

Contents

1. Manual:Extension/FlaggedRevs	2
2. Manual:Extension/FlaggedRevs/Server script	9
3. Manual:Extension/Workflows	11
4. Manual:Quality management	20
5. Manual:Semantic MediaWiki	27
6. Reference:BlueSpiceFlaggedRevsConnector	32
7. Reference:FlaggedRevs	35

Manual:Extension/FlaggedRevs

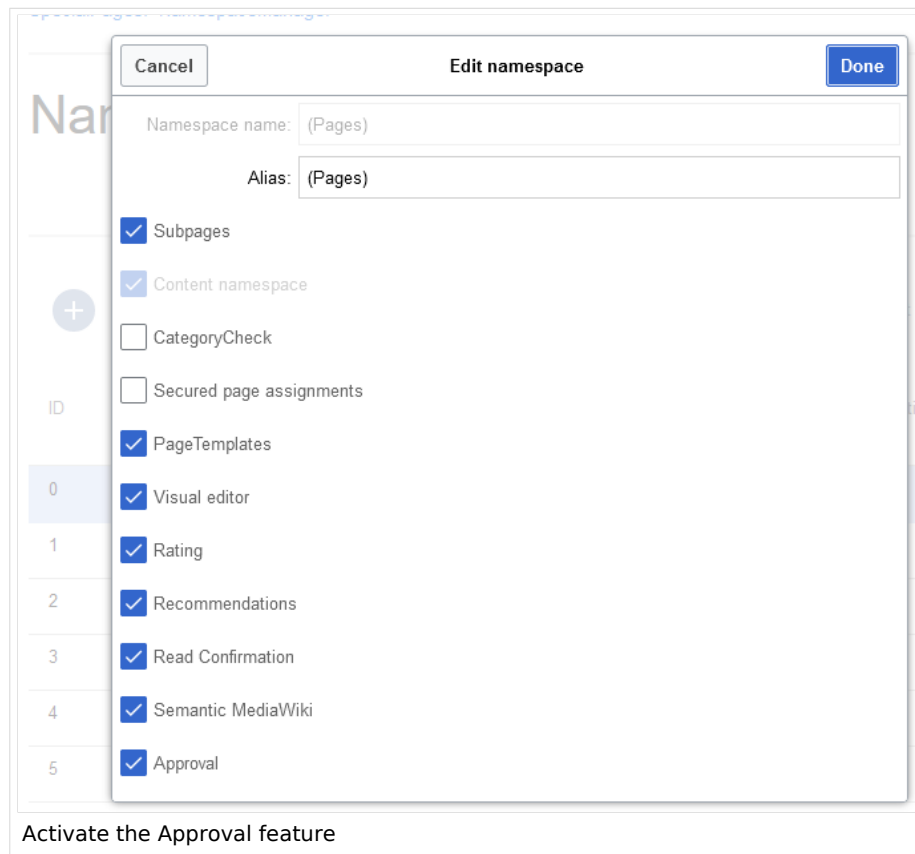
FlaggedRevs is a MediaWiki extension that activates an approval mechanism. New versions of a page are initially marked as drafts and remain so until a user with *review* rights approves the page.

Contents

1	Activating the acceptance feature	3
1.1	Existing pages	3
1.2	New pages	4
2	Accepting a page	4
3	Setting read permissions for drafts	5
4	Changes to embedded templates and files	7
5	Effects on Semantic MediaWiki properties	7
6	Quality management overview	8
7	Related info	8

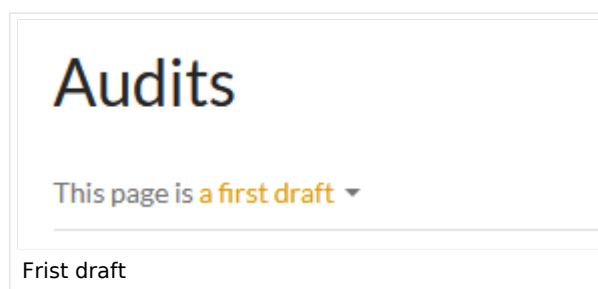
Activating the acceptance feature

Activating the acceptance feature is done on the namespace level. A user with admin permissions activates the feature in *Global actions > Management > Namespace manager* for individual namespaces. The dialog window opens after clicking the edit icon (wrench) in the *Actions* column. Select *Approval* and click *Done*.



Existing pages

After activating *Approvals* for a namespace, all pages in the namespace are set into *first draft* mode.



Each page in the namespace now displays the acceptance status in the title area.

New pages

When a new page is created and first saved, it will automatically be visible to all users. Subsequent page edits will be collected in the draft version of the page. If a page is edited and saved multiple times while in draft mode, each edit gets logged in the version history:



• (cur | prev) 14:04, 4 November 2019 WikiSysop (talk | contribs | block) . . (1,332 bytes) (+486) . . (Tag: Visual edit) [pending review]

• (cur | prev) 14:03, 4 November 2019 WikiSysop (talk | contribs | block) . . (846 bytes) (+213) . . (Tag: Visual edit) [pending review]

• (cur | prev) 13:36, 4 November 2019 WikiSysop (talk | contribs | block) m . . (633 bytes) (+38) . . (Tag: 2017 source edit)

[Reviewed: 13:57, 4 November 2019 | WikiSysop | Everything looks correct.]

• (cur | prev) 12:45, 4 November 2019 WikiSysop (talk | contribs | block) . . (595 bytes) (+595) . . (Created page with "FlaggedRevs is an extension to the MediaWiki software that allows a wiki to monitor the changes that are made to pag..." (Tag: Visual edit)

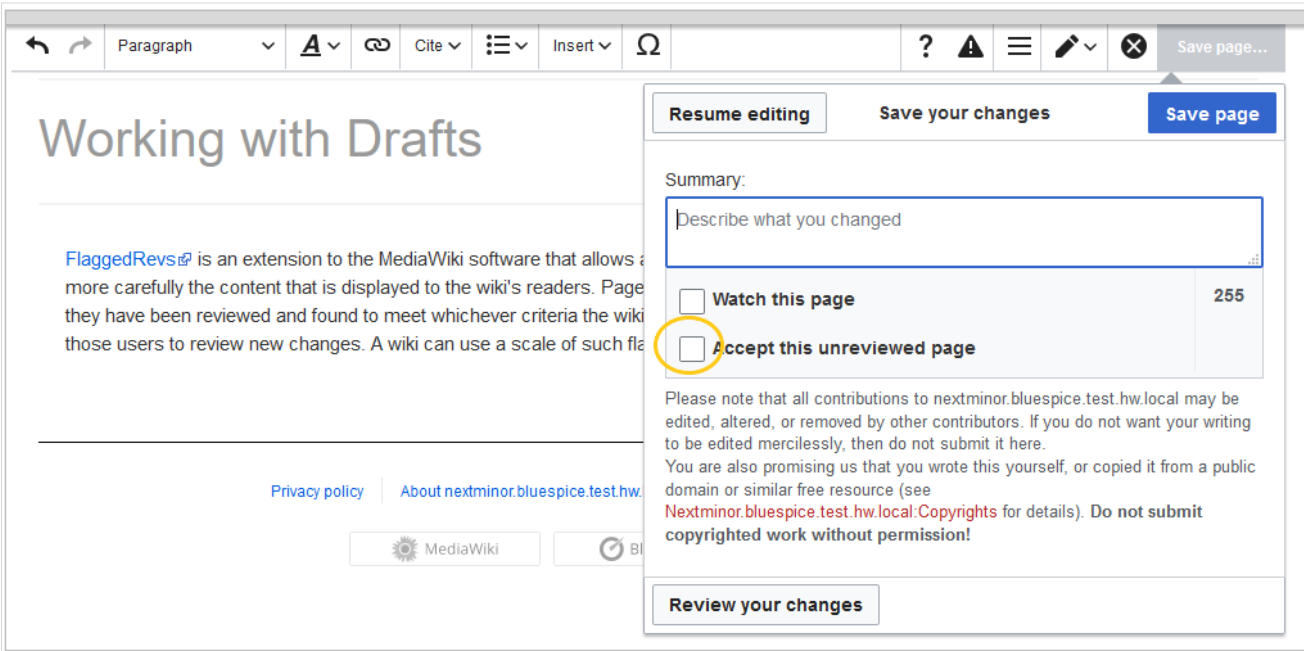
Compare selected revisions

Draft revisions

The revisions in draft status have a dotted border and are highlighted in orange. In addition, they are marked as *pending review*. These revisions can be compared before the pending changes of the page are accepted.

Accepting a page

Users with *review* rights can accept a draft page any time they save a page (*Accept this unreviewed page*):



Working with Drafts

FlaggedRevs® is an extension to the MediaWiki software that allows a wiki to monitor the changes that are made to pages more carefully. Pages that have been reviewed and found to meet whichever criteria the wiki has set are displayed to the wiki's readers. Pages that have not been reviewed are displayed to those users to review new changes. A wiki can use a scale of such flags to indicate the level of review.

Privacy policy | About nextminor.bluespice.test.hw.local

MediaWiki

Resume editing | Save your changes | Save page

Summary:

Describe what you changed

☐ Watch this page 255

☒ Accept this unreviewed page

Please note that all contributions to nextminor.bluespice.test.hw.local may be edited, altered, or removed by other contributors. If you do not want your writing to be edited mercilessly, then do not submit it here.

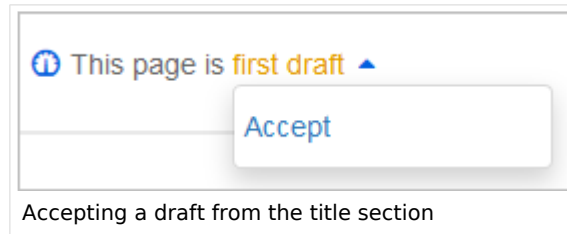
You are also promising us that you wrote this yourself, or copied it from a public domain or similar free resource (see Nextminor.bluespice.test.hw.local:Copyrights for details). Do not submit copyrighted work without permission!

Review your changes

Accepting pending changes

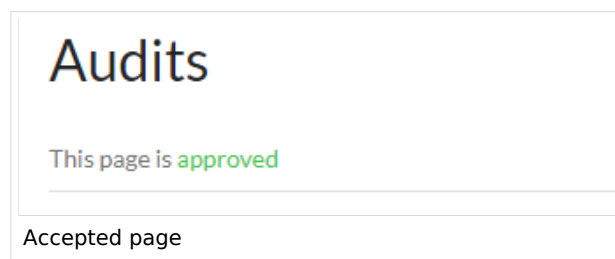
If the page is not accepted at this point, it can be accepted independently from the editing process.

A user with *review* rights can accept the page quickly with the review feature in the title section of the page. The draft message will display an arrow which includes the *accept* function:



When accepting the page, the reviewer can leave a comment. This comment will then be shown in the version history of the page.

The page status now shows the page version as accepted.



Setting read permissions for drafts

By default, all users can see the drafts of a page, even readers.

To hide drafts from readers, the group *user* has to be removed from the configuration. An administrator deletes the group user from "Show unstable pages for the following groups" in *Global actions > Management > Config manager*.

Config Manager

Feature ▼ | |

☐ Export	BlueSpiceChecklist ☒ Mark checklist changes as minor edit
☐ Skinning	
☐ System	
☐ Searching and Navigation	BlueSpiceExpiry ☒ Show watermark for expired pages
☐ Data analysis	
☐ Personalization	
☐ Book	BlueSpiceFlaggedRevsConnector ☒ Automatically set the page to stable, when the workflow was finished successfully ☒ Automatically delete the workflow, when a page was marked as stable Show unstable pages for following groups user X
☒ Quality assurance	
☐ Administration	
☐ Editor	

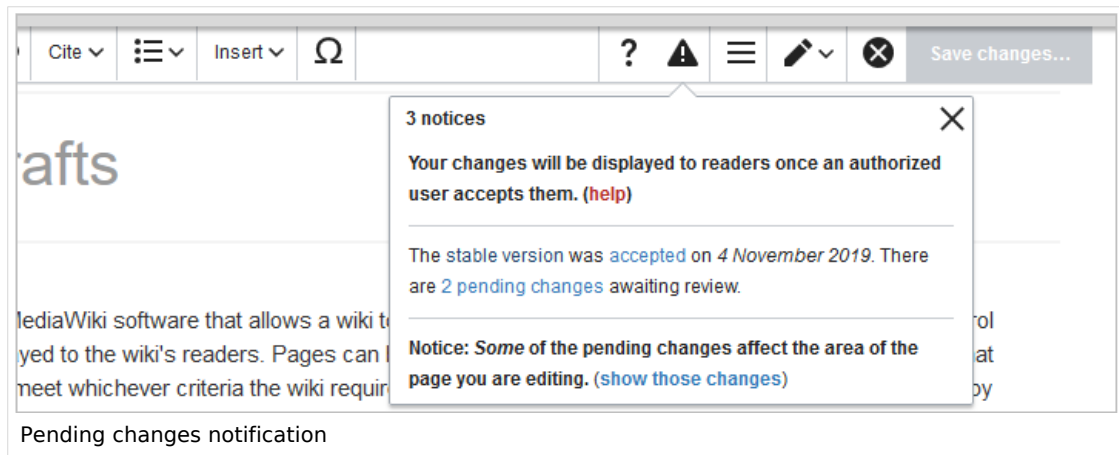
Warning! If the group *user* is removed from the *BlueSpiceFlaggedRevs* configuration, users in this group will not be able to see draft versions when they are added as reviewers in a review process.

When the group *user* is removed, only *sysop* and *reviewer* users can access drafts. If you want other groups, for example *editor*, to see draft versions, you need to add them to the configuration and save it:

The two checkboxes *Automatically set the page to stable...* (checkbox is available until version 3.1.) and *Automatically delete the workflow...* determine what happens at the end of a workflow, if there is an active [Review](#) for a page. A page can be set to accepted status by a workflow and the workflow can be deleted after the page is marked as stable.

Note: If no user groups are specified in the configuration field, not even reviewers and sysops users will see a visual status indication of a page. They will have access to drafts through the revision history of a page instead.

Anyone with edit permissions will receive a notification that there are pending changes for the page when they access the page in edit mode:



Pending changes notification

Changes to embedded templates and files

If a page uses templates or contains other files such as images and embedded PDFs and these files have been modified since the page was approved, those changes must also be approved. The wiki can be set up in two different ways to achieve this approval:

- The embedded resource is located in a namespace without approval feature:** If the embedded resource is in a namespace without its own approval function, the resource version at the time of the page approval is displayed. The draft version of the page shows the most recent version of the embedded resource. To approve the resource change, the page itself must be approved again. The approval has to be done on all pages that have this resource embedded.
[Information about automatic mass approval via server script.](#)
- The embedded resource is located in a namespace with approval feature:** The pages that include the changed resource do not give any indication that the resource has been changed. Which version of the included resource is displayed depends on the configuration of the wiki:

Setting	Description
FR_INCLUDES_STABLE (default)	Always shows the approved (stable) version of a resource, even if it was approved after the page that includes the resource was approved (this works only if the resource is located in a namespace with active approval feature).
FR_INCLUDES_CURRENT	Always shows the latest version of the included resource (this works only if the resource is located in a namespace with active approval feature).

Effects on Semantic MediaWiki properties

Wenn in einem Namensraum gleichzeitig mit der Seitenfreigabe auch [Semantic MediaWiki](#) aktiviert ist, sind je nach Freigabestatus der Seite folgende Attribute für SMW-Abfragen verfügbar:

Document state	Available properties
First draft	Current values
Approved	Values of the approved (stable) version
Draft	Values of the approved (stable) version (default setting)
Implicit draft ^[1]	Values of the approved (stable) version

The availability of the semantic properties is determined by the following configuration setting:

```
$bsgFlaggedRevsConnectorStabilizeSMWPropertyValues = true;
```

The value for this setting can be set to *false* in the *LocalSettings.php* of the server file system. If set to *false*, the property values of the draft version are used for semantic queries instead of the values for the stable version.

Quality management overview

The Special page *Special:Quality_management_overview* allows all users to view and filter a list of wiki pages and their review status. Pages for which a user does not have read permissions are not shown.

Quality management overview

Export table ▾

Page title	State ↑	Page categories
Manual:What is a wiki	Approved	
Manual:Create pages	Approved	
Manual:Subpage	Approved	
Setup:System requirements	Approved	
Setup:Release Notes	Approved	
Setup:Download	Approved	
Setup:Release History	Approved	
Setup:Portal	Approved	
Manual:Getting started - Login and configuration	Approved	Manual
Manual:Using the Version History	Approved	

↑ Sort Ascending
↓ Sort Descending
Columns >
Filters >

☒ First draft
☒ Draft
☒ Approved
☐ Not enabled
☒ Implicit draft

Special page Quality management overview

Related info

- [Quality management tools](#)

- [Reference:FlaggedRevs](#)
- [Reference:BlueSpiceFlaggedRevsConnector](#)
- [Manual:Semantic MediaWiki](#)

1. [↑](#) An approved page has an implicit draft, if there have been changes to embedded resources (e.g., a template or an image) since the last page approval.

[→](#) [Technical Reference: FlaggedRevs](#)

Manual:Extension/FlaggedRevs/Server script

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 ids	10
1.2.3	List of page names	11

Mass approval using a script

The script `extensions/BlueSpiceFlaggedRevsConnector/maintenance/BSBatchReview.php` can be used to approve all drafts in the wiki. This affects both first drafts, drafts of pages with already approved versions and drafts for embedded resources.

Options

Option	Description
<code>--username</code>	Required. The user name of the existing user to use as the "reviewer"
<code>--pageids</code>	Flat file containing page ids seperated by line break
<code>--pages</code>	Flat file containing page names seperated by line break
<code>--namespace</code>	Id of namespace to flag entirely

Note: The options `--pageids`, `--pages` and `--namespace` are mutual exclusive required. One - and only one - of them *must* be provided.

Examples

Whole namespace

```
php extensions/BlueSpiceflaggedRevsConnector/maintenance/BSBatchReview.php --username WikiSysop --namespace 3000
```

List of page ids

Such a list can be created from a database query.

File `/tmp/page-ids-to-review.txt` :

```
23
42
1337
```

Command line:

```
php extensions/BlueSpiceflaggedRevsConnector/maintenance/BSBatchReview.php --username WikiSysop --pageids /tmp/page-ids-to-review.txt
```

List of page names

Such a list can be created using a [SMW query](#). File `/tmp/page-titles-to-review.txt` :

```
Some/Page
Other_page
Help:Some_help_page
```

Command line:

```
php extensions/BlueSpiceflaggedRevsConnector/maintenance/BSBatchReview.php --username
WikiSysop --pages /tmp/page-titles-to-review.txt
```

Workflows

Contents

1	Introduction	12
2	Workflow activities	12
2.1	Single user approval	12
2.2	Expert document control	13
2.3	Group feedback	15
2.4	Feedback	16
3	Overview page	16
3.1	Tasks overview	17
4	Notifications	17
4.1	Events that trigger notifications	17
4.2	Sending out notifications	18
5	Workflow triggers	18
6	How to add a custom workflow	18
7	Permissions	19
8	Example tutorial	19

Introduction

In BlueSpice 4.1, workflows are based on [BPMN 2.0](#). Four different types of page-based workflows are already integrated. Their purpose is a page review to obtain feedback via a user vote or to trigger a page approval. In the following, these workflows are therefore called review workflows.

Types of reviews

Workflow type	Participants	Description
Single user approval	1 user	A single user is asked to vote about a page. If a user submits a positive vote, the page is automatically approved.
Expert document control	3 users	After a page has been edited by a specific user, the page is reviewed by an expert and then approved by a user who is responsible for approvals.
Group feedback	1 group	A group (which needs to exist in the group manager) is requested to leave a comment on a page.
Single user feedback	1 user	A user is asked to send a comment regarding a page.

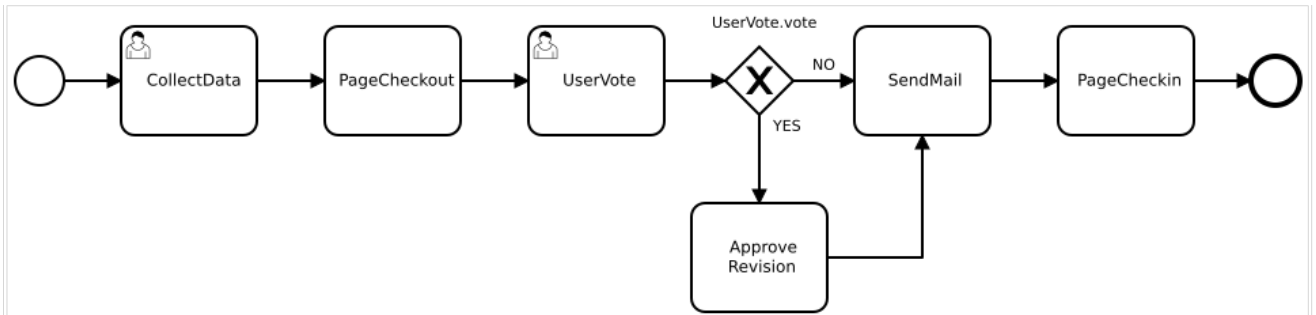
Workflow activities

All approval workflows start with a form where the necessary workflow data is entered by the workflow initiator. Each workflow results in one or more workflow activities.

Single user approval

Purpose: Approval of a draft page by a user with approval rights. This workflow only makes sense if the approval function ([FlaggedRevs](#)) is activated on a page.

Workflow instances: Only one approval workflow can run per page.



BPMN diagram of a "Single user approval" workflow

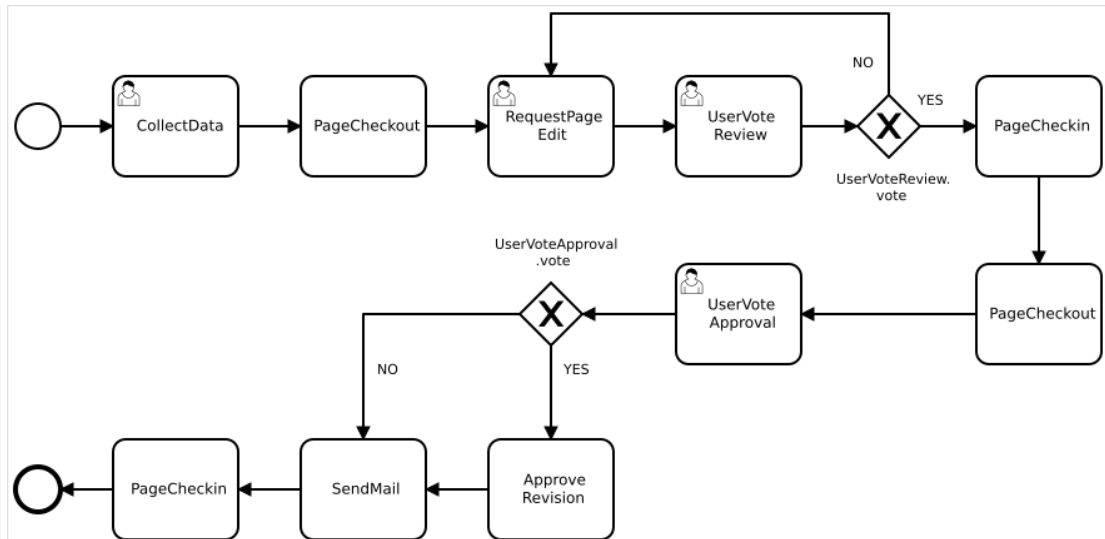
Activity	Description
CollectData	In the first workflow step, the workflow initiator enters the settings: • <i>User</i>: ser who is assigned to the task. • <i>Instructions</i>: A comment or instructions for the user to understand the task. • <i>Send report to</i>: An email report with the results will be sent to this email address when the review is finished. If a username is specified here, an email address must be stored in the user administration so that the report can be sent.
PageCheckout	The page is locked for editing. The reviewer also cannot make any edits.
UserVote	The assigned user carries out a vote and either accepts the page or rejects it. Alternatively, the task can be delegated. In the event of a rejection, the workflow skips the next step (ApprovePage). The workflow initiator gets an email about the voting result.
ApprovePage	Only if the user has submitted a positive vote (accept), the page is set to an approved state.
SendMail	An email report is sent to the report recipient who was specified in the first step.
PageCheckin	The page is unlocked.

[→ View BPMN](#)

Expert document control

Purpose: Approval of a draft page according to the "4-eyes principle".

Workflow instances: A page can only have one approval workflow at a time.



BPMN diagram of the "Expert document control" workflow

Activity	Description
CollectData	In the first workflow step, the workflow initiator enters the settings: <i>User:</i> User who is assigned to a task. Three different users have to be specified: Editor, Reviewer, Approver <i>Instructions:</i> A comment or instructions for the users to understand their tasks. <i>Send report to:</i> An email report with the results will be sent to this email address when the review is finished. If a username is specified here, an email address must be stored in the user administration so that the report can be sent.
PageCheckout	The page is locked for users who do not participate in the workflow. Only the Editor (first workflow participant) can edit the page during checkout. While the Reviewer (second participant) of the workflow is reviewing the page, the page stays checked-out to thecan edit the page can edit the page during checkout.checkout. in case the Reviewer requests more edits.
EditPage	The Editor user can edit the page and completes the task without comment.
UserVote	After the Editor step has been completed, the Reviewer user can review the page and submit a vote. Editing by the Reviewer is not possible. As an alternative, the Reviewer can delegate the task. If the vote is positive (Approve), the workflow continues. If the Reviewer rejects, the workflow goes back to the Editor. The workflow initiator gets an email about the voting result.

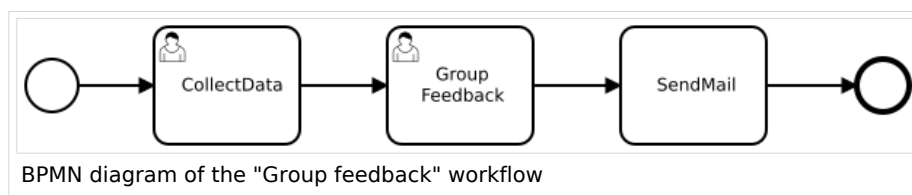
Activity	Description
PageCheckin	After the Reviewer submits a positive vote (accept), the page is checked in and the workflow continues.
PageCheckout	In this step, the page checkout locks the page for editing completely. The Approver user will not be able to change the page, but needs to approve it.
ApprovePage	The Approver can either complete or delegate the task. After the Approver (or the delegate) finishes the assigned task, the page is set from "draft" to "approved" status if the page was in draft status (only if the approver submits a positive vote). If not, this step is skipped.
SendMail	If an email or user was specified in the workflow settings, the report is now getting sent to that user.
PageCheckin	The page gets unlocked for editing.

→ [view BPMN](#)

Group feedback

Purpose: Obtaining feedback from the members of a user group. The group must exist in the [group manager](#).

Workflow instances: Several feedback workflows can run independently of one another on one page at the same time.



Activity	Description
CollectData	In the first workflow step, the workflow initiator enters the settings: • <i>Group</i>: User group who is assigned to the task. • <i>Instructions</i>: A comment or instructions for the users to understand their task. • <i>Send report to</i>: An email report with the results will be sent to this email address when the review is finished. If a username is specified here, an email address must be stored in the user administration so that the report can be sent.
	All users in the assigned group provide feedback via a comment field.

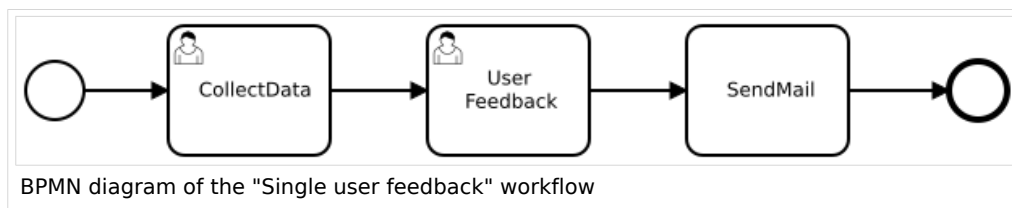
Activity	Description
GroupFeedback	This is a parallel workflow, which means that the order of the feedback does not matter.
SendMail	If an email or user was specified in the workflow settings, the report is now getting sent to that user.

→ [view BPMN](#)

Feedback

Purpose: Obtaining feedback from a single user on a page.

Workflow instances: Multiple feedback workflows can run independently of one another on one page at the same time.



Activity	Description
CollectData	In the first workflow step, the workflow initiator enters the settings: • <i>User</i>: User who is assigned to the task. • <i>Instructions</i>: A comment or instructions for the users to understand their task. • <i>Send report to</i>: An email report with the results will be sent to this email address when the review is finished. If a username is specified here, an email address must be stored in the user administration so that the report can be sent.
UserFeedback	The assigned user sends a comment.
SendMail	If an email or user was specified in the workflow settings, the report is now getting sent to that user.

→ [view BPMN](#)

Overview page

All workflows in the wiki are listed on the page *Special:Workflows overview*. A view for all *active* workflows and a view for *all* workflows can be selected.

Workflows related to this page						
All Active						
Type	Subject page	Current tasks	State	Started on	Last activity on	
Group Feedback	Main Page	-	Aborted	14:47, 29 November 2021	14:49, 29 November 2021	see details
Single user feedback	Main Page	-	Finished	15:43, 3 November 2021	15:43, 3 November 2021	see details
Expert document control	Main Page	UserVoteReview	Active	12:20, 21 December 2021	12:20, 21 December 2021	see details
Single user feedback	Main Page	-	Finished	16:34, 20 December 2021	17:13, 20 December 2021	see details
< 1 of 3 >						25 rows

Tasks overview

Users are informed about new and pending tasks in their notifications. They can view assigned workflows on their [My tasks](#) page.

Notifications

Events that trigger notifications

There are two types of events that trigger notification

- **generic:** notifications happen for every workflow/activity type
- **activity-specific:** activities themselves can decide to send additional notifications

Triggering event	Recipients	Generic	Notes
Task started (task assigned)	All assigned users	Yes	Only triggered for type <i>UserInteractiveActivity</i> , i.e., only for activities that have users assigned.
UserVote	Initiator	Yes	The workflow initiator gets an email about the voting result.
Workflow aborted (manual or automatic)	Initiator and all users that were assigned to the current task at time of aborting (not users who were assigned on previous tasks)	Yes	

Triggering event	Recipients	Generic	Notes
Workflow ended (only when naturally ended, not when aborted)	Initiator	Yes	
DueDateClose (2 days before Workflow will expire)	Initiator and all currently assigned users	Yes	
Workflow expired	Initiator and all currently assigned users	Yes	Expiration is just a type of workflow abort, so the same notification as for abort will be sent with the reason explaining that the workflow expired.
Task delegated	User to whom the task was delegated	No	Specific to <i>UserVote</i> activity. After delegation, the newly assigned user will be considered assigned and will receive all further notifications that go out to assigned users.

Sending out notifications

- Users can choose whether to subscribe to e-mail notifications in their preferences. All users are force-subscribed to web notifications.
- Web notifications are sent out immediately after triggering, while email notifications will be sent async, on `runJobs.php` execution. This applies to notifications in general, not only to workflows

Workflow triggers

Workflows can either be started manually on each wiki page or started only under certain conditions using individual [workflow triggers](#). Triggers also allow to define in which namespaces both manual and automatic workflows are available.

How to add a custom workflow

Users can upload an xml-file of a BPMN diagram with custom activities to the wiki. Currently, the following predefined activities exist:

Extension: Workflows

- [CustomForm](#)
- [UserVote](#)
- [GroupVote](#)
- [UserFeedback](#)
- [GroupFeedback](#)
- [SendMail](#)
- [EditRequest](#)

Extension: PageCheckout

- [PageCheckOut](#)
- [PageCheckIn](#)

Extension: BlueSpiceFlaggedRevsConnector

- [ApprovePage](#)

Example of a customized workflow (coming soon)

Permissions

The following permissions are used by this extension:

Permission	Included in role	Description
workflows-view	reader	• allows viewing workflow elements, including listing of workflows (e.g., viewing all running workflows on a page) • user can view the page <i>Special:Workflows_overview</i>
workflows-execute	editor, reviewer, admin	• allows starting a workflow and executing a task
workflows-admin	admin	• allows aborting, restoring and administering all workflows • user fcdan view and edit the page <i>MediaWiki:WorkflowTriggers</i>

Example tutorial

You can follow our [tutorial for creating a custom workflow](#) that allows users to classify a document and notify a user about the classification.

[Technical Reference: Workflows](#)

Quality management

Um die Qualität einer Wikiseite zu gewährleisten, gibt es zahlreiche Werkzeuge zum Qualitätsmanagement.

Contents

1 Dokumentenlenkung	21
2 Quality tools	21
3 Approval	21
4 Workflows	22
5 Aktualität	22
6 Erinnerung	22
7 Zuweisung	23
8 Seiteninformationen	23
9 Berechtigungen	24
10 Related info	27

Dokumentenlenkung

In BlueSpice können viele Benutzer Informationen erstellen. Spezialisten können den Inhalt dann überprüfen. Dies stellt sicher, dass die Qualität der Inhalte den Standards Ihrer Organisation entspricht. Dabei ist zu beachten, dass rigide Qualitätskontrollen und -prozesse schnell die Autonomie der Mitarbeiter einschränken. Trotz der Möglichkeit zur Qualitätskontrolle sollten Sie daher immer sicherstellen, dass Sie keine unnötigen Hürden einbauen, die die Akzeptanz des Wikis beeinträchtigen.

Folgende Bereiche sollten für die Qualitätssicherung im Wiki im Auge behalten werden:

1. Document quality ([Quality tools](#), Templates)
2. Content organization (Structure, Portal pages, Queries)
3. Wiki maintenance (Special pages, Processes, Watchlist, Notifications)

Quality tools

BlueSpice free

- Page assignments (5)
- Page information (6)

BlueSpice pro

- Approvals (1)
- Workflow (2)
- Expiry (3)
- Reminder (4)
- Page assignments (5)
- Page information (6)

Template:Message box.

Approval

Wenn die Seiten in einem Namesraum eine veröffentlichte und eine Entwurfsversion haben sollen, muss die Einstellung [Freigabe](#) für diesen Namesraum aktiviert werden. Alle Änderungen an einer Seite werden in einer Entwurfsversion erfasst, bis ein Benutzer mit Freigaberechten die Seite akzeptiert.

In der Konfigurationsverwaltung gibt es eine globale Option, mit der festgelegt werden kann, welche Benutzergruppen unveröffentlichte Inhalt lesen dürfen. Standardmäßig können alle angemeldeten Benutzer Entwürfe einsehen.

Eigenschaften

- **Revisionsbasiert:** Unterscheidung zwischen Erstentwurf, Entwurf, und freigegeben
- **Externe Dateien:** Eingebettete Medien und Vorlagen werden "eingefroren"
- **Automatische Freigabe** als Ergebnis einer Begutachtung möglich
- **Leserechte** für Entwürfe können (mit Vorsicht) konfiguriert werden

→ [Mehr Info zu Freigabe](#)

Workflows

Wenn ein dokumentierter Überprüfungsprozess erwünscht ist, können Benutzer einen Workflow starten. Es gibt 4 mitgelieferte Workflows:

- Freigabe
- Qualitätsgesicherte Begutachtung
- Gruppen-Feedback
- Feedback

Eigenschaften

- **Meistens nur eine aktive Begutachtung** pro Seite möglich
- **Zustimmung** oder **Ablehnung** der Teilnehmer bei Freigabe-Workflows
- **Checkout:** Seite ist in den Freigabeworkflows zur Bearbeitung gesperrt
- **Spezialseite** Workflows Übersicht
- **BPMN**-basiert

→ [Mehr Info zu Workflows](#)

Aktualität

Wenn eine Seite Inhalte mit einem bekannten Lebenszyklus enthält, kann festgelegt werden, dass die Seite an einem bestimmten Datum abläuft. Am Ablaufdatum wird der Seite ein Wasserzeichen "Veraltet" hinzugefügt, wenn die entsprechende Option im [ConfigManager](#) eingestellt ist. Darüber hinaus kann eine Erinnerung festgelegt werden, damit der Benutzer eine Benachrichtigung erhält, wenn die Seite abläuft. Der Ablauf kann jederzeit von der Seite gelöscht werden.

Eigenschaften

- **Wasserzeichen** im Seitenhintergrund und im PDF (abhängig von globaler Konfiguration)
- **Übersichtsliste:** *Spezial:Veraltete_Seiten*
- **Erinnerung** beim Erstellen aktivierbar
- **Dauer:** Veraltung ist jederzeit löscher

→ [Mehr Info zu Aktualität](#)

Erinnerung

Ein Benutzer kann eine Erinnerung für eine Seite erstellen, indem er ein Erinnerungsdatum auswählt und einen optionalen Kommentar hinzufügt. Erinnerungen können von Administratoren auch für andere Benutzer erstellt werden und sind eine schnelle und informelle Möglichkeit, Fälligkeitstermine für seitenbezogene Aufgaben festzulegen.

Eigenschaften

- **Benachrichtigung** über eigene Erinnerungen erhält eine Benachrichtigung (in Benutzereinstellungen deaktivierbar)
- **Wiederkehrende Erinnerungen** möglich
- **Erinnerungsliste:** Spezialseite für Administratoren (unter Globale Aktionen)
- **Schnellzugriff** über das persönliche Benutzermenü

➞ [Mehr Info zu Erinnerung](#)

Zuweisung

Seitenzuweisungen bieten die Möglichkeit, einem oder mehreren Benutzern oder Benutzergruppen die redaktionelle Verantwortung für eine Seite zu übertragen. Es ist auch möglich, einen Namensraum so einzurichten, dass zugewiesene Benutzer eine Lesebestätigung für jede größere Änderung an einer Seite abgeben müssen. Da Zuweisungen im Wiki als Seitenattribut zur Verfügung stehen, können sie in **semantischen Abfragen** verwendet werden.

Eigenschaften

- **Benachrichtigung** bei Änderung der Seitenzuweisung (in Benutzereinstellungen deaktivierbar)
- **Transparenz:** Zugewiesene Benutzer sind in den Seiteninformationen aufgeführt
- **Zuweisungsverwaltung:** Spezialseite für Administratoren
- **Schnellzugriff** auf eigene Zuweisungen über das persönliche Benutzermenü
- **Lesebestätigung möglich:** Zugewiesene Benutzer müssen dann das Lesen von Änderungen an der Seite bestätigen (Namensraum-spezifisch)

➞ [Mehr Info zu Zuweisungen](#)

Seiteninformationen

Die Seiteninformationen geben Ihnen einige interessante Statistiken über die Seite. Anhand dieser Informationen können Sie entscheiden, welche Maßnahmen zur Qualitätssicherung der Seite erforderlich sind. Informationen, die auf dieser Seite aufgelistet sind:

- Anzahl der Beobachter der Seite
- Alle Bearbeiter, letzte Bearbeitung
- Eigenschaften: Eingebundene Vorlagen, eingebundene Bilder, interne und externe Links, Kategorien
- Aktualität der Seite
- Autoren der Seite
- Leser der Seite (nur für Administratoren sichtbar)
- Erinnerungen (alle Erinnerungen nur für Administratoren sichtbar, ansonsten eigene Erinnerungen)

Berechtigungen

Diese Berechtigungstabelle zeigt, welche Rollen das erforderliche Recht haben, verschiedene Qualitätssicherungsaktionen durchzuführen. Der Name des Rechts, das eine bestimmte Berechtigung erteilt, wird in Klammern angezeigt:

QM permissions	
user (read)	
editor	
reviewer	
admin	
intenance-admin	
bot	
Can create and manage assignments for all users (pageassignments)	
x	
x	
x	
Can be assigned to a page (pageassigneable)	
x	

Can create and manage reminders for all users. <i>(remindereditall)</i>
x
x
Can create and manage their own reminders <i>(read)</i>
x
Can create and manage expiry of pages. <i>(expire article)</i>
x
x
x
Can execute workflows <i>(workflows- execute)</i>
x

x
x
Can view workflow elements <i>(workflows-view)</i>
x
Can manage workflows <i>(workflows-admin)</i>
x
Can release <i>(review)</i> a page
x
x
x
Own changes are automatically released <i>(autoreview)</i>
x
x

Rolled-
back
changes
are
**automatically
released**
(autoreviewrestore)

Related info

- [Namespace manager](#)
- [Workflows](#)

Semantic MediaWiki Basics

Contents

1 Scope and terminology	28
2 Define properties	31
3 Create a template	31
4 Create a form	31
5 Assign a category	32
6 Example	32

Scope and terminology

Semantic MediaWiki (SMW) is a MediaWiki extension that enables you to define and query data on a wiki page.

The following steps are usually taken to work with data-based content in your wiki :

1. **Define properties:** Properties determine which values can be queried in the wiki.
2. **Create a template:** This creates the output format of the data set on the wiki pages.
3. **Create a form:** Using this form, Wiki users can comfortably enter their data.
4. **Assign a category:** Each page that contains a semantic query is usually tagged with a category that was created for this purpose. This, in addition to the attributes, provides an important means of querying and filtering the pages. It also creates a form editing mode for the wiki pages that contain this category.

Properties



List of properties

1. Has First contact +  of type [Date](#) (0 uses) 
2. Has contact person +  of type [Text](#) (0 uses) 
3. Has location +  of type [Text](#) (0 uses) 

Semantic properties

Template



Location	Regensburg
Contact	Jane Doe
First contact	2020/12/21

Hallo Welt! was founded to make the technology and concept

Since 2007 Hallo Welt! builds collaborative platforms of knowl
online documentation and quality management.

Semantic template

Form



Location:

Contact person:

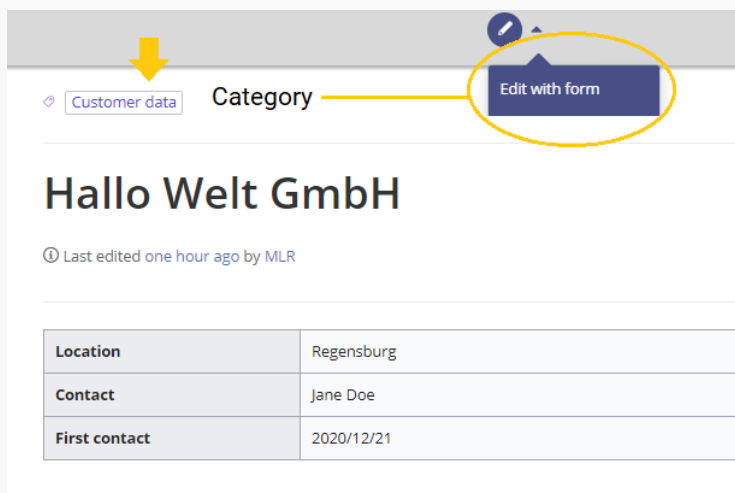
First contact: 

Free text:

Hallo Welt! was founded to make the technology and concept c

Since 2007 Hallo Welt! builds collaborative platforms of knowle
management, online documentation and quality management.

Data entry form



The screenshot shows the top of a MediaWiki page. A yellow arrow points to the 'Customer data' category label. Another yellow arrow points to the 'Edit with form' button, which is circled in yellow. Below the category, the title 'Hallo Welt GmbH' is displayed, followed by a note 'Last edited one hour ago by MLR'. A table below contains the data entry form fields.

Location	Regensburg
Contact	Jane Doe
First contact	2020/12/21

Category to organize data

Define properties

To make information available as data on a wiki page, we create so-called **properties**. For example, if we want to semantically record customer data, we create a wiki page for each customer. We collect customer data such as location on every customer page. To do this, we create a property that records the location of the customer.

```
[[Has location::Regensburg]]
```

Afterwards I can list all customers at a certain location or all customers with their associated locations in the wiki.

Create a template

Since we often want to display several such properties and their values on one page at the same time, we format a special area that clearly shows the collected page attributes. This requires a (data) template, which can then be used on any wiki page.

Location	Regensburg
Contact	Jane Doe
First contact	2020/12/21

Data template

Create a form

To support our users with data entry tasks, we create an input form. The input form is shown when users switch to the edit mode of a wiki page that has the semantic template included.

Location:	Regensburg
Contact person:	Jane Doe
First contact:	21/12/2020 

Data entry form

Assign a category

Wiki pages that use semantic templates are tagged with a category. This category enables the assignment of the input form to the wiki page. The category is also used to query the data collected via the template and display it anywhere in the wiki.

Example

On the following pages, we will create an example that collects and displays customer data it in the wiki.

Example: Collect customer data

Reference:BlueSpiceFlaggedRevsConnector

Extension: BlueSpiceFlaggedRevsConnector

→ [all extensions](#)

Overview			
Description:	Adds support for FlaggedRevs to a range of other extensions.		
State:	stable	Dependency:	BlueSpice
Developer:	HalloWelt	License:	GPL-3.0-only
Type:	BlueSpice	Category:	Infrastructure
Edition:	BlueSpice pro, BlueSpice Farm, BlueSpice Cloud	Version:	4.1- 4.2.x

Technical Information

This information applies to BlueSpice 4. Technical details for BlueSpice Cloud can differ in some cases.

Requirements

MediaWiki: 1.35.0
BlueSpiceFoundation: 4.0

- FlaggedRevs: *

Integrates into

- BlueSpiceExtendedStatistics
- BlueSpicePrivacy
- BlueSpiceReview
- BlueSpiceSMWConnector
- Workflows

Special pages

- QMOverview

Permissions

Name	Description	Role
autoreview	right-autoreview	bot, maintenanceadmin
autoreviewrestore	right-autoreviewrestore	bot, maintenanceadmin
movestable	right-movestable	admin, editor, maintenanceadmin, reviewer, structuremanager
review	right-review	admin, maintenanceadmin, reviewer
stablesettings	right-stablesettings	admin, maintenanceadmin, reader
unreviewedpages	right-unreviewedpages	admin, editor, maintenanceadmin, reviewer
validate	right-validate	admin, maintenanceadmin, reviewer

Configuration

Name	Value
FlaggedRevsConnectorBookshelfShowNoFR	true

Name	Value
FlaggedRevsConnectorBookshelfShowNoStable	true
FlaggedRevsConnectorBookshelfShowStable	true
FlaggedRevsConnectorDraftGroups	array (0 => 'user',)
FlaggedRevsConnectorFlaggedPagesStore	'BlueSpice\\FlaggedRevsConnector\\Data\\FlaggedPa
FlaggedRevsConnectorFlaggedRevsHandleIncludes	2
FlaggedRevsConnectorIndexStableOnly	false
FlaggedRevsConnectorPageApprovedReminderDelay	168
FlaggedRevsConnectorStabilizeSMWPropertyValues	true
FlaggedRevsConnectorUEModulePDFShowFRTag	true
FlaggedRevsConnectorautoDeleteWorkflow	true

API Modules

- [bs-flaggedpages-store](#)

Hooks

- [ArticleContentOnDiff](#)
- [ArticleViewHeader](#)
- [BSExtendedSearchRepoFileGetRepoFile](#)
- [BSExtendedSearchWikipageFetchRevision](#)
- [BeforePageDisplay](#)
- [BeforeParserFetchTemplateAndtitle](#)
- [BlueSpiceEchoConnectorUserLocatorValidUsers](#)
- [ChameleonSkinTemplateOutputPageBeforeExec](#)
- [EditPageGetCheckboxesDefinition](#)
- [FlaggedRevsRevisionReviewFormAfterDoSubmit](#)
- [ImagePageFileHistoryLine](#)
- [ImgAuthBeforeStream](#)
- [LoadExtensionSchemaUpdates](#)
- [MWStakeCommonUIRegisterSkinSlotComponents](#)
- [MessageCache::get](#)
- [OutputPageBeforeHTML](#)
- [PageHistoryLineEnding](#)
- [PageSaveComplete](#)
- [SMW::RevisionGuard::ChangeRevision](#)
- [SMW::RevisionGuard::ChangeRevisionID](#)
- [SMW::RevisionGuard::IsApprovedRevision](#)

- [SkinTemplateNavigation::Universal](#)
- [SpecialPage_initList](#)
- [checkPagelsReviewable](#)

Accessibility

Test status:	2-testing complete
Last test date:	2022-08-09
Extension type:	backend

Reference:FlaggedRevs

Extension: FlaggedRevs

→ [all extensions](#)

Overview			
Description:	Allows for article content management by editors and reviewers.		
State:	stable	Dependency:	MediaWiki
Developer:	Aaron Schulz, Joerg Baach	License:	-
Type:	MediaWiki	Category:	Quality Assurance
Edition:	BlueSpice pro, BlueSpice Farm, BlueSpice Cloud	Version:	4.1- 4.2.x
? View help page			

Features

FlaggedRevs is a MediaWiki extension that activates an approval mechanism.

New versions of an article are initially marked as drafts and remain so until a user with revision rights checks and approves the page.

The MediaWiki extension is customized in BlueSpice. It uses the [FlaggedRevsConnector](#) to work alongside [workflows](#), realising the complete functionality of an editorial process with a final approval step.

Warning! The setting `FR_INCLUDES_FREEZE` for handling the state of included files of an approved wikipage has [no long term support by](#) and should not be introduced in a new wiki for that reason.