Contents

1. Category:Social 2
2. Manual:All topics 2
3. Manual:Extension/BlueSpiceSocial 6
4. Manual:Extension/BlueSpiceSocialBlog 12
5. Manual:Extension/BlueSpiceSocialWikiPage 29

Category:Social

The page you are looking for does not exist.

Please take a look at our existing help topics.

If you are a contributor and want to create the page, simply log in now.

Pages in category "Social"

The following 3 pages are in this category, out of 3 total.

E

- Manual:Extension/BlueSpiceSocial
- Manual:Extension/BlueSpiceSocialBlog
- Manual:Extension/BlueSpiceSocialWikiPage

Manual:All topics

Helpdesk topics

Setup

Administration

Concepts

Content & Structure

Quality assurance

Personalization

Social wiki

Cloud

Note: If you can't find an answer on these pages, you can also visit our Community Forum.

Setup

- BlueSpice Operating Manual
- BlueSpice free Download
- Installation Guide
- Installing BlueSpice free with Docker
- Release History
- Release Notes
- System requirements
- Upgrade from BlueSpice 3 to 4

Administration

- Config manager
- Footer
- Main navigation
- Extended statistics
- Namespace manager

Permissions

- Group manager
- User manager
- Invite users
- 2-Factor-Authentication (2FA)

Concepts

- Creating and using categories
- Magic words
- Subpage
- The pages model
- Wikitext

Content and structure

- Attachments
- Content organization
- Checklists
- Ratings and recommendations
- Tag cloud
- Tasks - Overview
- Templates

Editing

- Create Pages
- Create pages
- Delete pages
- Extension/BlueSpiceCategoryCheck
- Context menu
- CountThings
- Filtering tables
- Insert links
- Insert tags and magic words
- Unsaved changes (autosave)
- Drawio
- Edit pages
- Insert files
- Insert images
- Version history
- Wikitext

Organization

- Content organization

- Creating and using categories
- Books
- Category manager
- Interwiki links
- Generate page lists (smart lists)
- Creating page lists with DPL3
- Menus - Overview
- Page lists
- Page templates
- Recent changes transclusion
- SMW queries
- Templates

Page tools

Search

- Extended search
- Interwiki search
- Search field in the page content (TagSearch)

Customization

- Displaying page sections as cards
- Error messages
- Custom menu
- Main navigation
- Namespace CSS manager
- Attachments
- Creating page lists with DPL3
- Integration of third party content
- Numbering ordered lists

Personalization

- Avatars
- Dashboards
- Notifications
- Visited pages
- Privacy center
- User menu
- Personal navigation
- Preferences

Quality assurance

- Collecting signatures

- Extension/BlueSpiceExpiry
- Page assignments
- Read confirmation
- Reminders
- Extension/BlueSpiceWikiExplorer
- Page approvals (Acceptance)
- Quality management
- Redirects

Maintenance

Social wiki

- Introduction to BlueSpice Social
- Blog
- Attachments

Cloud

BlueSpice pro Cloud topics:

- PDF Customization (Cloud only)
- Customizing the user interface (FlexiSkin)
- Invite users

Social Wiki

BlueSpiceSocial is a set of extensions that add social features around a wiki page. It allows users to quickly communicate on the wiki and to express their opinion on pages.

Contents

1 Scope 8
2 User profile 8
3 Discussions 9
4 Timeline 10
4.1 Creating new entries 10
4.2 Interacting with social entities 10

5 Blog 11
6 Permissions 11
7 Deactivating the comments feature 11
8 Related info 11

Scope

BlueSpice Social contains the following functionality:

- **User profile:** Provides the page *User:<username>.*
- **Attachments:** Connects uploaded files to wiki pages.
- **Discussions:** Enables discussions with comments created on wiki pages. Discussion entries can be marked as "closed".
- **Activities:** Automatically creates pages in the *SocialEntity* namespace to track page edits and creations.
- **Blog:** Provides the special page *Special:Blog*. A blog view can also be embedded into individual wiki pages.
- **Microblog:** Similar to blog entries, but without a headline.
- **Timeline:** A chronological list of all social entities. The timeline filter can be used for displaying social entities on wiki pages.

User profile

Every user has a profile that is displayed on the page *User:<username>*. Users can access their user profile by clicking on *Profile* in the user menu (located in the header bar).

The user profile shows the user's basic information (depending on the wiki setup) and the user's profile image.

Below the user info, the user's wiki activities are listed: Comments the user made, actions on pages, and in general the user's activity on the wiki. It also allows other users to leave comments on the user's profile page.

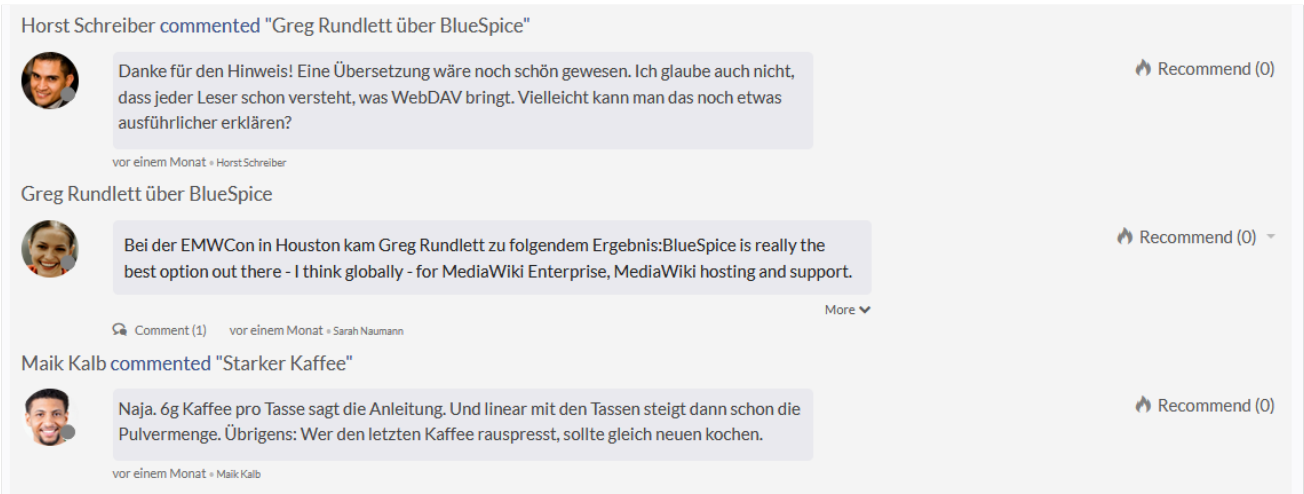
User profile

Discussions

BlueSpiceSocial integrates a discussion section at the bottom of every wiki page. This section allows users to create comments directly on a wiki page. If it is the first entry in a discussion, the discussion page is automatically created at this time if this is configured in the Configuration manager.

The discussion section can be disabled on the page-level by adding the MagicWord *__NODISCUSSION__* to the page in source editing mode.

Timeline



Timeline

Timeline is a special page that provides an overview of all social conversations on the wiki.

By default, it is listed on the *Main navigation*. You can also access the page directly at `Special:Timeline`. On this page, users can sort and filter existing social entries or create new entries.

Creating new entries

To create a new entry, click on the "Plus" icon to add a new timeline entry:

- **Blog entry**: Blog is used for topics to be shared with other users on the wiki
- **Microblog**: Like a blog entry, but without titles. It should be used for a short entry, e.g., a quick announcement.
- **Discussion topic**: Adds a discussion entry to a particular wiki page.

Interacting with social entities

Every entry in the timeline is called a social entity. Depending on the type of social entity, the following interaction options exist:

- **Comments:** Number of comments related to the entity. Clicking on *Comments* opens all current comments and allows users to add a comment (if user has the necessary permissions).
- **Recommend:** Number of users that recommended the entity. Uers can add a recommendation by clicking on *Recommend*.
- **Linked articles:** Number of articles linked to this entity Clicking on *Linked articles* shows the list of linked articles.
- **Watch/Unwatched:** Shows if the user is currently watching the entity (the receives notifications when the entity is changed). Clicking on Watch or Unwatch changes the state of watching.
- **Resolved/Unresolved**: The status of the entity. Some entities, like discussions on a question can have a status attached to them, so that other users know if the question is answered.

Blog

Blog is a special page showing an overview of all blog entities available on the wiki. By default, it is listed on the *Main navigation.* You can also access the page directly at `Special:Blog`. Users can sort and filter the list entries by clicking on the icons above the list.

Learn how to include the blog on any wiki page.

Permissions

From BlueSpice version 3.2.7 the following roles have associated "social" permissions. Earlier versions have no option to limit the creation and editing of blog entries by using the role *commenter.*

Rechte/Rolle	commenter	editor	admin	maintenance-admin	Berechtigung
Create and edit blog entry		x	x	x	social-blog
Comment on blog entry	x	x	x	x	social-blog-comment
Create and edit microblog entry	x	x	x	x	social-microblog
Comment on microblog entry	x	x	x	x	social-microblog-comment
Create discussion entry	x	x	x	x	social-topics
Comment on discussion entry	x	x	x	x	social-topics-comment
Create and edit attachments	x	x	x	x	soical-stash

The comments function for blog, microblog and discussion entries can be deactivated in the LocalSettings-file on the server.

Deactivating the comments feature

The comment function for comments on blog, microblog and discussion entries can be deactivated in the LocalSettings file on the server.

```
$bsgSocialCanHaveChildrenBlacklist = [
        'blog', // disable comments on blog
        'topic', // diable comments on discussion topics
        'microblog' // disable comments on microblogs
];
```

Related info

- Blog
- Discussions and Attachments
- Reference:BlueSpiceSocial

Technical Reference: BlueSpiceSocial

Blog

The extension *BlueSpiceSocialBlog* is part of *BlueSpiceSocial*, the communications module of BlueSpice. In addition to the type blog, there are other types of so-called social entities, such as *comment*, *discussion topic*, *attachment*, or *microblog*.

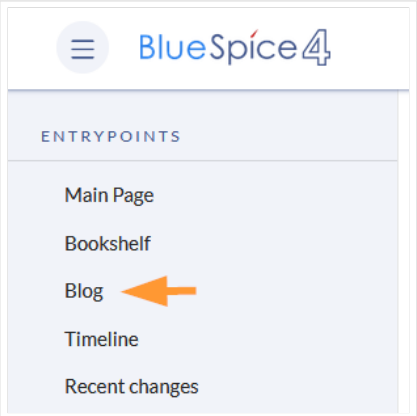
Contents

1 About the blog 14
2 Characteristics and features 14
3 Adding the blog to a page 14
3.1 All blog entries 15
3.2 Blog entries of a linked page 15
4 Blog parameters 16
4.1 Output format 16
4.2 Sorting 20
4.2.1 Syntax 20
4.2.2 Properties 20
4.2.3 Direction 20
4.3 Filtering 21
4.4 Filtering associated pages 21
5 Output type examples 22
5.1 Default 22
5.2 Short 22
5.3 Page 23
5.4 List 23
6 Setting blog permissions 23
6.1 Using the "commenter" role 23
6.2 Using groups 24
7 Configuration 26
8 Notifications 26

9 Show a list of recent blog entries 27
9.1 Embed the timeline 27
9.2 Define formatting instructions 28
10 Related info 29

About the blog

Basically, a blog is a list of comments, questions, or feedback from different users. The blog is often sorted chronologically. Lively communication can arise when blog entries are further commented, recommended, linked with related articles, or placed on a personal watchlist. The blog is ready to use on the special page *Special:Blog*. This page is linked directly from the main navigation in every new wiki installation. So you can start using the blog right away.



Navigation link to the blog page

Characteristics and features

- A blog entry consists of a title and content.
- **Comments:** Users can leave comments to each blog entry.
- **Actions:**
  - **Delete:** Editors can delete their own posts. Administrators can delete all posts. Deleted posts are no longer displayed. Exception: Administrators see deleted items with a red background and can restore them.
  - **Recommend:** Blog posts can be recommended. It is possible to generate a list of posts with the most recommendations via the timeline filter. (See: Embedding the blog on a page)
  - **Linked page:** Every blog entry is connected to a wiki page by the system. By default, this is the main page of the wiki. However, other pages can be added as "Linked page". Blog entries can then be filtered for linked pages via the *Special:Timeline* page.
  - **Watch:** Own blog posts and comments are automatically added to the watch list.
- **Permissions:** It is possible to allow users with read permissions on the wiki to comment on blog entries. See: BlueSpiceSocial Permissions.

Adding the blog to a page

It is also possible to include the blog functionality on any wiki page. Social contributions of many types can be consolidated and displayed chronologically in a timeline and filtered and sorted as needed.

All blog entries

If you want to include a view of your blog on another page of your wiki, for example your homepage, you need to insert the necessary codeblock in the source code of your page.

To include a blog view:

1. **Go to** the page where you want to insert the blog or create a new page.
2. **Open the page** in *Source editing* mode.
3. **Copy the following code block** to the section in your page where you want to show the blog.

```
<bs:timeline>
 {
     "showentitylistmenu": true,
     "preloadtitles": {
         "blog": "Template:Blog"
     },
     "preloadedentities": [{
         "type": "blog"
     }],
     "headlinemessagekey": "Community-Blog",
      "showheadline": true,
     "usemorescroll": false,
     "morelink": "Special:Blog",
     "limit": 5,
     "sort": [ {
         "property": "timestampcreated",
         "direction": "DESC"
     }],
     "lockedfilternames": [
         "type"
     ],
     "filter": [ {
         "type": "list",
         "property": "type",
         "value": [ "blog" ],
         "comparison": "ct"
     }]
 }
</bs:timeline>
```

After saving the page, you will see a list of existing blog entries based on the filter and sorting criteria you provided.

Blog entries of a linked page

If you want to create a blog that only displays the posts of the blog posts created on this page, copy the following timeline tag. Here, under the preloadedentities, new blog entries are already linked to the page and filtered. Replace Blog1 under preloadedentities and in the filter section with the appropriate page name:

```
<bs:timeline>
 {
     "showentitylistmenu": true,
     "preloadtitles": {
         "blog": "Template:Blog"
     },
       "preloadedentities": [{
         "type": "blog",
         "tags":["Blog1"]
     }],
```

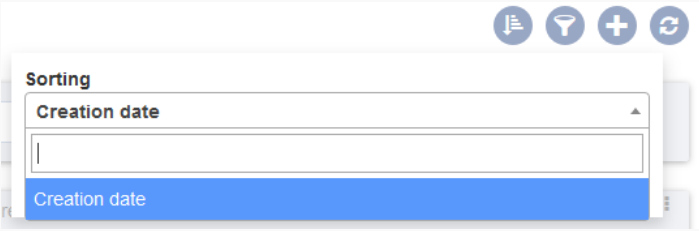
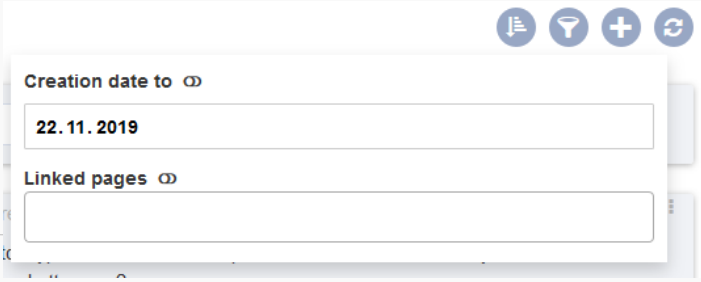
```
      "headlinemessagekey": "Community-Blog",
       "showheadline": true,
      "usemorescroll": false,
      "morelink": "Special:Blog",
      "limit": 5,
      "sort": [ {
          "property": "timestampcreated",
          "direction": "DESC"
      }],
      "lockedfilternames": [
          "type"
      ],
 "filter": [ {
        "property": "type",
        "value": ["blog"],
        "comparison": "ct",
        "type": "list"},
       {
        "property": "tags",
        "value": ["Blog1"],
        "comparison": "ct",
        "type": "list" },
       {
        "property":"archived",
        "value":false,
        "type":"boolean",
        "comparison":"eq"}
   ]

  }
</bs:timeline>
```

Blog parameters

Output format

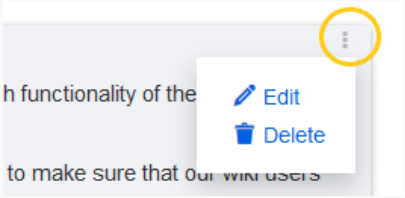
The following table shows the blog elements and the necessary paramaters to create the output.

Blog output

Element	Parameters	Parameter and element description	Type	Def va
1 - header	headlinemessagekey	Header text	string	*Tim*
	showheadline	Show the header text	boolean	*true*
2 - blog menu	showentitylistmenu	Show or hide the blog menu	boolean	*true*
	showentityspawner	Show the menu item for creating a blog entry	boolean	*true*
	availablesorterfields	Determines which data fields are included as sorting options in the sort menu. Example: allow sorting only by date created or by the user who created a blog entry: `"availablesorterfields": ["timestampcreated","ownerid"]`  List of typical sort fields	array	all
	lockedoptionnames	Determines which settings for sorting the user cannot change.	array	[]
	availablefilterfields	Determines which data fields are included as filter options in the filter menu. Example: allow filtering based on Creation date and Linked pages: `"availablefilterfields": ["timestampcreated","tags"]` 	array	all
		Determines which settings for filtering the user cannot change.		

Element	Parameters	Parameter and element description	Type	Def va
	lockedfilternames	Example: a user cannot change the filter values for *Linked pages.* The text field is greyed out: `"lockedfilternames": ["tags"]` 	array	[]
	persistsettings	Initial settings and custom sort and filter settings are included in the page url. The url can then be used to create a link to the page with the same filter and sort settings. `"persistsettings":true` 	boolean	*fals*

3 - new blog entry	preloadentities	Shows the text field for creating a new blog entry. If it is not set, the text field is not available.
		To show the text field:
		`"preloadedentities": [{"type":"blog"}]`
		If the wiki page that has a `<bs:timeline>` tag should only show its own blog entries, then all blog entries need to be automatically tagged with the current page name:
		`"preloadedentities": [{`
		`"type": "blog",`
		`"tags":["Blog1"]`
		`}]`
			array of objects	-

Element	Parameters	Parameter and element description	Type	Def va
		In the example above, `Blog1` is the associated page for all blog entries that are created on this page. The blog filter has to be adjusted accordingly to take the associated page into consideration.		
4 - existing blog entry	outputtypes	It defines the display format of the blog entry. Standard types are: *List, Short, Default, Page.* (see examples) Each blog entry is a "social entity".	object	*Def*
	limit	Defines how many social entities are loaded initially.	object	*Def*
4a - avatar	-	Shows the avatar of the user who created the blog entry or comment. It is displayed in all output types except *List.*	-	-
4b - info area	-	Link to the comments and "last edit" info.	-	-
4c - blog entry content	-	Shows the text body of a blog entry. The output type *Default* collapses the blog entry content after the first paragraph with a "More" link.	-	-
4d - actiion menu	-	Actions: *edit, delete, recommend, Linked page, watch* 	-	-
5 - new comment	-	Text field for creating a new comment.	-	-
6 - existing comment	-	Comments include the social tag *Recommend.* Each comment is a "social entity".	-	-
	morelink	Link target of the "more" button. For blogs, it makes sense to link to the page *Special:Blog*.	str	*Spe Tim*
		More blog entries are automatically loaded after the user		

Element	Parameters	Parameter and element description	Type	Def va
7 - "more" button	useendlessscroll	scrolls to the end of the blog.	boolean	*true*
	showentitylistmore	Shows a "more" button below the blog which loads more blog entries on click. This is only possible if *useendlessscroll* is set to *false*.	boolean	*fals*

Sorting

Most commonly, a blog shows the latest blog entries first. However, there are many different options for sorting the list of blog entries.

Syntax

```
[{
"property": "timestamptouched",
"direction": "DESC"
}]
```

Properties

There are some common properties of a blog entry that are useful for sorting:

Property	Description
timestampcreated	creation date
timestamptouched	last edit date
commentcount	number of comments
ratingcount	number of ratings
ownerid	username of the creator of a blog entry
ownerrealname	real name of the user

Direction

Blog entries can be sorted in descending and ascending order. Descending is the default order.

Property	Description
DESC	sorts the blog entries in descending order
ASC	sorts the blog entries in ascending order

Filtering

Without setting the filter parameter, there will be no output on the page.

To show the blog entries of your wiki, include the following codeblock like shown in the code example above:

```
"filter": [ {
         "type": "list",
         "property": "type",
         "value": [ "blog" ],
         "comparison": "ct"
     }]
```

This filter includes all social entities which match the type *blog*. Since we only want to show social entities of type *blog*, we only include *blog* entries in our filter. If you want to create a social timeline rather than a blog view, you can also add other types of social entities. The following example shows a timeline view that includes blog entries, discussion entries from wiki pages, and user profile information:

```
"filter": [ {
         "type": "list",
         "property": "type",
         "value": ["blog","topic","profile"],
         "comparison": "ct"
     }]
```

Filtering associated pages

If an embedded blog should only show posts that are associated with a paritcular page name (e. g., Blog1), this must be defined in the blog parameters (see *preloadedentities*) and then specified in the filter:

```
"filter": [ {
       "property": "type",
       "value": ["blog"],
       "comparison": "ct",
       "type": "list"},
      {
       "property": "tags",
       "value": ["Blog1"],
       "comparison": "ct",
       "type": "list" },
      {
```

```
        "property":"archived",
        "value":false,
        "type":"boolean",
        "comparison":"eq"}
  ]
```

Output type examples

The output type for a blog is added using the *outputtypes* parameter. Don't forget to include a comma before and after this parameter (unless this is the last parameter in your code):

```
"outputtypes": {
        "blog": "Default"
    }
```

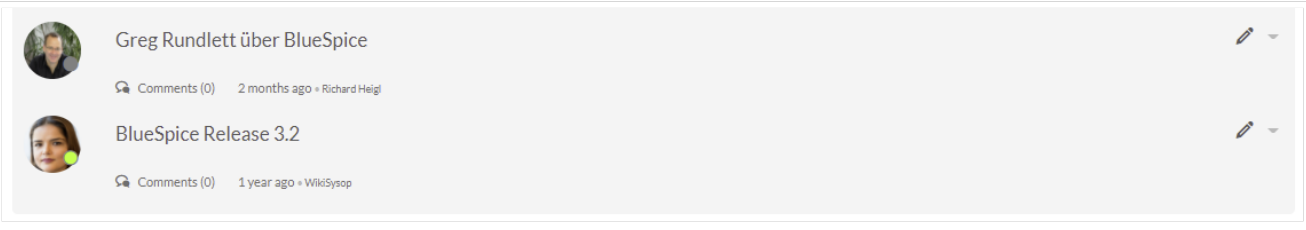
Default

The output type *Default* shows the content of a blog entry truncated after the first paragraph. To view the full text, users can click on the *More* link.

"Default" output

Short

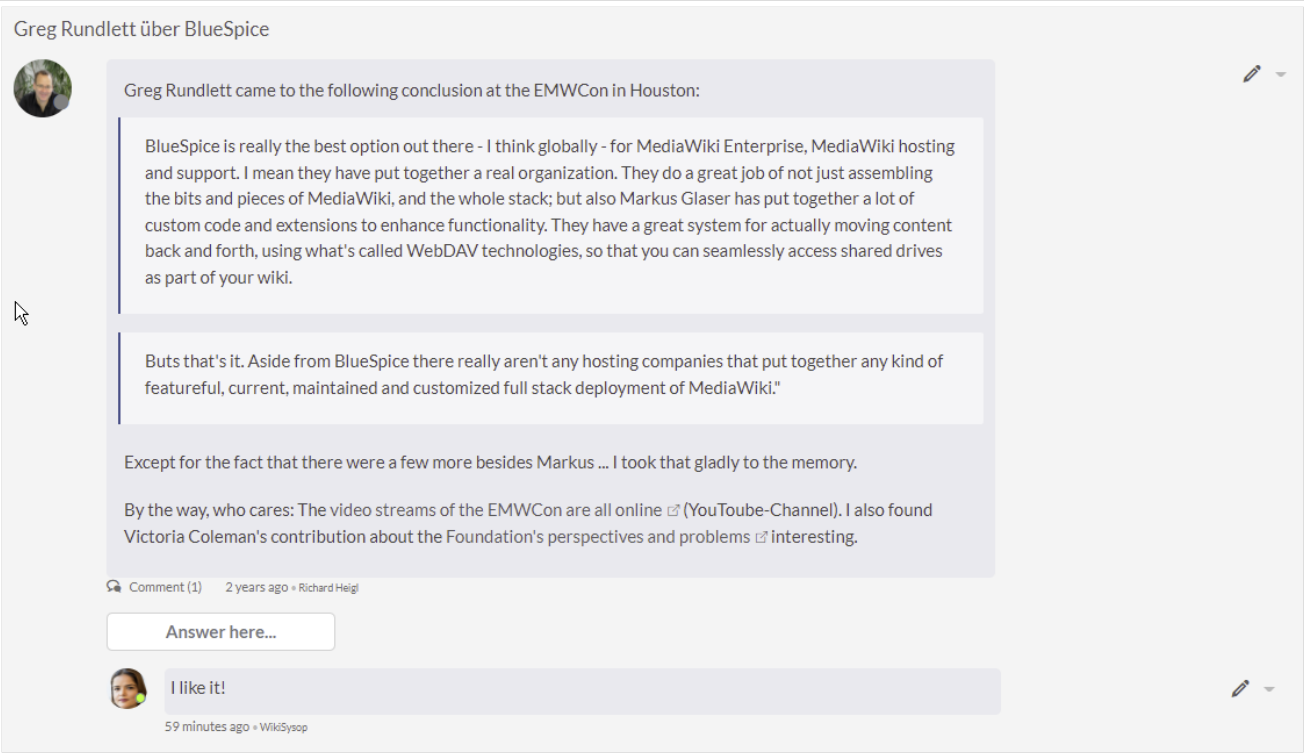
The output type *Short* does not show the content of the blog entry. To view the content, a user has to click on the title of a blog entry. This loads the actual page for the selected blog entity.



"Short" output

Page

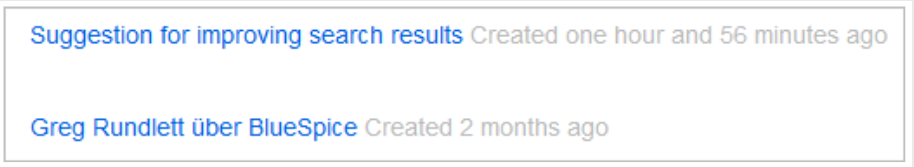
The output type *Page* shows all blog elements, including the comments, in an expanded view.



"Page" output

List

The output type *List* only shows the title and the timestamp.



"List" output

Setting blog permissions

Using the "commenter" role

In general, all users with edit rights can create blog entries and comments. However, the *commenter* role in the Permission manager does not allow you to create blog entries. This role only assigns the right to create comments on existing blog entries. See also BlueSpiceSocial permissions.

Using groups

is available from BlueSpice 4.2.

If a blog is integrated into a page, the creation and editing of discussion posts and comments can be restricted to certain user groups. This option does not exist on the *Special:Blog* page itself.

Add the blog to a wiki page as follows. Then, modify the *editgroups*, c*ommentgroups, deletegroups,* and r*eadgroups* as needed.

Important! Only set the groups that you really need, since individual rights are not inherited. Therefore, please read the following explanations carefully!

```
<bs:timeline>
{
    "showentitylistmenu": true,
      "preloadedentities": [{
       "type": "groupblog",
       "editgroups": ["blog_GF"],
       "commentgroups": ["blog_commenter"],
       "deletegroups": ["blog_delete"],
       "readgroups": ["blog_reader","blog_GF","blog_commenter","blog_delete"]
    }],
    "headlinemessagekey": "My Blog",
    "showheadline": true,
    "usemorescroll": true,
    "limit": 5,
    "sort": [ {
        "property": "timestampcreated",
        "direction": "DESC"
    }],
    "lockedfilternames": [
        "type"
    ],
    "filter": [ {
        "type": "list",
        "property": "type",
        "value": [ "groupblog" ],
        "comparison": "ct"
    }]
}
</bs:timeline>
```

The object *preloadedentities* defines the different permissions:

Name	Value (examples)	Description
type	groupblog	The type *groupblog* allows setting group-based permissions on a blog.
editgroups	["blog_GF"]	Permission to create blog entries
commentgroups	["blog_commenter"]	Permission to create comments

Name	Value (examples)	Description
deletegroups	["blog_delete"]	Permission to delete blog entries and comments
readgroups	["blog_reader"," blog_GF"," blog_commenter"," blog_delete"]	Read permission: All groups listed in editgroups, commentgroups and deletegroups must also be listed here explicitly if this group is defined.

Important notes:

1. If the blog already exists on a page and the group assignments are created later, the permissions will not be applied to existing blog entries.
2. There is no inheritance if a group definition is given explicitly. E.g., a user group that is in editgroups but not in commentgroups (if exists) cannot add comments.

“editgroups” exists:

- Only users in these groups see the button to start a new blog entry.

“commentgroups” exists:

- Only users in these groups see the “answer here” button and can add comments.

“deletegroups” exists:

- Only users in these groups see the delete link and can delete o**ther users'** posts. Note: “sysop” users can always delete or restore a post. See below for sysop user read permissions.

“readgroups” exists:

- Only set if necessary, as there is no inheritance. If present, all user groups of the other blog groups must be added here.

Examples:

- If the *blog_GF* user group is in *editgroups* but not in *readgroups*, the create button is not displayed despite explicit editing rights. If the *blog_commenter* user group is in *commentgroups* but not in *readgroups*, the new blog entries that have this setting applied will not be visible. If the user group *blog_delete* is in *deletegroups* but not in *readgroups*, the new blog entries are not visible and therefore cannot be deleted.

user group “sysop”:

- If *readgroups* is not defined, group sysop can delete all blog entries
- If *readgroups* is defined and sysop is not part of this group, sysop cannot see blog posts and therefore cannot delete them from the page. But with a direct link to the actual blog page, the sysop user can delete/restore the entry from there.

Important! The default user group "sysop" cannot be added as a user group to a definition. For example, to assign read rights to an admin user (if "readgroups" exists), the user must be added to a user group defined in "readgroups" (e.g. blog_reader)

more”-button:

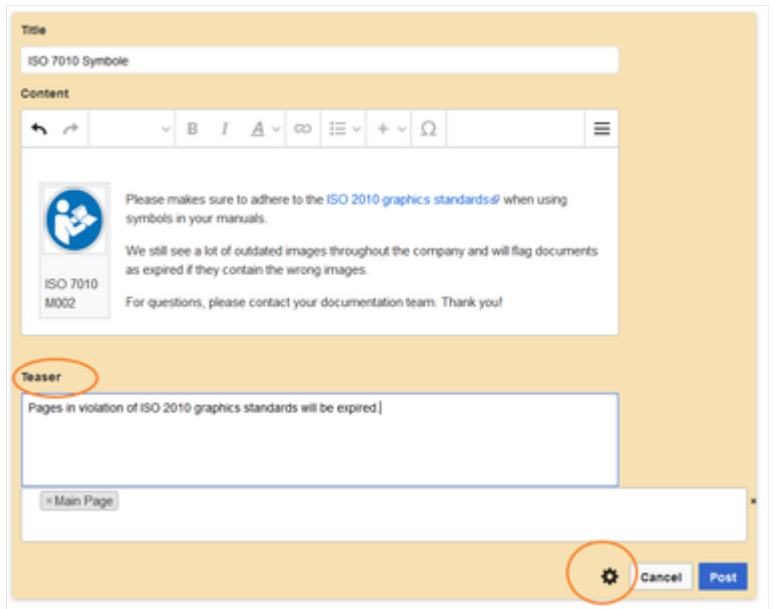
To avoid switching to the default *Special:Blog* page at the end of the blog, the m*ore*-button must be set as a scroll button instead of linking to the special page. To do this, set (as already indicated in the example):

```
"usemorescroll": true
```

Configuration

In the Config manager, you can change the following settings:

- **Use teaser text for blog entries in the timeline:** When you create a blog entry, you can create a teaser text for it by clicking the cogwheel symbol in the blog entry form. If this setting is active, the teaser text will be shown on the page *Special:Timeline* instead of the first part of the content.

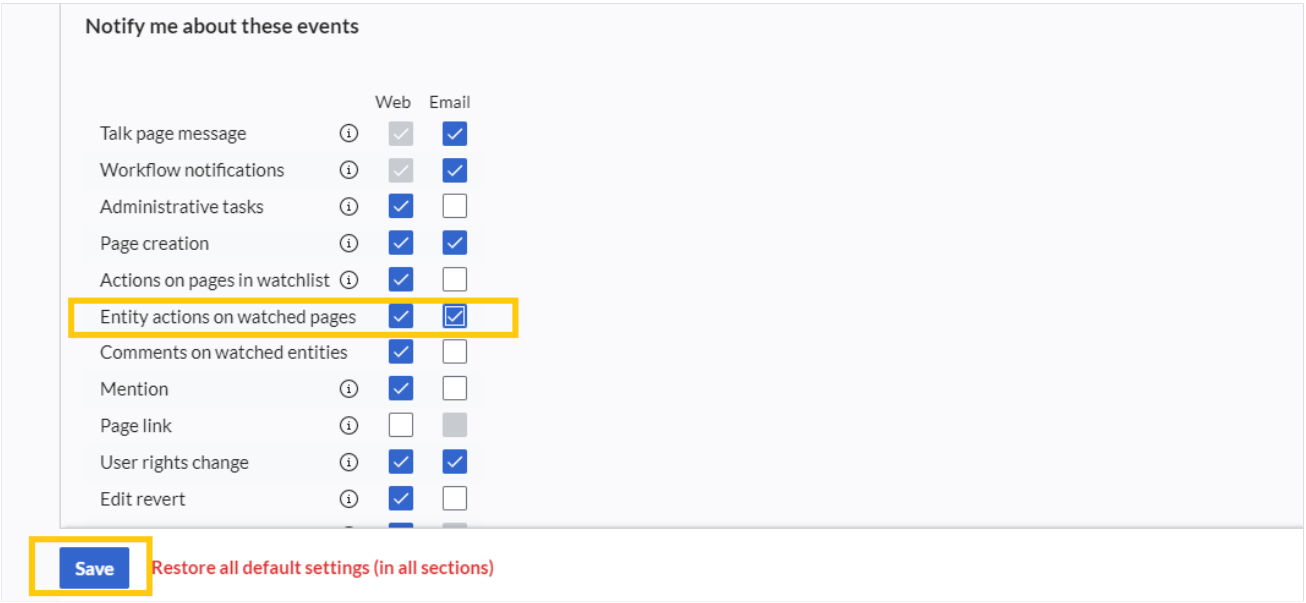


blog teaser text

- **Show entrypoint "Blog":** Shows a link to the page *Special:Blog* in the main navigation.

Notifications

If users want to receive notifications about blog posts in the wiki or by email, the setting "Entity actions on watched pages" must be activated in the user settings ("Notifications" tab).



Notification settings

Show a list of recent blog entries

If you want to display the latest blog entries of a specific blog (one that is associated with a particular page), for example on the main page, the following steps are necessary:

1. Embed the timeline tag on the main page
2. Define formatting instructions in *MediaWiki: Common.css*

Recent blog posts

Embed the timeline

Copy the following timeline tag to the main page (or any other wiki page)

```
<div class="bloglist"><bs:timeline>
 {
     "showentitylistmenu": false,
     "preloadtitles": {
      "blog": "Vorlage:Blog"
     },
     "limit": 5,
     "sort": [ {
         "property": "timestampcreated",
         "direction": "DESC"
     }],
     "showentitylistmore": false,
     "availablefilterfields": [],
     "lockedfilternames": [
         "type"
     ],
     "outputtypes": {
        "blog": "List"
```

```
      },
     "filter": [ {
        "property": "type",
        "value": ["blog"],
        "comparison": "ct",
        "type": "list"},
       {
        "property": "tags",
        "value": ["Blog1"],
        "comparison": "ct",
        "type": "list" },
       {
        "property":"archived",
        "value":false,
        "type":"boolean",
        "comparison":"eq"}
  ]
 }
</bs:timeline>
... [[Blog1|all entries]]
</div>
```

Note that the value ***Blog1*** in the filter and after the timeline tag as a link to the actual blog page must be replaced with the real name of your blog page. For the value *Blog1* or the name of your blog pageto be automatically generated for all blog posts on the actual blog page, it must be specified in the *<bs:timeline>* tag under the preloadedentities object as the value for the key "tags":

```
"preloadedentities": [{
    "type": "blog",
    "tags":["Blog1"]
}]
```

Define formatting instructions

Copy the following css-declarations into *MediaWiki:Common.css*:

```
/* Simple list view of timeline items */
.bloglist .bs-social-entity .bs-social-entity-right .bs-social-entity-actions, .bs-
social-entity-timecreated {display:none}
.bloglist  ul.bs-social-entitylist {background-color:transparent!important; list-style-
type:disc; padding:0!important}
.bloglist  ul.bs-social-entitylist li {margin:0!important; border-bottom:1px dotted #74
747488}
.bloglist  .bs-social-entity div.bs-social-entity-right {margin: 0}
.bloglist  .bs-social-entity-actions {display:none!important}
.bloglist  .bs-social-entity .bs-social-entity-title h3 {font-size: 1em!important; marg
in:0;}
.bloglist  .bs-social-entity-title {width:100%; margin:0!important}
#content .bloglist  .bs-social-entity-right a, #content .bloglist .bs-social-entity-
right a::before {color:#08529d!important}
```

If you don't see the styles applied as in the screenshot above, you might have to do a page refresh (`Ctrl+F5`)

Related info

- Micro-Blog
- Reference:BlueSpiceSocial

Technical Reference: BlueSpiceSocialBlog

Attachments

The extension **SocialWikiPage** allows users to attach a file directly to a wiki page.

Contents

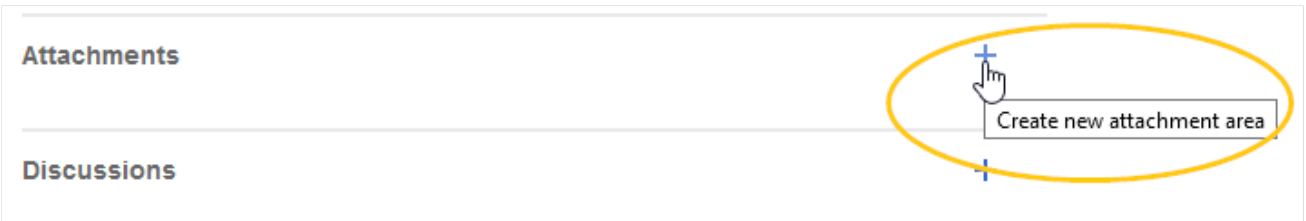
1 Characteristics of file attachments 30
2 Attaching a file 30
3 Deleting an attachment 31
4 Grouped attachments 32
5 Styling 32
6 Related info 33

Characteristics of file attachments

- The attachment process simply creates an association between a file in your wiki and the page to which it is attached.
- File attachments don't show up in the version history of a page.
- Clicking on an attachment, for example a PDF file, opens the file directly (if possible). If the file type is docx or xlsx, the file is downloaded since it cannot be opened in the browser.
- Administrators can still see attachments that have been deleted from a page. These attachments are marked with a red background and can be reactivated if needed.
- The Timeline can list all attachments in the wiki.

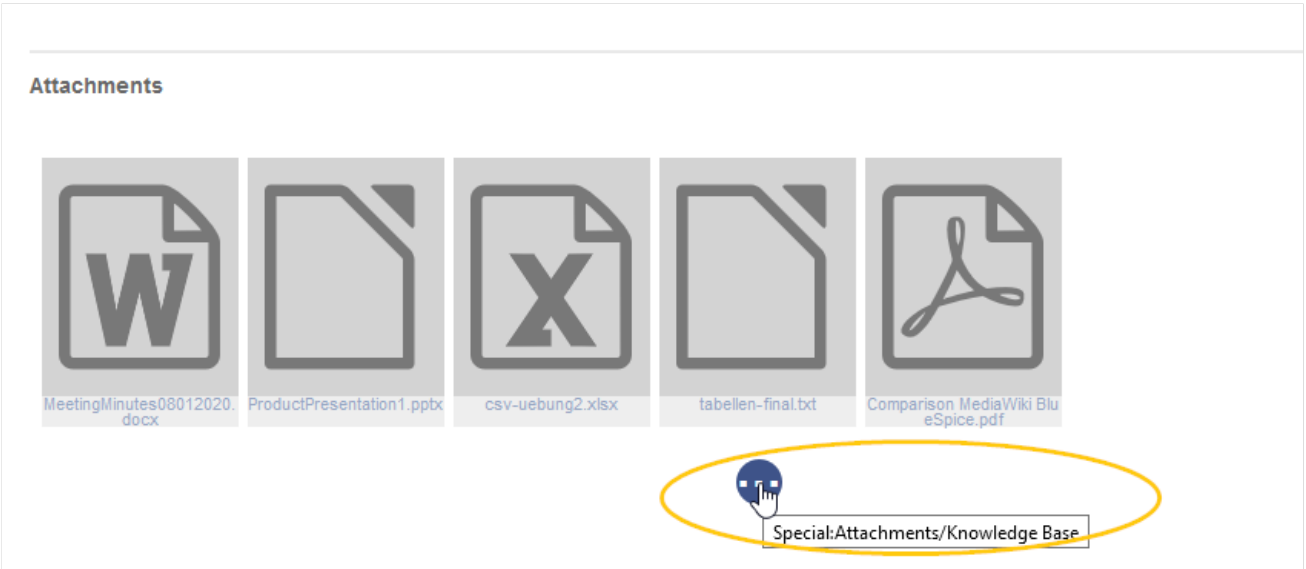
Attaching a file

To create the first attachment at the bottom of a wiki page, open the attachments area by clicking on the "plus" button.



Attachments area

If the attachments area already exists, click the "more"-button to switch to the attachments page.

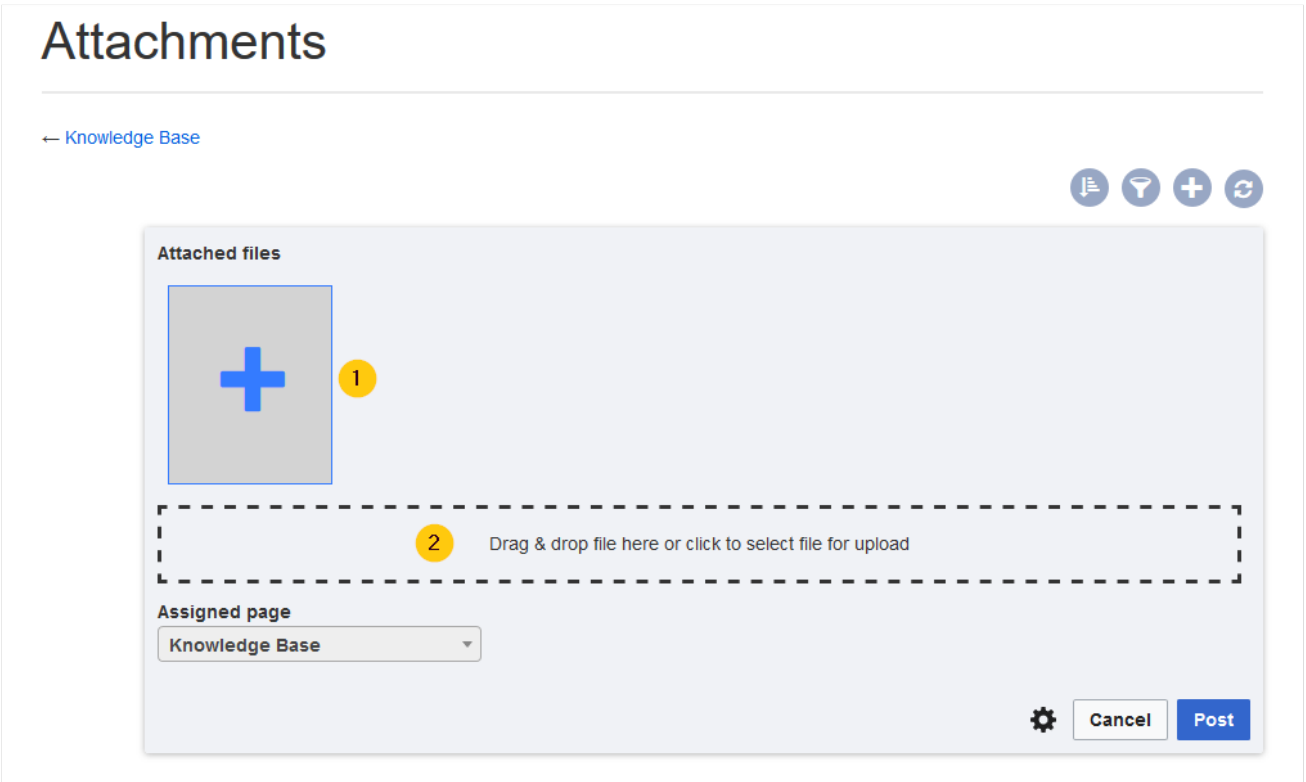


Switch to the attachments page

On the attachments page you can add a new attachment:

(1) **Add existing file:** Click the plus button to add a file that has already been uploaded to the wiki.

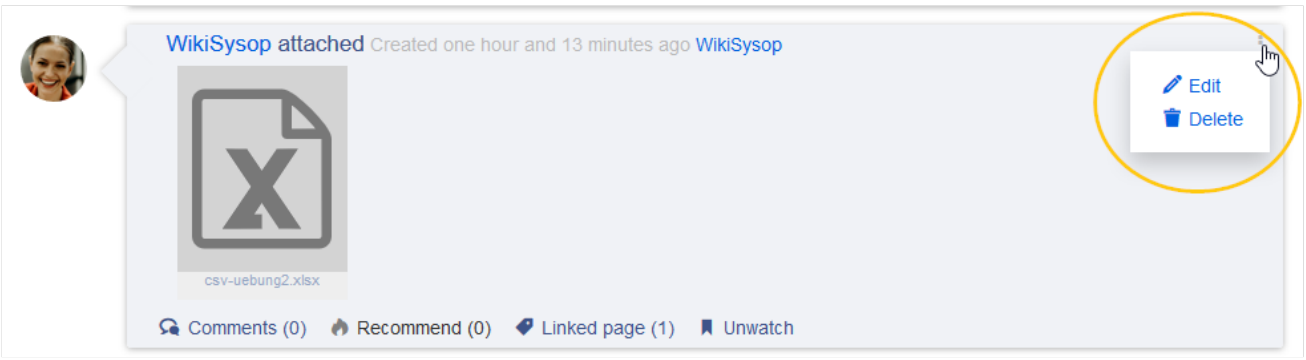
(2) **Upload a new attachment:** Click the upload area or drag the file into the upload field to upload a new attachment to the wiki.



Add an attachment

Deleting an attachment

To delete an attachment, go to the attachments page and click on the Delete button in the actions menu of the attachment. Deleting an attachment on a page does not delete the file itself from the wiki.



Delete attachment

Grouped attachments

Clicking the Edit button in the actions menu of an attachment allows you to group more attachments with it. When you group attachments, they are:

- displayed together on the page
- considered one entity for commenting, recommending, linking, and watching (as social entity)

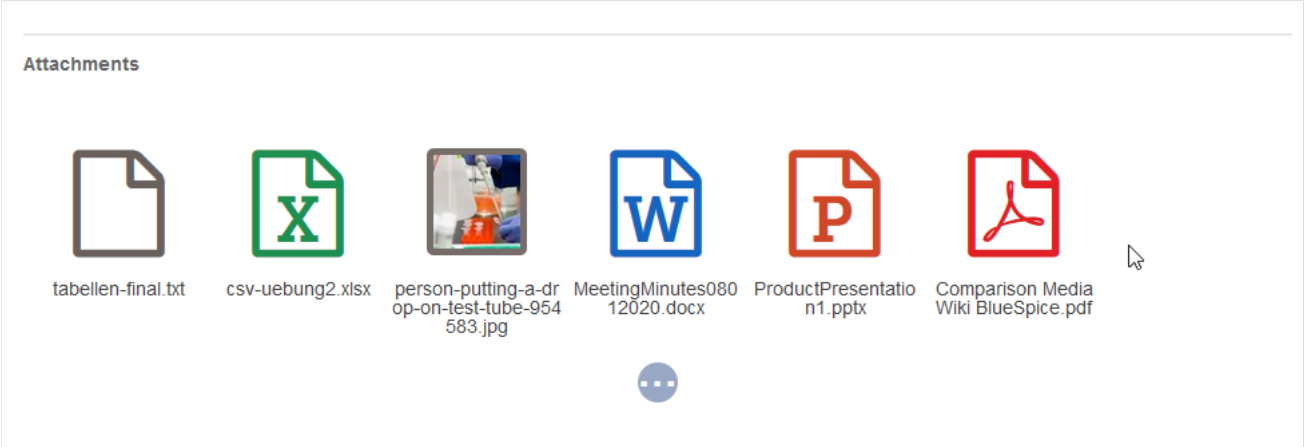
Styling

By default, the icons for the attachments are shown in grey with a small label and a large icon. By updating the file *MediaWiki:Common.css,* the styles for the attachments can be adjusted as needed. This requires administrator rights.

Add the following styles:

```
/* Attachments */
.bs-social-entity-attachment-wrapper {background: transparent;}
.bs-social-entity-content-attachments div.bs-social-entity-attachment-file a span::befo
re {line-height: inherit;font-size: 80px;  color: #695f5c;}
.bs-social-entity-content-attachments div.bs-social-entityattachment-unknown a span::be
fore  {font-family:fontawesome; content:"\f016"; color: #695f5c;}
.bs-social-entity-content-attachments div.bs-social-entityattachment-pdf a span::before
  {font-family:fontawesome; content:"\f1c1"; color: #e12025;}
.bs-social-entity-content-attachments div.bs-social-entityattachment-excel a span::befo
re {font-family:fontawesome; content:"\f1c3"; color:#1d8f50;}
.bs-social-entity-content-attachments div.bs-social-entityattachment-pptx a span::befor
e {font-family:fontawesome; content:"\f1c4"; color:#d04727;}
.bs-social-entity-content-attachments div.bs-social-entityattachment-word a span::befor
e {font-family:fontawesome; content:"\f1c2"; color:#1565c0;}
.bs-social-entity-attachment-wrapper p.attachment-name, .bs-social-entity-content-
attachments div a p, .bs-social-entity-content-attachments div span p {font-size:100% !
important; word-break:break-word;}
.bs-social-entity-content-attachments div, .bs-social-entity-content-attachments div a,
 .bs-social-entity-content-attachments div span {color: #444 !important; background-
color: #ffffff;}
.bs-social-entity-attachment-wrapper img {  border: 5px solid #766d6b;}
.bs-social-entity-content-attachments div a img, .bs-social-entity-content-attachments
div span img {height: 80px;  width: 75px;margin-top: 16px;margin-bottom: 18px; border-
radius: 5px;}
.bs-social-entity-content-attachments div a p, .bs-social-entity-content-attachments di
v span p { height:4em}
```

You will get:



Custom css styles for attachments

Related info

- Blog
- Micro-Blog
- Social comments
- User Profile
- Reference:BlueSpiceSocial

Technical Reference: BlueSpiceSocialWikiPage