



- User Guide -

Last modified 2012-04-19

Content

| | | |
|-----|---|---|
| 1 | Preconditions..... | 2 |
| 2 | Concepts..... | 3 |
| 2.1 | Review..... | 3 |
| 2.2 | Comment..... | 3 |
| 3 | Views of the AgileReview perspective..... | 4 |
| 3.1 | Review Explorer..... | 4 |
| 3.2 | Comments Summary View..... | 4 |
| 3.3 | Detail View..... | 5 |
| 3.4 | Additional Views..... | 5 |
| 4 | Preferences..... | 6 |
| 5 | The review process..... | 7 |
| 5.1 | Creating a review..... | 7 |
| 5.2 | Creating a comment..... | 7 |
| 5.3 | Sharing comments and reviews..... | 8 |
| 6 | Advanced Functionality..... | 9 |
| 6.1 | Keyboard Shortcuts..... | 9 |
| 6.2 | Export Review Data..... | 9 |
| 6.3 | Project Cleanup..... | 9 |

1 Preconditions

Conducting reviews in cooperation with other developers using AgileReview requires the possibility to **exchange and merge text files** via e.g. a version control system like SVN (see section 5.3). Furthermore, the **review process itself is not managed by AgileReview**. Instead, AgileReview provides the possibility to be integrated into an existing review process (see section 2.1). Therefore, the scope of a review (in terms of e.g. classes or methods) has to be defined separately.

2 Concepts

2.1 Review

A review is a set of – semantically or temporally related – comments. Furthermore, a review includes information like person in charge or a description about itself. In order to be integrated into an existing review process, the data of a review can contain a reference in form of an URL (e.g. an URL of a Bugzilla ticket).

In addition, a review can be “active“ or “inactive“. It holds that in a single eclipse instance at each point in time only one review can be “active“. If a certain review is “active“, all comments created by the user are assigned to it.

In order to limit the number of comments that are displayed, a review can be closed (which is similar to the “close project“ mechanism of Eclipse). Comments of reviews that are closed are neither displayed nor loaded which increases performance.

2.2 Comment

A comment is a set of information that is related to a file. A file can be related to more than one comment since the comments can be unambiguously identified by the triple (review ID, author, comment ID).

Comments can be “global“ (only related to a file) or “bound“ (related to a section of code within a file). “Bound“ comments can only be added to structured file. A structured file is a code or data file with content that defines multi-line comments¹.

2.3 Source Folder

The source folder is a special project that stores reviews and comments. As the data is not stored within reviewed projects, comments of a review can be added to multiple projects.

Since AgileReview v0.7.0 it is possible to have multiple source folders. As with reviews, in a single eclipse instance only one folder can be “active“ at a time. This folder is marked by a green hook. The user can activate a certain source folder using the preferences page. If a certain source folder is “active“, all comments and reviews created by the user are stored in the respective source folder. When the AgileReview plug-in is activated for the first time² a wizards which lets you e.g. create (fig. 2) or import an AgileReview Source Folder will be presented to you (see fig. 1).

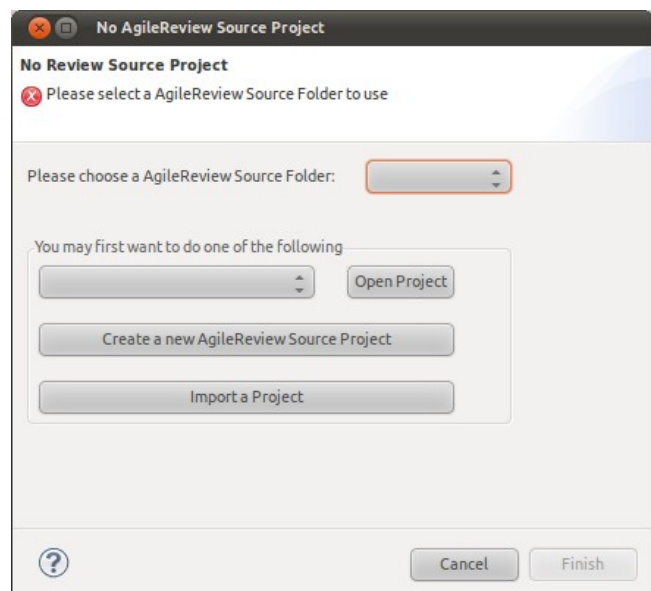


Figure 1: Wizard to select an AgileReview Source Folder

¹ e.g. Java ▶ “/*” and “*/”; XML ▶ “<!--” and “-->”

² e.g. by switching to the AgileReview perspective, refactoring, or opening an editor

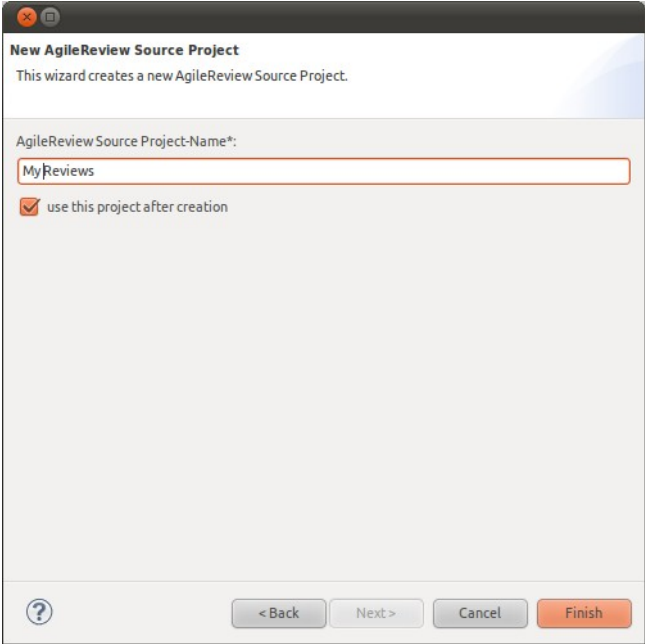


Figure 2: Wizard to create a new AgileReview Source Folder

3 Views of the AgileReview perspective

3.1 Review Explorer

Basically, the Review Explorer allows the **management of reviews** as well as files that comments were added to.

Reviews are displayed as top level elements of a hierarchical structure. In addition, the structure contains projects, folders and at the bottom level source files that are encompassed by the review (i.e. that have been commented).

The management functions of the Review explorer include the possibilities to create, delete, open, close, activate and export reviews, expand and collapse sub-nodes of a selected node and change the source folder. These functions can either be invoked via the toolbar icons of the view or by using the context menu of the elements displayed in the hierarchical structure. In contrast to the toolbar, the context menu provides the possibility of selectively removing projects, folders or files from a review. Thereby, comments that have been added to the project, folder or file (and in case of projects or folders the contained sub-hierarchy) are deleted.

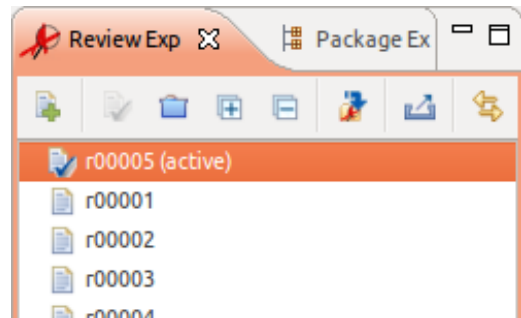


Figure 3: The Review Explorer

3.2 Comments Summary View

The Comments Summary View shows a list of comments along with several attributes of them. Usually the Comments Summary View shows all comments of currently open Reviews. To avoid information overload, the list of comments can be filtered by a given term via the search field. The filter can be restricted to a given column of the table. In addition, the list of comments can be filtered by the “Link Review Explorer” button of the Review Explorer. After applying this filter, the list of comments reflects the selections of the Review Explorer.

| ReviewN | CommentID | Author | Recipient | Status | Priority | Date created | Date modified | Replies | Location |
|---------|-----------|--------|-----------|--------|----------|----------------------|----------------------|---------|--|
| r73+r87 | c0 | Malte | anyone | closed | low | 18.11.2011, 11:17:45 | 18.11.2011, 16:38:30 | 1 | AgileReview_next_Release/src/de/tukl/cs/softech/agilereview/pref |
| r73+r87 | c1 | Malte | anyone | closed | low | 18.11.2011, 11:20:20 | 18.11.2011, 16:44:20 | 1 | AgileReview_next_Release/src/de/tukl/cs/softech/agilereview/pref |
| r73+r87 | c2 | Malte | anyone | closed | low | 18.11.2011, 11:25:16 | 18.11.2011, 16:44:11 | 1 | AgileReview_next_Release/src/de/tukl/cs/softech/agilereview/pref |
| r73+r87 | c3 | Malte | Peter | closed | low | 18.11.2011, 12:22:54 | 22.11.2011, 16:51:20 | 2 | AgileReview_next_Release/src/de/tukl/cs/softech/agilereview/view |
| r73+r87 | c4 | Malte | Peter | closed | low | 18.11.2011, 12:26:25 | 19.11.2011, 13:51:55 | 1 | AgileReview_next_Release/src/de/tukl/cs/softech/agilereview/view |
| r73+r87 | c5 | Malte | Thilo | closed | low | 18.11.2011, 12:40:14 | 18.11.2011, 16:49:50 | 1 | AgileReview_next_Release/src/de/tukl/cs/softech/agilereview/pref |
| r73+r87 | c0 | Thilo | | closed | low | 15.11.2011, 18:46:20 | 18.11.2011, 11:30:20 | 0 | AgileReview_next_Release/src/de/tukl/cs/softech/agilereview/pref |
| r73+r87 | c1 | Peter | | closed | low | 17.11.2011, 14:08:55 | 22.11.2011, 16:49:50 | 3 | AgileReview_next_Release/src/de/tukl/cs/softech/agilereview/ann |
| r73+r87 | c2 | Peter | | closed | low | 17.11.2011, 14:09:05 | 22.11.2011, 16:50:40 | 2 | AgileReview_next_Release/src/de/tukl/cs/softech/agilereview/ann |
| r73+r87 | c3 | Peter | | closed | low | 17.11.2011, 14:09:20 | 18.11.2011, 12:20:40 | 0 | AgileReview_next_Release/src/de/tukl/cs/softech/agilereview/ann |

Figure 4: The Comments Summary View

Comments can be added to selected ranges of code (“bound”) or files (“global”) via the buttons of the toolbar. Similarly, comments that have been selected by keyboard or mouse can be removed by the “Delete Comment(s)” button which is also part of the toolbar.

3.3 Detail View

The Detail View presents

- a) either the Details (ID, reference, responsibility,...) of a review, if the last selection that was done is a review³,
- b) or the Details (author, status, priority,...) of a comment, if the last selection that was done is a comment⁴.

The Detail View provides functionality for reverting recent changes in the fields of comments and reviews, or deleting the displayed element via its toolbar. Furthermore, comments can be relocated and replies can be added to them, which allows a rudimentary form of discussion.

With the release of AgileReview v0.7.0, different background-colors were introduced in order to easily distinguish the Review and Comment Details. While the Details View is colored light green when showing a Review, the Comment Details are colored with the users preferred comment color.



Figure 5: The Detail View

³ That means a “review” node of the Review Explorer was selected most recently.

⁴ That means an entry of the Comments Summary View was selected most recently.

4 Preferences

Before starting the review process it is recommended to adjust the preferences. This can be done via selecting “Window►Preferences” from the main menu. Next, select the “AgileReview” category on the left. On the right side, a set of settings of the AgileReview plug-in will appear that contains information such as author name and the color commented lines will be highlighted with (see fig. 6).

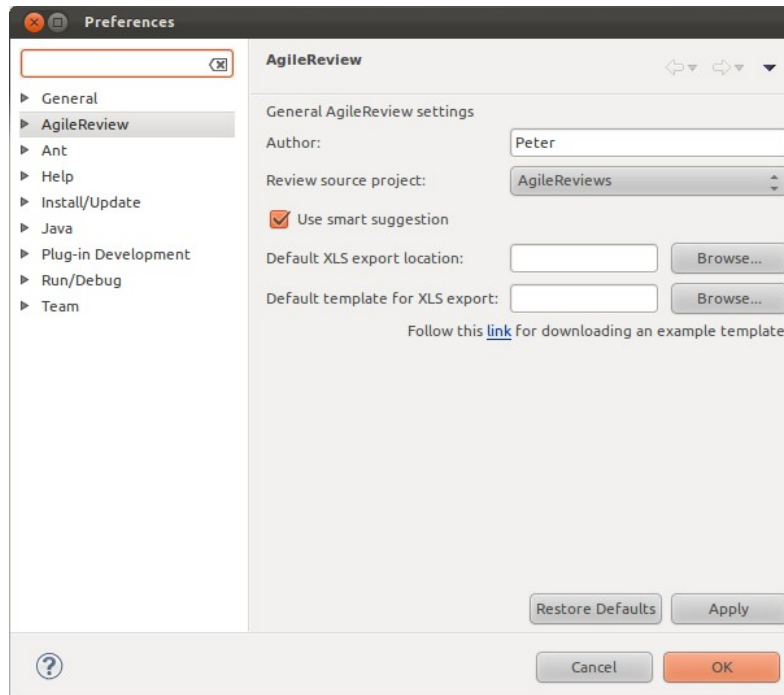


Figure 6: Preferences page of the AgileReview plug-in

The subsection “Language Support” of the “AgileReview” preferences category lets you easily add support for additional programming languages by specifying file ending, and begin- and end-tag of multi-line comments of the programming language (see fig. 7).

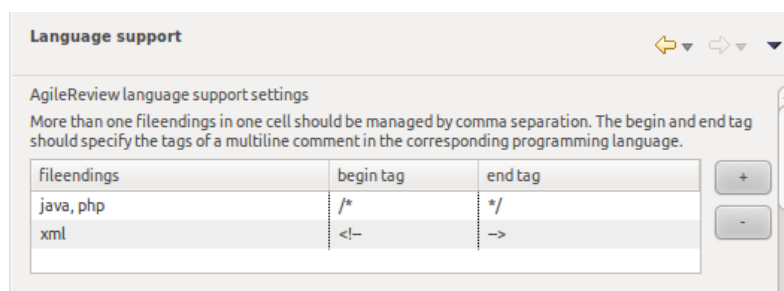


Figure 7: Adaptable list of supported programming languages

Multi-color support of comments can be en-/disabled via the preferences “Comment Colors” subsection. Furthermore, custom colors for up to 10 authors (including the user of Eclipse) can be chosen. If a review is conducted by more than 10 authors the comments of authors that got no color assigned by AgileReview are highlighted in the default color.

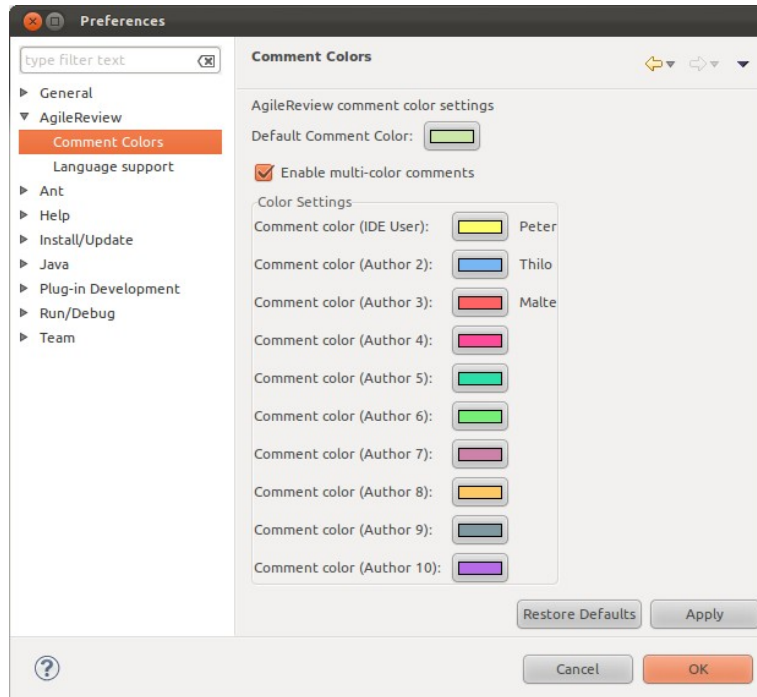


Figure 8: Multi-color support of comments

5 The review process

5.1 Creating a review

As mentioned in section 1 a precondition for using AgileReview is an existing process for assigning code reviews in terms of

1. pieces of code (classes, methods, ...) and
2. employees that conduct the review.

After this initial assignment the main part of the review process can be performed. This requires certain steps:

1. Switch to the AgileReview perspective
2. Create a new review using the Review Explorer by either
 - a) selecting the “Create new review” button from the toolbar, or
 - b) selecting “Create new review” from the context menu, or
 - c) using the “New” wizard (“CTRL+N” or “File►New►Other”), where you can select “Review” from the “AgileReview” category.
3. Enter the required data in the following dialog (see fig. 9).
4. The new review is set to the “active” state. All comments that are added following are assigned to this review, until another review is set to the “active” state.
5. Details of the review will be displayed in the Detail View. In order to later on change the details of a review select it from the Review Explorer and manipulate the data via the Detail View.

Figure 9: "New Review" wizard dialog

5.2 Creating a comment

After creating a review you can proceed with the process by adding comments to your documents. It is assumed that the AgileReview perspective is selected. Adding comments can be achieved in multiple ways.

5.2.1 Adding a bound comment

1. Open a structured document in the Eclipse editor (e.g. select it from the Package Explorer).
2. Select a range of code or data by mouse or keyboard.
3. Add a comment by either
 - a) pressing the “Add new comment” button at top-right corner of the Comments Summary View, or

- b) using the “CTRL+SHIFT+ ' + ' “ keyboard shortcut.
- 4. Enter the data such as recipient or priority using the Detail View.
- 5. Leaving the Detail View will automatically save the comment.

5.2.2 Adding a global comment

- 1. Open a document in the Eclipse editor (e.g. select it from the Package Explorer).
- 2. Add a comment by either
 - a) pressing the “Add new comment” button at top-right corner of the Comments Summary View, or
 - b) using the “CTRL+SHIFT+ ' + ' “ keyboard shortcut.
- 3. Enter the data such as recipient or priority using the Detail View.
- 4. Leaving the Detail View will automatically save the comment.

5.3 Sharing comments and reviews

Sharing comments and reviews can be achieved in multiple ways. Nonetheless we suggest to use a VCS⁵ such as SVN. Using e.g. SVN in combination with the Eclipse Subversive plug-in the sharing can be achieved as follows:

- 1) Select the folder that stores the review data.
- 2) Use the context menu and select “Team ► Share project” and share the project into an existing SVN repository. This step can also be done using an arbitrary SVN tool (command line tool, Tortoise, RabbitVCS, ...).
- 3) Commit the commented code⁶.

Other reviewers can then participate in the review process by simply checking out code and the review data folder.

After the initial steps for sharing, you can proceed with reviewing. Be sure to

- a) select the file(s) you commented as well as the folder that contains the review data⁷ in the Package Explorer and commit them to the SVN repository.
- b) update the file(s) you want to review as well as the folder that contains the review data

before proceeding, as other authors might have added comments and reviews.

⁵ [Wikipedia - Revision Control](#)

⁶ In order to link comments to pieces of code, tags are inserted. These tags have to be committed, too.

⁷ default: “AgileReviews”

6 Advanced Functionality

6.1 Export Review Data

1. Mark the reviews you want to export by pressing and holding CTRL and selecting them via mouse.
2. Click on the “Export selected reviews” button.
3. Enter the required data in the “Export Review Data” wizard.
 1. Select an export template⁸ and an export location (“2” and “3” in fig. 11).
 2. The pre-selection of step 1 can be changed (see “3” in fig. 11). Click on a review to see responsibility and description (see “4” in fig. 11).
4. Click on “finish”.

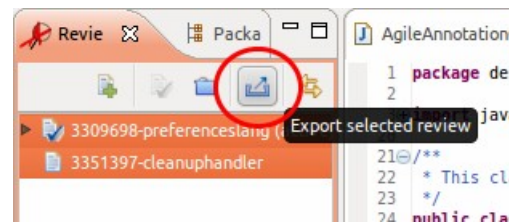


Figure 10: Exporting selected reviews

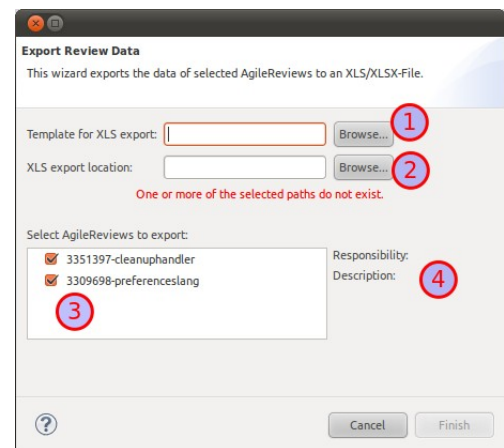


Figure 11: Exporting review data

6.2 Project Cleanup

In order to remove all tags from a certain project (e.g. if reviewed code of a branch is to be merged into the trunk, AgileReview provides the “cleanup” function.

1. Select the project you want to clean up from the Package Explorer (“1” of fig. 12).
2. Use the context menu and select “AgileReview project cleanup” (“2” of fig. 12).

6.3 Keyboard Shortcuts

| Function | Shortcut |
|----------------------|--|
| New review | CTRL+N, select „AgileReview ► Review” |
| New comment | CTRL+SHIFT+' + ' |
| Show comment | CTRL+SHIFT+' # ' 9 |
| Delete comment | CTRL+SHIFT+' - ' (editor) ALT+SHIFT+' - ' (detail view) |
| Add reply | ALT+SHIFT+' + ' (detail view) |
| Next comment in list | Ctrl+Shift+' N' |
| Next comment in file | Ctrl+Alt+Shift+' N' |

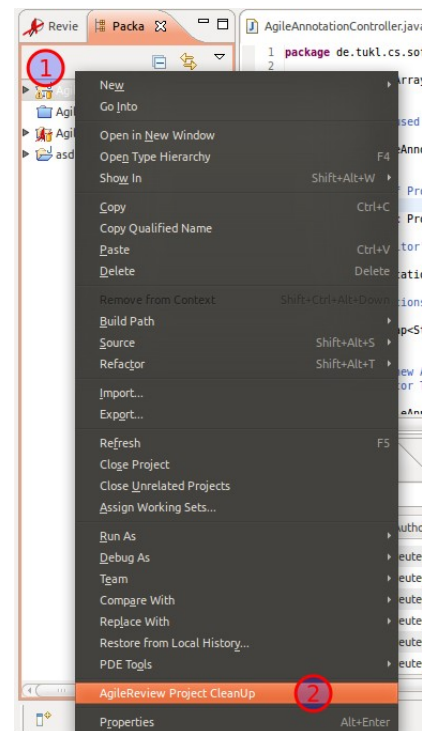


Figure 12: AgileReview cleanup function

⁸ An exemplary template can be found at the [AgileReview Sourceforge page](#).

⁹ Cursor needs to be in a highlighted range of code.