other shell command) on a
## Linux server, this will need to be set to the name of an
## available UTF-8 locale. This should ideally be set to an English
## language locale so that the behaviour of C library functions will
## be consistent with typical installations. Use $wgLanguageCode to
## localise the wiki.
$wgShellLocale = "C.UTF-8";

## Set $wgCacheDirectory to a writable directory on the web server
## to make your wiki go slightly faster. The directory should not
## be publicly accessible from the web.
#$wgCacheDirectory = "$IP/cache";

# Site language code, should be one of the list in ./languages/data/Names.php
$wgLanguageCode = "en";

$wgSecretKey = "68a265061a4101d0d3dee2a06eeb734abaa6710a7dbe103838f2a26a50fc7835";

# Changing this will log out all existing sessions.
$wgAuthenticationTokenVersion = "1";

# Site upgrade key. Must be set to a string (default provided) to turn on the
# web installer while LocalSettings.php is in place
$wgUpgradeKey = "8775d57b99d672b8";

## For attaching licensing metadata to pages, and displaying an
## appropriate copyright notice / icon. GNU Free Documentation
## License and Creative Commons licenses are supported so far.
$wgRightsPage = ""; # Set to the title of a wiki page that describes your license
/copyright
$wgRightsUrl = "";
$wgRightsText = "";
$wgRightsIcon = "";

# Path to the GNU diff3 utility. Used for conflict resolution.
$wgDiff3 = "/usr/bin/diff3";

## Default skin: you can change the default skin. Use the internal symbolic
## names, ie 'vector', 'monobook':
$wgDefaultSkin = "bluespicediscovery";

# End of automatically generated settings.
# Add more configuration options below.


# This is the main settings file for all BlueSpice extensions and settings
# It will include all files in "$IP/settings.d/" directory
require_once "$IP/LocalSettings.BlueSpice.php";


$wgUserMergeProtectedGroups = array();
$wgUserMergeUnmergeable = array();
$wgMetaNamespace = 'Project';
$bsgGroupRoles['*']['reader'] = false;

# Convenience for debugging
# $wgShowSQLErrors = true;
# $wgDebugDumpSql  = true;
# $wgShowExceptionDetails = true;
# $wgShowDBErrorBacktrace = true;
```

2. Edit this *LocalSettings.php* and set the correct values to fit your installation. This current *LocalSettings.php* might look like different from your MediaWiki installation, because BlueSpice comes with a separate folder for custom settings (*settings.d*).

Migrating the system

The system is ready to migrate.

1. Run the following script to start the migration:

```
php /PATH/TO/MEDIAWIKI/FOLDER/maintenance/update.php --quick
```

   Now BlueSpice should be reachable.
2. Execute the following additional scripts:

```
php /PATH/TO/MEDIAWIKI/FOLDER/maintenance/rebuildall.php
#Finally for the search index (ElasticSearch 6.x and ingest-attachment plugins must
be installed) php /PATH/TO/MEDIAWIKI/FOLDER/extensions/BlueSpiceExtendedSearch
/maintenance/initBackends.php --quick
php /PATH/TO/MEDIAWIKI/FOLDER/extensions/BlueSpiceExtendedSearch/maintenance
/rebuildIndex.php --quick
php /PATH/TO/MEDIAWIKI/FOLDER/maintenance/runJobs.php --memory-limit=max
```

Your migration is complete and your BlueSpice is ready to use!

You can use your old credentials to login.

Questions? Visit the BlueSpice help forum on sourceforge.net.

Caching

Contents

1 Tips for the Document 34
2 Linux vs. Windows 34
3 Indicate Cache-Directory 34
4 Name Resolution of the Database 34
5 Activate Memcached 35
6 Increase PHP Memory Limit 35
7 Activate Bytecode Cache in PHP 35
8 Deactivate JobQueue 36

This document describes different caching options to enhance the performance of your BlueSpice installation.

Tips for the Document

- 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).

Linux vs. Windows

Please note that with equal hardware conditions, BlueSpice empirically runs more performantly on Linux than on Windows servers.

Indicate Cache-Directory

This forces BlueSpice to use the filesystem instead of the database for some internal caches. To do this, create the file `<installpath-bluespice>`/settings.d/001-Directories.php and add the following content:

```
<?php
$wgCacheDirectory = "$IP/cache";
```

Save and close the file.

> **Hinweis:** The directory "cache" in `<installpath-bluespice>` must be writable by the web server. For more information, read the file system permissions.

Name Resolution of the Database

If this hasn't already been done while setting up MediaWiki, please make sure that the database server will be adressed via IP if it runs on the same server than BlueSpice. Especially Windows has problems with the resolution of "localhost".

To do this, open `<installpath-bluespice>`/LocalSettings.php and locate the variable **$wgDBserver**. The line should ideally be:

```
$wgDBserver = "127.0.0.1";
```

Activate Memcached

Create the file `<installpath-bluespice>`/settings.d/001-Memcached.php and add the following content:

```
<?php

$wgMainCacheType = CACHE_MEMCACHED;
$wgSessionCacheType = CACHE_DB;
$wgMemCachedServers = [ "127.0.0.1:11211" ];
```

Save and close the file.

Hinweis: The prerequisite is an installed and configured memcached server on the BlueSpice server. Read the memcached article for more information.

Increase PHP Memory Limit

A higher memory limit of PHP results in a faster execution. ¶ This can be changed in php.ini. Locate and adjust the following option:

```
memory_limit = 512M (means 512 MB - varies from the available memory)
```

After saving and closing php.ini, the web server must be restarted.

Activate Bytecode Cache in PHP

Since PHP 5.5 the bytecode cache "opcache" is included.

This needs to be activated in the php.ini first. Check if the module is integrated there. This is done via the line:

```
zend_extension=opcache.dll (Windows)
zend_extension=opcache.so (Linux)
```

Hinweis: Please also note the specification of the distributor for activating PHP modules under Linux.

Please, do the following configuration of the opcache in php.ini. Normally, all options should already exist, but are commented out by semicolon (;) at the beginning of the line. Locate the respective option in php.ini, remove the semicolon at the beginning of the line and adjust the settings as follows:

```
opcache.enable=1
opcache.memory_consumption=512 (means 512 MB - varies from the available memory)
opcache.max_accelerated_files=5000
opcache.validate_timestamps=1
opcache.revalidate_freq=2
```

After saving and closing php.ini, the web server must be restarted.

Deactivate JobQueue

For this, read the paragraph "runJobs.php" in the "Cronjobs" article.

Cronjobs

Contents

1 Tips for this Document 37
2 runJobs.php 37
3 processRunner.php 37

This document describes cronjobs an related configurations, which should be set for your BlueSpice (depending on the used version).

Tips for this Document

- 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 regarding the documentation "Folder structure under Windows ") or `/var/www/bluespice` (Linux).
- The placeholder `<tomcat-webapps` stands for the path to the webapps directory of your Tomcat server, e.g `C:\Program Files\Apache Software Foundation\Tomcat 8.5\webapps` (Windows) or `/var/lib/tomcat8/webapps` (Linux).

> **Hinweis:** It is recommended to specify the complete path to php.exe under Windows.

runJobs.php

Time-consuming processes will be moved to a processing queue in the background of BlueSpice. With every page impression, a part of these will be processed. For the case that there are lots of processes in the queue and there are relatively litte page impressions, the tasks will not be processed promptly.

For this reason, the queue should be processed regularly by cronjob.

Create a cronjob (Windows: "Scheduled task") depending on your operating system. The command to execute is:

```
php <installpath-bluespice>/maintenance/runJobs.php (Linux)
php.exe <installpath-bluespice>\maintenance\runJobs.php (Windows)
```

Execute the cronjob every **10 minutes**.

processRunner.php

Certain tasks require server resources that can not be provided in the regular context of a webrequest. They will be moved to a special processing queue in the background of BlueSpice.

Create a cronjob (Windows: "Scheduled task") depending on your operating system. The command to execute is:

```
php <installpath-bluespice>/vendor/mwstake/mediawiki-component-processmanager
/maintenance/processRunner.php <installpath-bluespice>/maintenance/Maintenance.php --
max-processes=100 --wait (Linux)
php.exe <installpath-bluespice>\vendor\mwstake\mediawiki-component-
processmanager\maintenance\processRunner.php <installpath-bluespice>\maintenance\Maint
enance.php --max-processes=100 --wait (Windows)
```

Execute the cronjob every **minute**.

Time Zone

Contents

1 Tips for this Document 39
2 Changing the Time Zone 39

Even if every user can change his time zone himself in the user settings, BlueSpice sets the default time zone "Europe/Berlin" when the user first logs in. This document describes how to change this if necessary.

Tips for this Document

- 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 regarding the documentation "Folder structure under Windows ") or `/var/www/bluespice` (Linux).
- The placeholder `<tomcat-webapps` stands for the path to the webapps directory of your Tomcat server, e.g `C:\Program Files\Apache Software Foundation\Tomcat 8.5\webapps` (Windows) or `/var/lib/tomcat8/webapps` (Linux).

Changing the Time Zone

Copy the file `<installpath-bluespice>`/settings.d/001-DefaultSettings.php to `<installpath-bluespice>`/settings.d/001-DefaultSettings.local.php and open it. In delivery stat, you can find these two lines there:

```
$wgLocaltimezone = 'Europe/Berlin';
$wgDefaultUserOptions['timecorrection'] = 'ZoneInfo|' . (date("I") ? 120 : 60) .
'|Europe/Berlin';
```

Replace "Europe / Berlin" with your time zone. An overview of all possible time zones can be found in the official PHP documentation.
Save and close the file.

Setup:Installation Guide/Patch Update

This page contains instructions for a patch update (e.g., from version 4.1.x to a higher version 4.1.x+).

Contents

1 Backup 41
2 Renew the code base 41
3 Local settings and data 41
3.1 Compare local files and some manual made settings 43
3.2 Copy from your backup Local files and some manual made settings 43

4 Run the update 43
5 Check your current Version 43
6 If problems occur 44

Backup

First, you need to make a backup of your database:

```
mysqldump -u root -p --all-databases > all_databases.sql
```

To make your code base more secure, save it to a secure location of your choice:

```
#Linux bash
cp -r codebase your_secure_place
```

```
# Win bash
xcopy codebase your_secure_place\ /E
```

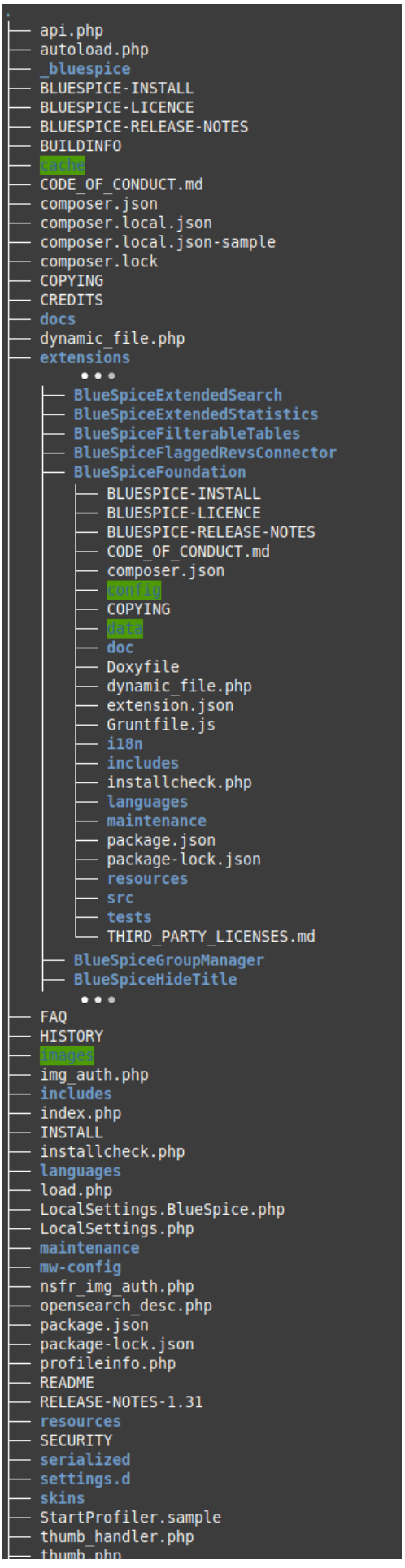
Renew the code base

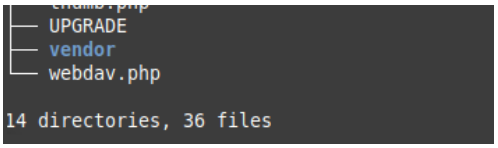
Unzip the archive (new Wiki version). Overwrite the old wiki code base with the new wiki version code:

```
#Linux bash
cp -r new_version_codebase old_version_codebase
rm -rf new_version_codebase
```

```
#Win bash
xcopy new_version_codebase old_version_codebase\ /E/H #path/ -  to tell xcopy that
path is a directory.
rmdir new_version_codebase /Q/S
```

Local settings and data

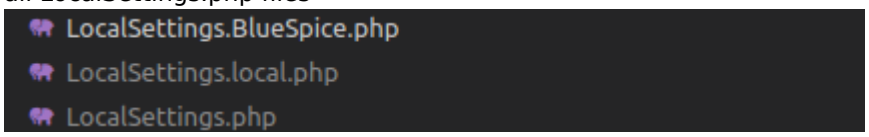

.
├── api.php
├── autoload.php
├── _bluespice
├── BLUESPICE-INSTALL
├── BLUESPICE-LICENCE
├── BLUESPICE-RELEASE-NOTES
├── BUILDINFO
├── cache
├── CODE_OF_CONDUCT.md
├── composer.json
├── composer.local.json
├── composer.local.json-sample
├── composer.lock
├── COPYING
├── CREDITS
├── docs
├── dynamic_file.php
├── extensions
│ • • •
│ ├── BlueSpiceExtendedSearch
│ ├── BlueSpiceExtendedStatistics
│ ├── BlueSpiceFilterableTables
│ ├── BlueSpiceFlaggedRevsConnector
│ ├── BlueSpiceFoundation
│ │ ├── BLUESPICE-INSTALL
│ │ ├── BLUESPICE-LICENCE
│ │ ├── BLUESPICE-RELEASE-NOTES
│ │ ├── CODE_OF_CONDUCT.md
│ │ ├── composer.json
│ │ ├── config
│ │ ├── COPYING
│ │ ├── data
│ │ ├── doc
│ │ ├── Doxyfile
│ │ ├── dynamic_file.php
│ │ ├── extension.json
│ │ ├── Gruntfile.js
│ │ ├── i18n
│ │ ├── includes
│ │ ├── installcheck.php
│ │ ├── languages
│ │ ├── maintenance
│ │ ├── package.json
│ │ ├── package-lock.json
│ │ ├── resources
│ │ ├── src
│ │ ├── tests
│ │ └── THIRD_PARTY_LICENSES.md
│ ├── BlueSpiceGroupManager
│ ├── BlueSpiceHideTitle
│ • • •
├── FAQ
├── HISTORY
├── images
├── img_auth.php
├── includes
├── index.php
├── INSTALL
├── installcheck.php
├── languages
├── load.php
├── LocalSettings.BlueSpice.php
├── LocalSettings.php
├── maintenance
├── mw-config
├── nsfr_img_auth.php
├── opensearch_desc.php
├── package.json
├── package-lock.json
├── profileinfo.php
├── README
├── RELEASE-NOTES-1.31
├── resources
├── SECURITY
├── serialized
├── settings.d
├── skins
├── StartProfiler.sample
├── thumb_handler.php
├── thumb.php



Compare local files and some manual made settings

The following files and folders contain local changes:

- settings files
  - settings.d/ folder **OR**
  - all files that contain 'local.php' in settings.d/ folder
- extension/BlueSpiceFoundation/config/ folder (only available until BlueSpice 4.2.x) -v4.2.x
- images/ folder
- all LocalSettings.php files



Copy from your backup Local files and some manual made settings

For a fast copy, use this bash command:

```
#Linux Bash
cd your_secure_place
cp -r images/ LocalSettings.* settings.d/ codebase/
cp -r /extensions/BlueSpiceFoundation/config /codebase/extensions/BlueSpiceFoundation/
```

```
#Win Bash
cd your_secure_place
for %I in (images/ LocalSettings.* settings.d/) do copy %I codebase/
for %I in (config/ data/) do copy %I codebase/extensions/BlueSpiceFoundation/
```

Run the update

After all preparations are done and code base is overwritten, do the following steps:

1. open a console and go to the installation folder (LocalSettings.php directory-level, see: right-image)
2. enter the following command:

```
php maintenance/update.php
```

Note: If you use Linux for your BlueSpice installation, please note that the file system permissions may be lost when overwriting the code base!

Check your current Version

Open the page `Special:Version` in your Wiki and check *BlueSpice(Credits)*:

Product	Version
MediaWiki	1.35.7 (7fe2bdb) 07:29, 31 August 2022
PHP	8.0.20 (fpm-fcgi)
MariaDB	10.8.3-MariaDB-1:10.8.3+maria~jammy
ICU	66.1
Lua	5.1.5
BlueSpice (Credits)	4.3.0-alpha



If problems occur

- Visit our SourceForge support forum or report a bug.
- For general questions regarding the installation, maintenance, and usage of BlueSpice free, go to our SourceForge help forum.

File System Permissions

Contents

1 Tips for this Document 45
2 Affected Folders 45
3 Assignment of Permissions with Windows 45
4 Assignment of Permissions with Linux 46

For trouble-free usage of your BlueSpice installation, the web server requires write permissions to several folders of the file system. However, it is recommended to keep the rights for all other files and folders to a minimum. This document shows you the relevant folders and the correct permissions setting.

Tips for this Document

- 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).

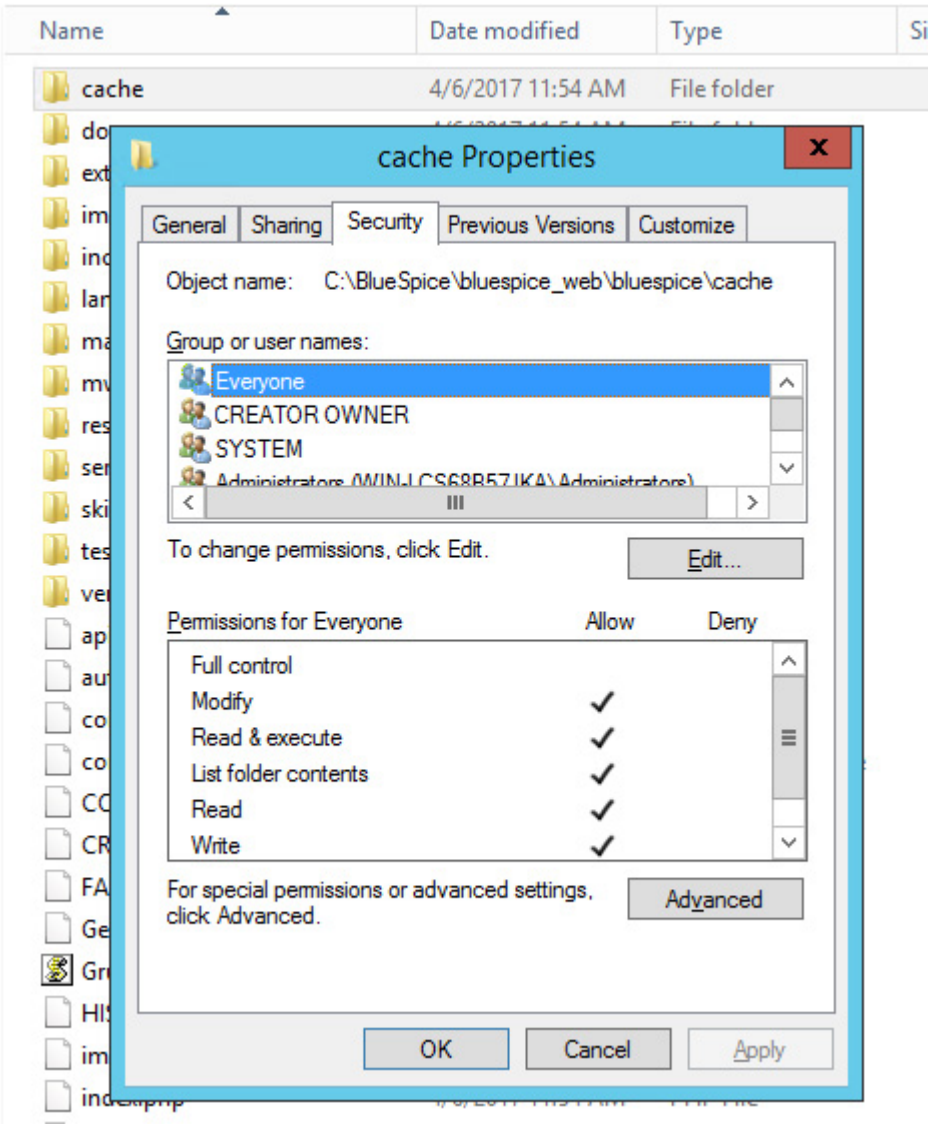
Affected Folders

The folders, which require write permissions, are:

- `<installpath-bluespice>`/cache
- `<installpath-bluespice>`/images
- `<installpath-bluespice>`/extensions/BlueSpiceFoundation/config (only until BlueSpice v. 4.2.x) - v4.2.x
- `<installpath-bluespice>`/extensions/BlueSpiceFoundation/data
- `<installpath-bluespice>`/extensions/Widgets/compiled_templates **(only BlueSpice pro)**

Assignment of Permissions with Windows

Assign "change" rights for this folder for the local user "Everyone".



Assignment of Permissions with Linux

Here you can assign rights much more strictly, which is also recommended. Transfer the directory `<installpath-bluespice>` recursively to the user root (CHMOD for Files 644, CHMOD for Directories 755) and after this, transfer the mentioned directory recursively to the user and the group, under which the Apache Web Server runs (Debian/Ubuntu e.g. "www-data").

In the following, we provide a bash script, which works through this tasks for you with only one command.

To do so, create the file /usr/local/bin/setWikiPerm and copy the following code into this file:

```
#!/bin/bash

WWW_USER="www-data"
WWW_GROUP="www-data"

WWW_HOME=`eval echo ~$WWW_USER`
WWW_CFG=$WWW_HOME/.config

if [ $# -eq 0 ]; then
        echo "You must enter the path of your MediaWiki installation."
```

```
        exit
elif [ ! -d $1 ]; then
        echo "$1 does not exist or is no path."
        exit
fi

if [ ! -f $1/LocalSettings.php ]; then
echo "$1 contains no Localsettings.php"
exit
fi


PATH=`echo "$1" | sed -e 's#/$##'`

/usr/bin/find $PATH -type d -exec /bin/chmod 755 {} \;
/usr/bin/find $PATH -type f -exec /bin/chmod 644 {} \;

/bin/chown -R root:root $PATH

pathes=(
        "$PATH/cache" \
        "$PATH/images" \
        "$PATH/_sf_archive" \
        "$PATH/_sf_instances" \
        "$PATH/extensions/BlueSpiceFoundation/data" \
        "$PATH/extensions/Widgets/compiled_templates" \
)

for i in "${pathes[@]}"; do
        if [ -d $i ]; then
                /bin/chown -R $WWW_USER:$WWW_GROUP $i
        fi
done

if [ ! -d $WWW_CFG ]; then
        /bin/mkdir $WWW_CFG
fi

/bin/chown -R $WWW_USER:$WWW_GROUP $WWW_CFG

/usr/bin/find $PATH/extensions -iname 'create_pygmentize_bundle' -exec /bin/chmod +x
{} \;
/usr/bin/find $PATH/extensions -iname 'pygmentize' -exec /bin/chmod +x {} \;
/usr/bin/find $PATH/extensions -name 'lua' -type f -exec /bin/chmod 755 {} \;
```

If needed, replace the content of the variables

```
WWW_USER="www-data"
WWW_GROUP="www-data"
```

with the approprate user and group of your distribution.

After this, assign CHMOD 755 to this file. Now you can run the script and let it do the complete permission setting automatically with the following command:

```
setWikiPerm <installpath-bluespice>
```

Note: Note that when the update.php is executed on the console, the rights can be partially discarded. For this reason, set the rights to "update.php" again.

Save Directories

Contents

1 Tips for this Document 49
2 Explanation for Saving Directories 49

Tips for this Document

- 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).

Explanation for Saving Directories

It is highly recommended, to lock several directories to save them from access from external browers, so that calling up the directory or the content delivers the HTTP status code (Forbidden).

To find out how this works with the webserver you are using, please read the documentation of your webserver software.

The following directories should be saved:

- `<bluespice-url>`/cache
- `<bluespice-url>`/images
  Please keep in mind that you need to unblock the subfolder `<bluespice-url>`/images/bluespice/flexiskin afterwards, because this subfolder must be accessable.
- `<bluespice-url>`/mw-config

Setup:Installation Guide/System Preparation/Linux

- Introduction
- Apache and PHP
- MariaDB
- Jetty
- OpenSearch (from BlueSpice 4.4) / Elasticsearch (up to BlueSpice 4.3)
- Python
- Memcached
- Mathoid **(BlueSpice pro only)**
- PhantomJS **(BlueSpice pro only)**

Setup:Installation Guide/System Preparation/Windows

This part of the installation guide provides step-by-step instructions for the Windows system preparation. This is necessary if you want to install BlueSpice afterwards.

Follow these chapters in the order listed. At the end of each chapter you will see a link to the next chapter.

Please also pay attention to the chapter "Introduction". It contains additional information about this manual.

- Introduction
- Setting the system rights of the temporary Windows folder
- .NET Framework 3.5
- IIS-Webserver
- urlrewrite
- Microsoft Visual C++ Redistributable
- PHP Manager
- PHP
- MariaDB
- OpenJDK
- Apache Tomcat
- OpenSearch (from BlueSpice 4.4) / Elasticsearch (to BlueSpice 4.3)
- Python

Upgrade from BlueSpice 3 to 4

Contents

1 System requirements 51
1.1 Browser 51
1.2 Server environment 51
2 Back up the old wiki 51
3 Upgrade step 1 52
3.1 Create a "substitute user" for deleted users 52
4 Sql dump 52
5 Optional files for migration 52
6 Migrate to the new wiki 53
6.1 Modify LocalSettings.php 53
7 Upgrade step 2 53
8 Optional 53
9 Last steps 54
10 Known possible issues after the update 54
10.1 Problems with VisualEditor 54

System requirements

Browser

- Microsoft Edge
- Google Chrome
- Firefox

Server environment

- **Operating system:**
  - We strongly recommend Linux (preferably Debian 11, Ubuntu 22.04, CentOS 7)
  - You might use Windows Server starting at 2016, but we have seen performance issues on Windows Server
- **Webserver:**
  - Apache 2.4.x, IIS >= 10 *or* nginx 1.x (*nginx not possible in WikiFarm*)
- **PHP:**
  - PHP 8.1 / PHP 8.2
- **Database**:
  - MySQL: >= 5.6 or
  - MariaDB >= 10.3
- **(Virtual) hardware requirements:**
  - **CPU:**
    - **Linux: 8 Cores** (min. 4 Cores)
    - Windows: 16 Cores (min. 8 Cores)
  - **Main memory:**
    - **Linux: 16 GB** (min. 8GB)
    - Windows: min. 16 GB
  - **Available hard drive space:**
    - > 20 GB (depends on the planned storage of data)
- **Other:**
  - Apache Tomcat >= 9 oder Jetty >= 9 (for PDF export and LaTexRenderer)
  - ElasticSearch 6.8 with plugin “ingest-attachment”
  - OpenJDK >= 10
  - NodeJS 16

> **Note:** You might need to add external repos to get the correct php on your server! Please check your package manager for the available php versions on your system.

Back up the old wiki

In BlueSpice 3, create a temporary migration folder and copy all necessary data:

```
mkdir /tmp/migration
cd /tmp/migration
wikifolder=/path/to/wikifolder
cp -r $wikifolder/images .
```

```
cp -r $wikifolder/extensions/BlueSpiceFoundation/config .
cp -r $wikifolder/LocalSettings.* .
#check for for locals in settings.d
find $wikifolder/settings.d/ -iname '*.local.php' -exec cp --parent {} /tmp/migration \
;
```

Also search for specialized php files in `settings.d` (normally `090-` or `099-`).

Upgrade step 1

This step has to be completed before the *Upgrade step 2* further below.

> **Warning!** BlueSpice 4.x cannot process wiki pages and files that were uploaded and edited by **deleted users**. To prevent data loss, don't skip any upgrade steps!

Create a "substitute user" for deleted users

In BlueSpice 3.x, create a user with a random password:

```
php maintenance/createAndPromote.php --force DeletedUser someR4ndomPass
```

> **Note:** You may need to temporarily disable authentication plugins like LDAP-Stack, SAML or OpenIDConnect extensions.

Get the `user_id` of the substitute user by running the following command in the database:

```
SELECT user_id FROM user WHERE user_name = "DeletedUser";
```

Sql dump

Create the backup of your data:

```
mysqldump -u root -p bluespice > migrationdump.sql
```

Optional files for migration

If any of the following features apply to your installation, check if migration steps are necessary. In that case, add everything to your *tar* file and send it to the new server.

Examples of files that you might have to consider:

- ssl-certs
- apache.conf
- ldapprovider.json
- kerberos files (krb5.conf/keyfile)

Migrate to the new wiki

Grep `LocalSettings.php` for `wgDB`

```
grep wgDB LocalSettings.*
LocalSettings.local.php:$wgDBserver = "Your-IP";
LocalSettings.local.php:$wgDBname = "Your-datbase-name";
LocalSettings.local.php:$wgDBuser = "Your-database-user";
LocalSettings.local.php:$wgDBpassword = "Your-password";
```

and create the **database** and **DBuser** with the **DBpassword**. Grant `all` on database to the **DBuser** in mysql. Migrate the data:

```
mysql -u root -p bluespice < migration.sql
cp -r images $wikifolder
cp -r extensions $wikifolder
cp -r LocalSettings.php $wikifolder
setWikiPerm $wikifolder
```

Modify LocalSettings.php

Replace:

```
$wgLogo = "$wgResourceBasePath/resources/assets/wiki.png";
```

with:

```
$wgLogos = [ '1x' => "$wgResourceBasePath/resources/assets/wiki.png" ];
```

- Check other `LocalSettings.*` for individual configurations from the old BlueSpice installation.
- Compare the `migrated settings.d/*.local.php` with the new `settings.d/*.php` and check for individual (probably deprecated) configurations. Do the same with `09*-xx.php`.
- Copy to `$wikifolder/settings.d/` what is needed.

Upgrade step 2

Assign all pages and files without a valid editor user to your newly created substitute user:

```
cd $wikifolder
 php extensions/BlueSpiceFoundation/maintenance/PrepareActorMigration.php --
unknownUserId=<user-id-from-step I. > --unknownUserName=DeletedUser
```

Optional

- Copy the following files (if applicable in your case): `apache.conf`, `krb5.conf,` `Kerberos-key`, `ldapprovider.json`
- Add to VirtualHost in `Apache.conf` after rewrite rules:

```
AllowEncodedSlashes NoDecode
```

Last steps

Run the following maintenance scripts:

```
cd $wikifolder
php maintenance/update.php --quick
php extensions/BlueSpiceExtendedSearch/maintenance/initBackends.php --quick
php extensions/BlueSpiceExtendedSearch/maintenance/rebuildIndex.php --quick

screen
    while  [ "$(php maintenance/showJobs.php)" != "0" ]; do php maintenance/runJobs.
php --maxjobs 100; done
```

The last step will take a while.

Known possible issues after the update

Problems with VisualEditor

Add in `includes/libs/http/MultiHttpClient.php` after line 327:

```
curl_setopt( $ch, CURLOPT_SSL_VERIFYHOST, 0 );
curl_setopt( $ch, CURLOPT_SSL_VERIFYPEER, 0 );
```

Setup:Installation Guide/Webservices/Drawio

To use drawio, the Service has to be installed separately.

Download link:

https://bluespice.com/filebase/drawio/

Setup:Installation Guide/Webservices/Mathoid

Mathoid is a nodjs service which takes various forms of math input and converts it to MathML + SVG or PNG output.

Important! Beginning with BlueSpice 4.x, Mathoid is no longer distributed and has to installed separately.

Download:
https://buildservice.bluespice.com/webservices/REL1_39-4.3.x/mathoid.tar.gz or https://buildservice.bluespice.com/webservices/REL1_39-4.3.x/mathoid.zip

If mathematical formulas are not rendered in your wiki, please check that `extensions/Math/webservices/mathoid.tar.gz` is deployed in the correct folder:

- Linux: `/opt/mathoid`
- Windows: `C:\BlueSpice\bin\mathoid`

PDF-Export integration
The PDF-Export does not support SVGs at the moment. Therefore the SVG needs to be converted into a PNG when embedded into a PDF. This is done automatically, but requires `$wgSVGConverter` to be set up properly. It is recommended to use `rsvg` as a converter. On Debian based linux environments it can be installed with

```
apt install librsvg2-bin
```

In the `LocalSettings.php` file it must be configured with

```
$GLOBALS['wgSVGConverter'] = 'rsvg';
```

Next step

If you have completed all steps successfully and want to install BlueSpice pro, proceed to the next step PhantomJS

Webservice: PDF-Export

Contents

1 Moving the BShtml2PDF Application 57

2 Checking the functionality 57
3 Enable PDF Export in BlueSpice 57
4 Installing BlueSpice 3 with installer 58
5 Troubleshooting 59
6 Download Web services 59

Make sure that the Apache Tomcat server is completely installed and configured on the BlueSpice server before setting up PDF-Export.

Moving the BShtml2PDF Application

Important! Starting in BlueSpice 4.x, BShtml2 is no longer in the distribution and has to be downloaded and installed separately.

Download: https://bluespice.com/filebase/html2pdf/

Download the file "BShtml2PDF.war" and copy it to `<tomcat-webapps>`

The default paths are as follows:

- C:\Program Files\Apache Software Foundation\Tomcat 10.0\webapps **(Apache Tomcat on Windows)**
- /var/lib/jetty9/webapps **(Jetty on Debian)**

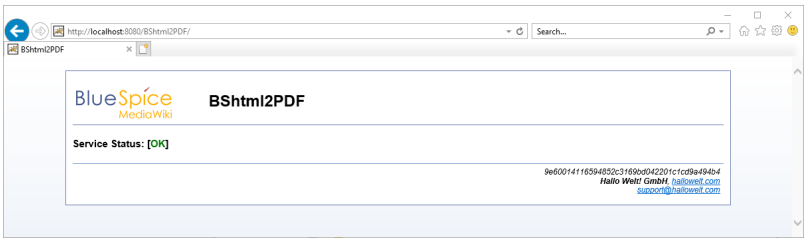
Or run the command `sudo mv /var/www/bluespice/w/extensions/BlueSpiceUEModulePDF/webservices/BShtml2PDF.war /var/lib/jetty9/webapps` in the terminal. **(Linux)**

The application usually starts automatically without the need for rebooting the application server.

Checking the functionality

The correct functionality of the BShtml2PDF service can be checked as follows:

On Windows, open Internet Explorer using the URL http://localhost:8080/BShtml2PDF. With correct functionality, you should see the following website:



Enable PDF Export in BlueSpice

Go to the directory `<installpath-bluespice>/settings.d`. Open the file **020-BlueSpiceUEModulePDF.php** in a text editor.

Or open the 020-BlueSpiceUEModulePDF.php in the terminal using the command `sudo nano /var/www/bluespice/w/settings.d/020-BlueSpiceUEModulePDF.php`. **(Linux)**

In the second line is the code to activate the extension:

```
return; // Disabled. Needs Tomcat
```

You can either add a comment character (#) to the beginning of this line or delete the line completely, save the file and exit it.

Note: Give the user "Everyone" full access to the Temp folder under C:\Widows.

Installing BlueSpice 3 with installer

1. Open `localhost/w` in your Browser and follow the instructions.
2. For BlueSpice WikiFarm, move the .htaccess and web.config files to C:\bluespice\var

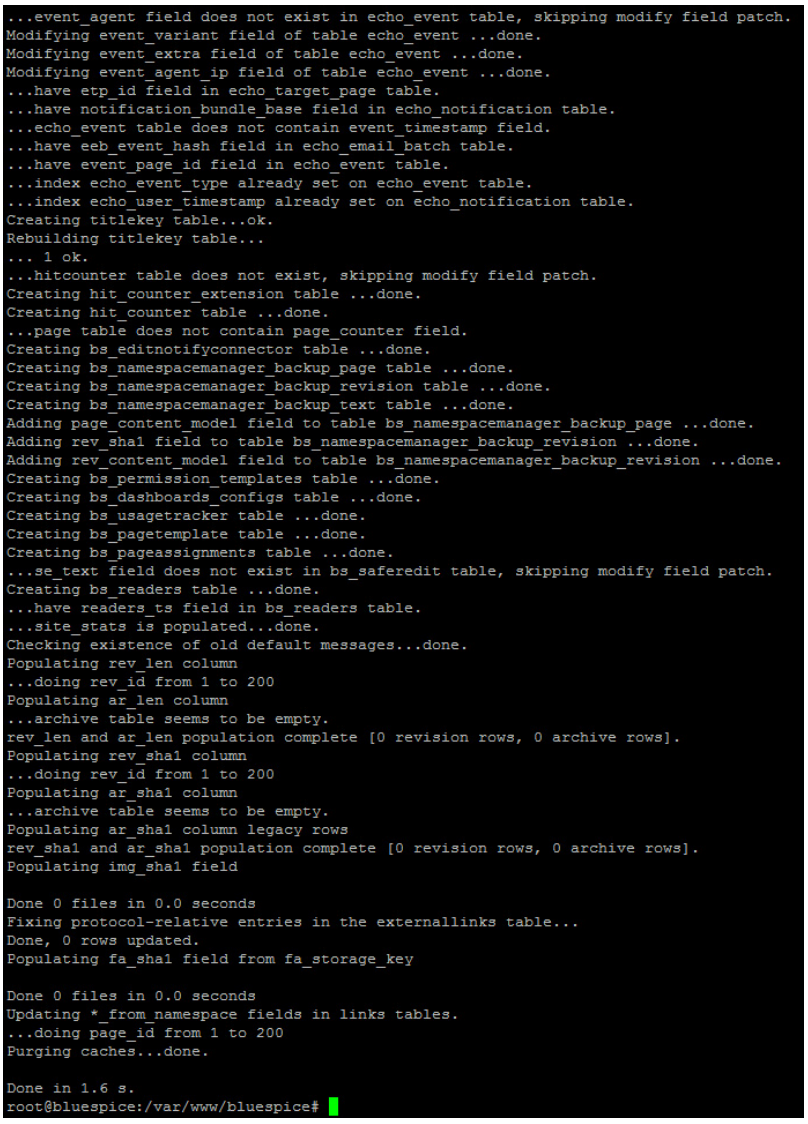
Open a command line and change into the folder `<installpath-bluespice>`. Execute the following command:

```
php maintenance/update.php (Linux)
php maintenance\update.php (Windows))
```

Note: In Windows, make sure that the environment variables are set correctly to access the "php" command.

Note: Note that you must update the file system permissions after running update.php.

When the script is finished with the "Done" message, you have successfully isntalled the web service .



Troubleshooting

If there is a problem with the PDF export in BlueSpice, first check the file system permissions.

Download Web services

If the required war-file is missing in your installation package, you can download it here: Webservices.

Installation BlueSpice 4 with Installer

Contents

1 Download installation package 61
2 Introduction 61
3 Step-by-step installation 61
4 BlueSpice WikiFarm 64
5 Next steps 65
6 Linux .....
7 Windows .....

Download installation package

Download the installation package from the bluespice.com website.

Store the installation package in a web root of your choice (see our folder structure recommendations for Windows installations), where you can access it via your browser.

Make sure that file system permissions are set up properly.

Introduction

BlueSpice 4 is installed as a complete package with MediaWiki 1.39 and BlueSpice 4. This installation guide gives you step-by-step instructions to install BlueSpice.

Important! If you do a fresh install of BlueSpice 4, it might be necessary to run update.php afterwards if you get an error.

This document only describes the steps for installing the web application itself. For the installation of extended functions such as VisualEditor and Extendedsearch, please refer to the corresponding documentation.

We will not go into details about your operating system installation here, but assume that you have already checked the system requirements and your server configuration. You can find more details in our detailed installation guide.

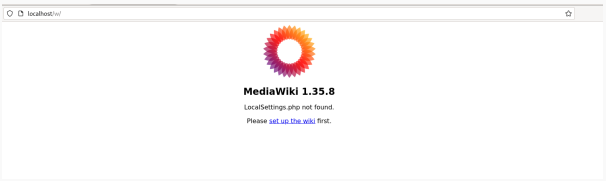
Further help on optimizing the operating system can be found in our compendium.

Step-by-step installation

Note: If you also install the paid extension "BlueSpiceWikiFarm", make sure that the codebase is not located directly in the DocumentRoot of your web server or VirtualHost - it must be located in the /w subfolder!

Note: If Error "Fatal exception of type MWException" shows up, you need to check the rights of your Bluespice Folder and give it reading- and writing permissions. If there is still a error you need to add "$wgShowExceptionDetails = true" to your localsettings.php file and reload your URL to get preciser information of your error.

First use your browser to call up the URL under which the web application can be reached. Then complete the following steps:



Step 1: Start the installation setup

First, open the URL of your web application in your browser. Then click on "set up the wiki" t start the installation.

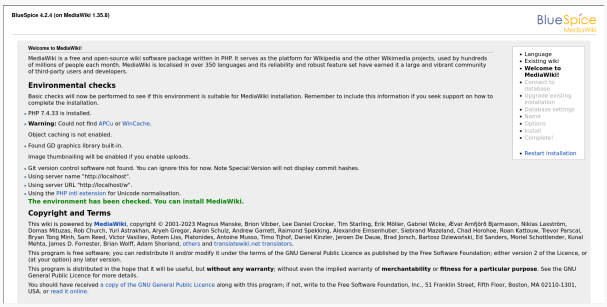
If you don't meet all necessary requirements f PHP yet, you won't be able to view this setup page yet. Please make the necessary corrections to your PHP installation first. Otherwise, you are now on the page shown in the screenshot (click on the thumbnail to see the full screenshot).



Step 2: Select your language

You are now in the language selection. "Your language" defines the language that guides yo through the installer,

"Wiki language" the later language settings of your wiki.



Step 3: MediaWiki installation check

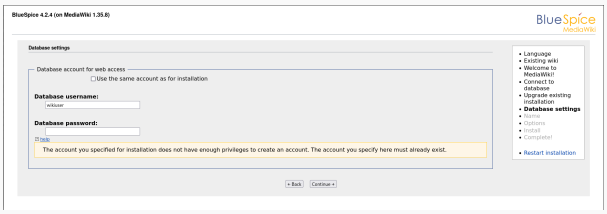
This step takes you to the MediaWiki installatio check. Please read this check carefully, as possible hints may not block the installation itself, but may lead to restrictions of the wiki functionality in practice. Make the necessary corrections to your system and reload the pag in your browser before proceeding with the setup.

Step 4: Establish the database connection

In this step, you configure the database connection. Here you can either specify the ro user of your MySQL server and in the following step have the installer create an additional use who only has access to the BlueSpice databas The database will also be created automaticall if it does not yet exist. Or, alternatively, you ca

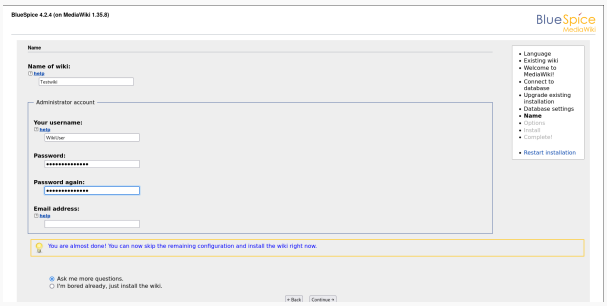


create a user and a database in advance and directly enter this information here. You can fi information on the manual creation of users ar databases in the official Mediawiki documentation.



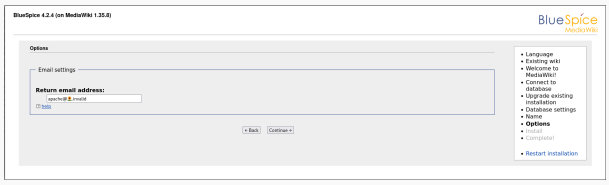
Step 5: Create the BlueSpice database user

Depending on the procedure you followed in th previous step, you can now define an exclusiv user for your BlueSpice database. If you have already created the user explicitly for the database in advance, leave the check mark at "Use the same database account as during the installation process".



Step 6: Create your wiki name and administrator account

Enter the name for your wiki. This name is displayed in the title bar of your browser and i various other places such as in notification mails. Pay extra attention to the correct spellir because this name can afterwards only be changed with a certain effort and certain risks Next, enter the username, password and e-ma address of the first wiki administrator. The administrator account is then created by the wiki during setup.



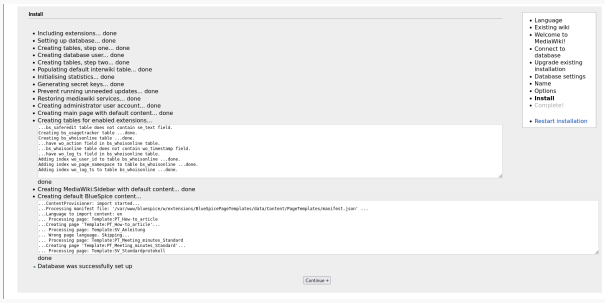
Step 7: Options

In this step, you specify the e-mail address for notification e-mails.



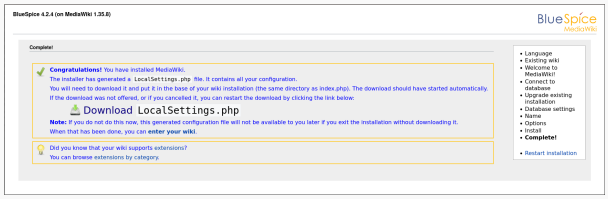
Step 8: Installation

This step announces the installation of the wik If you don't need to make any changes to the previous steps, click "Next" to start the installation process.



Step 9: Installation status

After the successful installation of BlueSpice, you see a confirmation page that you confirm with "Next".

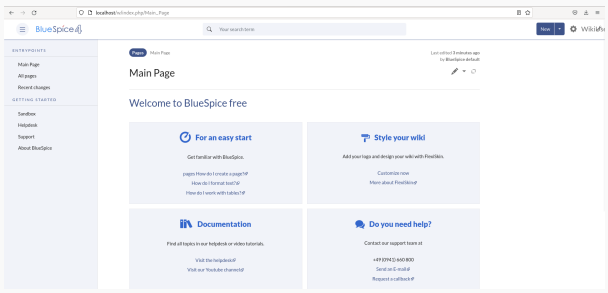


Step 10: Download LocalSettings.php

In the last step of the installation, you will be asked to download the configuration file "LocalSettings.php" of your installation. Download it and place it in the root directory o your BlueSpice codebase.

Step 11: Rename the template folders

1. Open the folder `<installpath-bluespice>\extensions\BluesSpiceFoundati` and rename the subfolder `config.template` t `config` and create the new folder `data`.
   These folders should have read and write acces see File System Permissions.
   Note: Since BlueSpice 4.3.0, `config.template` directory no longer exists, so step of renaming should be skipped
2. Run `php <installpath-bluespic e> \maintenance\update.php` from the console.



Step 12: Load the BlueSpice welcome page

If you now access the URL you specified for BlueSpice, you now see the BlueSpice welcom page. You can start using BlueSpice immediately.

BlueSpice WikiFarm

If you are installing **BlueSpice WikiFarm,** follow these instructions. If not, skip to Next steps.

Now open a command line and change to the folder `<installpath-bluespice>`.

Run the following command:

- `php maintenance/update.php` *(Linux)*
- `php maintenance\update.php` *(Windows)*

Next steps

> **Note:** If the page is not loading, give the group "users" change-permissions for the folder C:\Windows\Temp.

For security reasons, it is recommended to prevent access to the path `mw-config/` after the installation. This can be implemented as follows:

Linux
Windows

In Apache, add the following entry to the file `000-bluespice.conf`:

```
    <Directory /var/www/bluespice/w/mw-config>
        Require all denied
    </Directory>
```

In IIS, create a file `web.config` in the sub-directory `mw-config/` with the following content:

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <system.webServer>
        <security>
            <authentication>
                <anonymousAuthentication enabled="false" />
            </authentication>
        </security>
    </system.webServer>
</configuration>
```

Please note that this entry may have to be removed temporarily if the web installer is used to update the application.

> **Important!** As next step, you can configure VisualEditor and Extendedsearch.

Setup:System requirements

For a trouble-free installation of the current version of BlueSpice 4, we recommend the following system requirements.

Browser

- Microsoft Edge
- Google Chrome
- Firefox

Server Environment

- **Operating system:**
  - We strongly recommend Linux (preferably Debian 11, Ubuntu 22.04, CentOS 7)
  - You might use Windows Server starting at 2016, but we have seen performance issues on Windows Server
- **Webserver:**
  - Apache 2.4.x, IIS >= 10 *or* nginx 1.x (*nginx not possible in WikiFarm*)
- **PHP:**
  - PHP 8.1 / PHP 8.2
- **Database**:
  - MySQL: >= 5.6 or
  - MariaDB >= 10.3
- **(Virtual) hardware requirements:**
  - **CPU:**
    - **Linux: 8 Cores** (min. 4 Cores)
    - Windows: 16 Cores (min. 8 Cores)
  - **Main memory:**
    - **Linux: 16 GB** (min. 8GB)
    - Windows: min. 16 GB
  - **Available hard drive space:**
    - > 20 GB (depends on the planned storage of data)
- **Other:**
  - Apache Tomcat >= 9 oder Jetty >= 9 (for PDF export and LaTexRenderer)
  - ElasticSearch 6.8 with plugin “ingest-attachment”
  - OpenJDK >= 10
  - NodeJS 16