Contents

Using Crucible .................................................. 4
   The Crucible workflow ...................................... 4
Using the Crucible Screens .................................. 10
   Browsing All Reviews ...................................... 12
   Browsing Source Files ..................................... 13
   Using the Dashboard ....................................... 15
Crucible Icons .................................................. 20
Searching Crucible .............................................. 21
Viewing People's Statistics in Crucible ..................... 26
Browsing Projects .............................................. 29
   Viewing Project Statistics .................................. 31
   Viewing Reports ............................................. 33
      Review Coverage Report ................................. 36
Changeset Discussions ........................................ 40
Changing your User Profile ................................... 42
Roles and Status Classifications .............................. 42
Conducting a Review .......................................... 43
   Creating a Review .......................................... 43
      Creating a Patch Review .................................. 44
      Creating a Review within Crucible .................... 52
      Creating a review from FishEye ........................ 55
      Creating a review from JIRA ............................ 56
      Creating a Review from a URL ......................... 57
      Creating a Snippet Review ............................... 58
   Selecting the Files for the Review ....................... 59
      Iterative Reviews ....................................... 66
   Adding Reviewers .......................................... 66
   Issuing a Review .......................................... 69
   Performing the Review ..................................... 71
      Adding Comments ........................................ 72
      Flagging Defects ........................................ 74
      Completing your Review ................................. 75
   Sending a Review's Comments via Email ................. 76
   Using JIRA Integration in Crucible Reviews ............ 78
   Using the Review History Dialog ......................... 81
   Tracking Crucible Review Metrics ....................... 82
      Using Progress Tracking ................................ 82
      Using Time Tracking .................................... 84
   Summarising and Closing the Review ...................... 85
   Moving a Review to Another Project ..................... 87
   Deleting a review .......................................... 88
   Defining your Workflow .................................. 89
   Viewing Your Favourites .................................. 91
   Using Favourites .......................................... 93
   Using Keyboard Shortcuts in Crucible ................... 96
   Using RSS Feeds in Crucible .............................. 98
   Using Wiki Markup in Crucible ......................... 98
   Using Gadgets in Crucible ................................ 103
   Using Review Reminders .................................. 108
   Transitioning JIRA issues .................................. 108
Glossary .......................................................... 109
<table>
<thead>
<tr>
<th>Term</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>approve</td>
<td>110</td>
</tr>
<tr>
<td>authors in Crucible</td>
<td>110</td>
</tr>
<tr>
<td>code review</td>
<td>110</td>
</tr>
<tr>
<td>comment</td>
<td>110</td>
</tr>
<tr>
<td>creator</td>
<td>111</td>
</tr>
<tr>
<td>defect</td>
<td>111</td>
</tr>
<tr>
<td>moderator</td>
<td>111</td>
</tr>
<tr>
<td>participant</td>
<td>111</td>
</tr>
<tr>
<td>permission scheme</td>
<td>111</td>
</tr>
<tr>
<td>permissions in Crucible</td>
<td>111</td>
</tr>
<tr>
<td>projects in Crucible</td>
<td>113</td>
</tr>
<tr>
<td>review duration</td>
<td>113</td>
</tr>
<tr>
<td>reviewer</td>
<td>113</td>
</tr>
<tr>
<td>role</td>
<td>113</td>
</tr>
<tr>
<td>state</td>
<td>113</td>
</tr>
<tr>
<td>statement of objective</td>
<td>114</td>
</tr>
<tr>
<td>users in Crucible</td>
<td>114</td>
</tr>
</tbody>
</table>
Using Crucible

This page is an index of the content in the Crucible User’s Guide. Click on a link below to see the desired page.

- The Crucible workflow
- Using the Crucible Screens
  - Browsing All Reviews
  - Browsing Source Files
  - Using the Dashboard
  - Crucible Icons
  - Searching Crucible
  - Viewing People’s Statistics in Crucible
  - Browsing Projects
    - Viewing Project Statistics
- Viewing Reports
  - Review Coverage Report
  - Changeset Discussions
- Changing your User Profile
- Roles and Status Classifications
- Conducting a Review
  - Creating a Review
    - Creating a Patch Review
    - Creating a Review within Crucible
    - Creating a review from FishEye
    - Creating a review from JIRA
    - Creating a Review from a URL
    - Creating a Snippet Review
  - Selecting the Files for the Review
  - Iterative Reviews
- Adding Reviewers
- Issuing a Review
- Performing the Review
  - Adding Comments
  - Flagging Defects
  - Completing your Review
  - Sending a Review’s Comments via Email
  - Using JIRA Integration in Crucible Reviews
  - Using the Review History Dialog
  - Tracking Crucible Review Metrics
    - Using Progress Tracking
    - Using Time Tracking
- Summarising and Closing the Review
- Moving a Review to Another Project
- Deleting a review
- Defining your Workflow
- Viewing Your Favourites
  - Using Favourites
- Using Keyboard Shortcuts in Crucible
- Using RSS Feeds in Crucible
- Using Wiki Markup in Crucible
- Using Gadgets in Crucible
- Using Review Reminders
- Transitioning JIRA issues

The Crucible workflow

This page contains a basic overview of Crucible workflows, followed by a simple example showing a code review.
between two people.

Crucible is a flexible application that caters for a wide range of team sizes and work styles. You will need to know about the basic roles used in Crucible.

Roles:
There are several roles that review participants can take up:

- **authors**: Usually the creator of the code; the person who will act on the review's outcome.
- **reviewer**: A participant that will comment on the source files in the review, raising points and discussion on the work that was done.
- **moderator**: Usually the person who starts the review and is responsible for deciding the outcomes and closing it. The moderator is disabled for the "agile" permission scheme to simplify workflow.

You will also need to understand how workflow is conducted in Crucible. This is configurable, but the most basic example follows.

Crucible Workflow:
There are a number of different ways in which you can use Crucible for code reviews. The following diagram shows the basic workflow that applies to most Crucible code reviews.

**On this page:**

Example Workflow: Two Participant Code Review
1. The Author Starts the Review
2. The Reviewer Comments on the Code
3. The Author Responds to the Comments
4. The Author Closes the Review

Diagram: Workflow for One-to-One Reviews

Need more information? Read more about the different forms of [workflow in Crucible](#).

Next, we explore the workflow in a two-person code review in Crucible.

**Example Workflow: Two Participant Code Review**

This is a simplified set of instructions for executing a one-to-one review involving two people. In this example, the code author wears "two hats", acting as **review creator** and **code author**, managing the review process as well as taking final responsibility for closing the review. The second person is the reviewer.

For instructions on Crucible workflow with more than two people, see [this page](#).
1. The Author Starts the Review

To begin, the code author sets up the review. There are a number of ways to do this, but for this example, the author starts from the FishEye Source view of the file he wants to review:

Screenshot: Opening a review from the FishEye Source view

From the FishEye Source view, the author chooses Reviews > Create Review. If there are multiple projects, the Select Project dialogue opens.

Screenshot: The Select Project dialogue

In the Select Project dialogue, you are prompted to choose a project for this review from the drop-down list. Once the selection is made, the author clicks Create Review. The Edit Review Details dialogue opens, where the author can create and issue the review.

Screenshot: Creating a review in the Edit Review Details dialogue
In the Edit Review dialogue, the author enters information needed for the review. This includes entering a title and description for the review, selecting reviewers, a due date and the key for a related JIRA issue (if any). The project and author are pre-selected.

The author can also add more content to the review, if they wish, by clicking Add Content. See Selecting the Files for the Review.

When finished, the author clicks Done. The review will now be created in a draft form.

Screenshot: A new Crucible review

The draft review opens. In the draft stage, the author can check the contents of the review files to ensure they are correct and put in any notes for reviewers as comments. During the draft phase, no notification emails are sent out to reviewers. Once the author is finished with the draft phase, he clicks Start Review.

The review will now be started and a notification email will go out to all participants. This lets them know that the review is under way and prompts them to take action, providing a URL for direct access to the review.
2. The Reviewer Comments on the Code

The reviewer will receive an email from Crucible with a link that they can follow to the review.

*Screenshot: A Crucible review notification email*

When the reviewer clicks the link in the notification email, the Crucible Review screen opens.

*Screenshot: The Crucible Review screen*

On the Crucible Review screen, the code changes under review are displayed. The reviewer clicks file names to expand the code for in-line reviewing. As the reviewer reads the changes, they can simply click on any line to enter a comment there (multiple lines can be selected by clicking and dragging).

The reviewer clicks **Post** when each comment is finished.

The reviewer repeats this process for all files in the review. Reviewers can leave the session and resume it later;
their work is automatically saved.

When the reviewer has finished their code review work, they click Complete.

By default, an email is sent to participants every time a comment is posted. This is an individual setting. Each reviewer can configure their own profiles to adjust the list of events that will trigger email notifications.

3. The Author Responds to the Comments

During the review process, the author can also make contributions, responding to reviewer comments and making corrections.

Screenshot: Comment threads in Crucible

4. The Author Closes the Review

When all reviewers have "Completed" their reviews, the author is notified via email. The author clicks the link in the notification email, returning to the Review screen.

The author will then add any final comments, then click Close when finished.

This closes the review, signalling the end of work. A final email notification will be sent to the review participants, informing them that the review is now closed. The closed review screen will load, archiving the completed review as read-only.

Screenshot: Viewing a closed review
If the author ever needs to resume work on the closed review, they can simply click **Reopen** when viewing this screen. Doing this will return the review's status to "open".

For more information on workflow in Crucible and best practices for code reviews, see [Requesting and Conducting a Review](#).

**Using the Crucible Screens**

This page contains an overview of the Crucible interface and the actions that can be carried out in the application.

**On this page:**

- [Tour of the Crucible Interface](#)
- [Left Navigation Sidebar](#)
  - [Related Links](#)

**Tour of the Crucible Interface**

When you first log in to Crucible, the Dashboard Screen opens, as shown in the screenshot below. This view shows recent general activity in Crucible.

[Screenshot: The Dashboard Screen in Crucible](#)
The table below explain the top-level tabs in the Crucible User Interface. Click on the name of a tab for more information.

<table>
<thead>
<tr>
<th>Element name</th>
<th>Function</th>
<th>Appears</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dashboard Tab</strong></td>
<td>Displays reviews and system activity related to you.</td>
<td>All screens.</td>
</tr>
<tr>
<td><strong>Source Tab</strong></td>
<td>Displays contents of connected source repositories.</td>
<td>Only when FishEye is used with Crucible.</td>
</tr>
<tr>
<td><strong>Source Dropdown Menu</strong></td>
<td>Displays a list of links to repositories recently visited by the user</td>
<td>Only when FishEye is used with Crucible for logged-in users.</td>
</tr>
<tr>
<td><strong>Projects Tab</strong></td>
<td>Displays reviews and content from specific projects.</td>
<td>All screens.</td>
</tr>
<tr>
<td><strong>Projects Dropdown Menu</strong></td>
<td>Displays a list of links to projects recently visited by the user</td>
<td>All screens for logged-in users.</td>
</tr>
<tr>
<td><strong>People Tab</strong></td>
<td>Displays metrics on the users of the Crucible instance.</td>
<td>All screens.</td>
</tr>
<tr>
<td><strong>People Dropdown Menu</strong></td>
<td>Displays a list of links to user pages recently visited by the user</td>
<td>All screens for logged-in users.</td>
</tr>
<tr>
<td><strong>Reviews Tab</strong></td>
<td>Allows you to search and report on reviews.</td>
<td>All screens.</td>
</tr>
<tr>
<td><strong>People Dropdown Menu</strong></td>
<td>Displays a list of links to reviews recently visited by the user as well as links to the Crucible Inbox and Outbox</td>
<td>All screens for logged-in users.</td>
</tr>
</tbody>
</table>
Left Navigation Sidebar

The navigation bar at the left of the screen applies specific filters to what is shown in the centre pane. See the page on Using the Dashboard for more information.

The left navigation sidebar can be hidden or displayed by clicking the blue ‘information’ icon at the top left of the sidebar.

Related Links

Browsing Source Files
Browsing Projects
Viewing People's Statistics in Crucible
Viewing Reports
Searching Crucible
Using RSS Feeds in Crucible
Changing your User Profile

Browsing All Reviews

The instructions on this page describe how to browse all reviews on the ‘Reviews’ screen, by people or for projects that you have selected as favourites. This includes reviews that you are involved with. You can also generate reports on review blockers for all people.

The ‘Reviews’ tab is essentially a shortcut to viewing your reviews (or a custom filtered list, if you have set up a filter). You can also view all reviews filtered by different statuses, e.g. Open, Closed.

To browse all reviews:

1. Click the arrow next to the ‘Reviews’ tab at the top of the screen, and click ‘All Reviews’ in the dropdown menu. The ‘Reviews’ page will be displayed, showing all open reviews (‘All Open Reviews’ will be highlighted in the sidebar) unless you have previously used a custom filter. See the screenshot below.
2. Browse your reviews, by clicking the links under the ‘My Reviews’ and ‘My Snippets’ sections in the sidebar.
3. Browse reviews for all people by clicking the links under the ‘Everyone’s Reviews’ and ‘Everyone’s Snippets’ sections in the sidebar, as follows:
   - ‘Everyone’s Reviews’ — Click ‘See reviews by state...’ to expand the list of review categories in the sidebar. You can then click any of the links to view the relevant list of reviews:
     - ‘All Open Reviews’ — Click to view all open reviews, i.e. reviews that have not been summarised and closed yet.
     - ‘All Closed Reviews’ — Click to view all closed reviews, i.e. reviews that have been summarised and closed.
     - ‘All Reviews’ — Click to view all reviews, including open reviews, closed reviews and draft reviews.
   - ‘Everyone’s Snippets’ — Click ‘See snippets by state...’ to expand the list of snippet categories in the sidebar. You can then click any of the links to view the relevant list of snippet reviews, similar to the links described above. See Browsing All Reviews for more information.
     - ‘All Open Snippets’ — Click to show all open snippets.
     - ‘All Snippets’ — Click to show all snippets, i.e. open and closed snippets.

Screenshot: Browsing all Reviews on the 'Reviews' screen
When FishEye is installed with Crucible, you have the additional 'Source' tab available in the navigation tabs at the top of the screen.

To browse source files:

1. Click the 'Source' tab. The 'Repositories' view will be displayed, showing summary information if you have multiple repositories set up. See the 'Viewing all repositories' screenshot below.
2. Click the desired repository to view its contents. See the 'Viewing a repository' screenshot below.
3. Browse the repository for the desired source file using the directory tree in the left menu. Click the file that you want to view. The file will be displayed in the main panel. See the 'Viewing a file' screenshot below.
4. You can view various information about the file, as outlined in the table below:

<table>
<thead>
<tr>
<th>Sub-Tab Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revisions</td>
<td>When viewing a file, shows the latest revisions of the file.</td>
</tr>
<tr>
<td>Files</td>
<td>When viewing a folder, shows the contents of the directory.</td>
</tr>
<tr>
<td>Activity</td>
<td>Shows recent activity on the item. There are a number of sub-options here:</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>* All Activity — The default view, showing commits, reviews and JIRA issues.</td>
</tr>
<tr>
<td></td>
<td>* Commits — Shows commits in the activity stream.</td>
</tr>
<tr>
<td></td>
<td>* Reviews — Shows review activity in the activity stream.</td>
</tr>
<tr>
<td></td>
<td>* Scroll to Changeset — Opens the changeset ID specified in the text field (press Enter to carry out the action).</td>
</tr>
<tr>
<td></td>
<td>* Filter — Applies constraints to the current activity stream.</td>
</tr>
<tr>
<td></td>
<td>* Show Revisions — If this is selected, then changeset items are automatically expanded to show modified files.</td>
</tr>
<tr>
<td></td>
<td>* Earlier Activity (Left Arrow icon) — Loads a page of earlier activity.</td>
</tr>
<tr>
<td></td>
<td>* Later Activity (Right Arrow icon) — Loads a page of later activity.</td>
</tr>
</tbody>
</table>

| Users             | Shows the commit history of the different users that have committed changes on the item. |

| Reports           | Shows activity charts for the item. Various chart options can be selected in the left navigation bar. |

| Source            | Shows the contents of the file. |

| Query             | Allows you to run an advanced search. |

To download files, firstly click through the desired file. From there, you will see a control bar directly above the code content which contains the 'FishEye' item. Clicking this leads to a drop-down menu where 'Download Raw File' is available. You can use this to download the file in context only.

**Screenshot: Viewing all repositories**

![](image1.png)

**Screenshot: Viewing a repository**

![](image2.png)
Using the Dashboard

The Dashboard is the first screen that you see when you log into FishEye/Crucible. It is the home page for the instance and can be accessed by clicking the FishEye/Crucible icon, or by clicking the 'Dashboard' tab at the top.

Screenshot: Viewing the Dashboard
The Dashboard itself has a sidebar displaying your reviews and an activity in the main pane.

Using the 'My Reviews' Sidebar

The 'My Reviews' sidebar contains information about reviews/snippets that you are involved with, e.g. how many reviews require your approval. Read more about the sidebar in Browsing All Reviews.

- Click on any of the links in the sidebar to navigate to that information in the 'Reviews' tab of the Dashboard.
- Hover your mouse over the sidebar and click the collapse icon (▼) to hide any of the sections. Click the expand icon (►) to expand any collapsed sections.
- Click the 'i' icon (ℹ️) to hide/show the entire sidebar.

<table>
<thead>
<tr>
<th>'Inbox'</th>
<th>Click to show all reviews in 'To Review', 'To Summarize', 'In Draft' and 'Require My Approval' statuses:</th>
</tr>
</thead>
<tbody>
<tr>
<td>'To Review'</td>
<td>— Click to show all reviews where you are a reviewer and haven't yet completed your review work.</td>
</tr>
<tr>
<td>'To Summarize'</td>
<td>— Click to show all reviews where you are a moderator and haven't yet summarised and closed the review.</td>
</tr>
<tr>
<td>'In Draft'</td>
<td>— Click to show all reviews that you have created but have not yet been moved to the 'Approval' state or the 'Require Approval' state.</td>
</tr>
<tr>
<td>'Require My Approval'</td>
<td>— Click to show all reviews where you are a moderator and need to approve the review.</td>
</tr>
</tbody>
</table>
Using the Activity Stream

The Dashboard has an activity stream that displays the following information:

- **reviews activity** — This includes the addition of review comments, opening and closing reviews, etc.
- **source activity** — This includes files being committed to a repository.
- **issues activity** — If you have linked a JIRA server with your Crucible project, the activity stream will also include updates to linked JIRA issues.

Your activity stream will only display information from projects, reviews, people, repositories, etc, that you have selected as favourites as well as your own activity. For more information on favourites, see [Using Favourites](#).

The following instructions describe how to view the activity stream and filter it to display specific information.

- **Commits** tab — See [Browsing Source Activity only](#) below.
- **Reviews** tab — See [Browsing Reviews Activity only](#) below.
- **Issues** tab — This tab will only display if you have connected a JIRA instance to your FishEye/Crucible instance. See [Browsing Issues Activity only](#) below.
- **Show My Activity** checkbox — Untick this checkbox to filter out your activity from the activity stream displayed. This toggle can be used in any of the tabs for the activity stream.

**Browsing Source Activity only**

Source activity includes files commits to repositories that you have selected as favourites.

Click the **Commits** tab to filter the activity stream to display only source activity (see screenshot below).

**Browsing Source Activity on the Dashboard**
Browsing Reviews Activity only

Reviews activity includes updates to reviews in all projects that you have selected as favourites. See Browsing All Reviews for more information about browsing reviews.

Click the 'Reviews' tab to filter the activity stream to display only reviews activity (see screenshot below).

Browsing Reviews Activity on the Dashboard
**Browsing Issues Activity only**

Issues activity includes updates to issues in JIRA projects that are associated with your favourite Crucible projects. For more information about integrating JIRA with Crucible, see [JIRA Integration in Crucible](#).

Click the ‘Issues’ tab to filter the activity stream to display only issues activity (see screenshot below).

**Browsing Issues Activity on the Dashboard**
Watching an Activity Stream

You can "watch" an activity stream in FishEye/Crucible. Watching the activity stream allows you to receive emails when updates occur in the activity stream. You can view all of your watches and configure the frequency of your watch emails in your user profile. See Changing your User Profile for more information.

Note, the option to add a watch will only be available if the administrator has enabled watches for the repository.

To watch an activity stream:

1. Navigate to the activity stream that you want to watch.
2. Click the 'Tools' menu and click 'Watch'. The page will reload and a watch will be set up for the activity stream (the watch icon will be coloured, not grey).
   - If you want to remove the watch, navigate the activity stream, click the 'Tools' menu and click 'Watch'. The watch will be removed (the watch icon will be coloured, not grey).
   - You can also remove watches via your user profile.

Notes

Related Topics

- Browsing All Reviews
- Using Favourites
- Changing your User Profile

Crucible Icons

This page contains a list of Crucible icons and an explanation what each one represents in the user interface.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>📝</td>
<td>View review-level comments</td>
</tr>
</tbody>
</table>
Searching Crucible

Crucible has a powerful search engine that allows you to find reviews. There are two methods for searching in Crucible:

- **Quick Search** — The Quick Search allows you search all Crucible projects by entering a single search string. This search is the default search and will suggest "quick nav" results (header search box only). Results are weighted by most recent edit date.
- **Filtering Reviews** — An alternative method for searching for reviews is to display all reviews and apply a custom filter to the list. This is generally slower than searching, but allows you to specify filter criteria against a range of fields.
- **Comment Search** — If you want to find specific review comments, Crucible provides a powerful comment search.

On this page:
- Using the Quick Search
- Filtering Reviews
- Searching for Review Comments
- Notes

Using the Quick Search

Before you begin:

- The Quick Search will also return changesets and files, if you are using FishEye with Crucible. For information on searching FishEye, see Searching FishEye in the FishEye documentation.

To search Crucible using the Quick Search:

1. Enter your search criteria in the search box in the Crucible header (Quick Nav). Crucible offers a number of parameters and functions that you can use to refine your expected results, see Refining your Quick Search Criteria below.
2. "Quick Nav" results will appear in a dropdown, as you type. "Quick Nav" will attempt to match against the review name, project and user.
- If you want to use a quick nav result, use the up- and down-arrows on your keyboard and press enter or use your mouse to select the item.
- If the quick nav results don't have what you are looking for, press enter to run a search. Ensure that no items in the dropdown are selected when you press enter.

3. The Quick Search results page will be displayed. You can filter your results further, as described in Refining Quick Search Results below.
   Results are sorted by relevance and boosted if they were edited recently. A maximum of 10 results are displayed per page.
   - If you have integrated your Crucible instance with a JIRA instance, you can display a summary of any JIRA issues referenced in your search results by hovering over the issue key. For more details, see JIRA Integration in Crucible.

4. If you want to run another search, enter your new criteria in the main search box or in the search box in the header.
   Note, only the search box in the header provides “quick nav” results.

![Screenshot](image.png)

Screenshot above: Quick Search displaying "quick nav" matches

**Refining your Quick Search Criteria**

The Crucible Quick Search has a number of powerful tools that you can use to refine your search criteria before executing the search.

<table>
<thead>
<tr>
<th>Search Tool</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
</table>

Created by Atlassian in 2012. Licensed under a Creative Commons Attribution 2.5 Australia License.
Field Handles

Use a field handle in your criteria to restrict your search to a particular field. Note, you cannot have multiple field handles in a query.

- `detail` — Search against title, objective, key, linked reviews and linked issues.
- `reviewcomment` — Search against review comments.

Searching for Discrete Strings

Enter a specific string within quotation marks and Crucible will match against the exact string. Note, this search is not case-sensitive.

Enter "CR-2818" and Crucible will only return results that match that exact string, i.e. it will not return a result with CR-FE-2818 or CR-28 189.

Filtering Quick Search Results

Once you have a set of search results on the Quick Search page, you can filter them to a subset of the original results. The filter controls are in the left panel of the Quick Search page in the 'Source' section.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Projects</td>
<td>Select or enter the name of the project that you want to restrict your results to. For example, if you enter 'CR' then the search results page will refresh to display only reviews in the 'CR' project. <strong>⚠️ If you are using Fisheye with Crucible, there will be a repositories dropdown in the 'Source' section. Selecting a FishEye repository in this dropdown will not filter the Crucible search results. It is only used to filter files and changesets returned in the search results. See Searching FishEye.</strong></td>
</tr>
<tr>
<td>Reviews</td>
<td>Click this link to restrict your results to reviews that have a title, objective, key, linked reviews or linked issues that match the search criteria.</td>
</tr>
<tr>
<td>Comments</td>
<td>Click this link to restrict your results to reviews that have comments that match the search criteria.</td>
</tr>
</tbody>
</table>

Filtering Reviews

Crucible allows you to view all the reviews/snippets that you are involved with, as well as everybody's reviews/snippets. You can filter these lists to find reviews.

To filter a list of reviews:

1. Navigate to the 'Reviews' tab.
2. Click the list of reviews that you want to start with, in the reviews sidebar of the 'Review Dashboard', e.g. 'All Open Reviews'.
3. Click the ‘Custom Filter’ in the reviews sidebar.
4. Update the filters with your search criteria (see table below) and click ‘Apply Filter’ to filter the reviews.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Find reviews by searching for words within the title.</td>
</tr>
<tr>
<td>Project</td>
<td>Find reviews under a particular project.</td>
</tr>
<tr>
<td>Author</td>
<td>Find reviews moderated by a particular authors.</td>
</tr>
<tr>
<td>Moderator</td>
<td>Find reviews moderated by a particular moderators.</td>
</tr>
<tr>
<td>Creator</td>
<td>Find reviews created by a particular creator.</td>
</tr>
<tr>
<td>Reviewer</td>
<td>Find reviews that are reviewed by a particular reviewer.</td>
</tr>
<tr>
<td>Reviewer Status</td>
<td>This is reliant on the above filter and is used to show reviews that have either been completed by the reviewer, not completed or all reviews.</td>
</tr>
<tr>
<td>Match Roles</td>
<td>To use all the above filters, choose ‘all’. To use any of the filters, choose ‘any’.</td>
</tr>
<tr>
<td><strong>Review state checkboxes</strong></td>
<td>Check any of the review state checkboxes (e.g. 'Draft', 'Pending Approval') to filter for reviews in those states.</td>
</tr>
</tbody>
</table>

**Searching for Review Comments**

To search for review comments:

1. Navigate to the ‘Reviews’ tab.
2. Enter your search criteria in the ‘Comment Search’ section at the bottom of the reviews sidebar.
3. Click ‘Search Comments’.
4. The ‘Comment Search’ page will display your results. You can refine your search using the search criteria on the page:

<table>
<thead>
<tr>
<th>Project</th>
<th>Find comments on reviews under a particular project.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comment content</td>
<td>Find comments that contains the specified text.</td>
</tr>
<tr>
<td>Review PermaId</td>
<td>Find comments made on the specified review.</td>
</tr>
<tr>
<td>After</td>
<td>Find comments made after after a particular date.</td>
</tr>
<tr>
<td>Before</td>
<td>Find comments made after before a particular date.</td>
</tr>
</tbody>
</table>
### Comment Author
Find comments made by a particular user.

### Search Type
You can:

- Tick the 'Defects' check-box to find comments that are flagged as [Defects](#).
- Tick the 'Comments' check-box to find comments that are not flagged as [Defects](#).
- Tick neither check-box (or both of them) to find all.

<table>
<thead>
<tr>
<th>Review State</th>
<th>Find comments on reviews that are in a particular state. See Review State Filter (above).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ranking</td>
<td>Find defects have been given a particular ranking (e.g. 'Major', 'Minor').</td>
</tr>
<tr>
<td>Requires Re-Review</td>
<td>Find defects that have been marked as requiring re-review (or not).</td>
</tr>
<tr>
<td>Classification</td>
<td>Find defects that have been given a particular classification (e.g. 'Missing', 'Ambiguous').</td>
</tr>
</tbody>
</table>

![Search Criteria](image)
Searching FishEye (FishEye documentation)

Viewing People's Statistics in Crucible

This page contains instructions on how to use the People tab in Crucible to see charts and activity from people with accounts on the system.

On this page:

- Opening the List of People
- Viewing a Person's Activity Screen
- Viewing Charts on a Person's Activity

Opening the List of People

To view statistics on People in Crucible, (that is, code authors, committers and reviewers) click the People tab at the top of the page. The list of all People appears.

Screenshot: List of all People in Crucible (when using FishEye with Crucible)
The list of all people shows all users that have accounts on the system. By default, each user has a unique avatar that is randomly formed from the text in their email address. Users can choose to upload their own avatar image by uploading an image to an external service such as Gravatar, which Crucible supports. See the page on Changing your User Profile.

**Viewing a Person’s Activity Screen**

Click on a user to see a listing of activity from them as well as charts showing statistics for their activity. The People Activity screen opens.

**Screenshot: The People Activity Screen in Crucible**

In the right hand pane, we can see a list of all activity that relates to this user. You can click the icons to view full commit information in FishEye, click JIRA issue names to open the work ticket on an item, click the long button to see the list of files in context or click the star icon to add an item to your favourites.

In the left hand pane, we can see charts around this activity, such as the following: number of active reviews; charted history of lines of code; code committing activity and general statistics.
Some users may not appear to have the correct number of Files Changed or LOC, despite regularly committing. In this situation, if they have committed to a directory which is not covered by the regexes in your symbolic definition (i.e. they have committed to a directory that is neither trunk, branches or tags) then that directory will be counted as part of trunk. Also note that creating tags and branches themselves does not count toward the totals.

**Viewing Charts on a Person's Activity**

To see information on a person's activity charted in detail, click the headings in the left-hand pane. Each heading will show more information on demand, when clicked. The information available and what it means is listed below.

⚠️ The Charts in this section are only available when using FishEye.

*Screenshot: People Activity Charts in Crucible*
**Username** heading:
The username section shows the email address, then the first and latest commit dates for the person in context. It also shows username mappings from various systems if they have several usernames in play.

**Reviews** heading:
The Reviews section shows several filters that you can click to constrain the review items shown in the right-hand pane. The options are To Review, To Summarize, Out For Review, Open and Closed.

**Line History** heading:
The Line History section shows a graph with the number of lines committed to the repository, charted over time.

**Commit Activity** heading:
The Commit Activity section shows three smaller charts; the first showing the volume of commits over a 52 week period; the second showing the relative number of commits on days of the week; the third showing the relative number of commits by the hour of the day when they were lodged.

**Stats** heading:
The Stats section shows data points for the previous week and all-time. It shows number of commits, number of files changed and number of lines changed.

---

**Browsing Projects**
To browse the content in a project, click the Projects tab at the top of the page. The 'Projects' view opens (see 'The Crucible Projects Index' screenshot below).

A list of projects will be shown if there is more than one. Click the name of the desired project to open it. The 'Project Activity' page opens (see 'The Crucible Project View' screenshot below). In the left navigation bar, charts showing overall project statistics are displayed.

There are a number of sub-tabs on this page, listed in the table below.

<table>
<thead>
<tr>
<th>Sub-Tab Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| Activity     |  - All Activity — The default view.  
               - Commits — Shows commits in the project (visible when using FishEye).  
               - Reviews — Shows reviews in the project.  
               - Issues — Shows JIRA issues related to this project. Only visible if you have set up JIRA Integration in Crucible.  
               - Show Revisions — Shows or hides revisions in the project (visible when using FishEye).  
               - Earlier Activity (Left Arrow icon) — Loads a page of earlier project activity.  
               - Later Activity (Right Arrow icon) — Loads a page of later project activity. |
| Reviews      | Shows recent reviews in the project. |

The Projects tab is only visible in Crucible. Read more about the definition of a project.

**Screenshot: The Crucible Projects Index**

**Screenshot: The Crucible Project View**
Viewing Project Statistics

This page explains the layout of the Project Summary page.

On this page:

- Project Name Panel
- Project Line History Panel
- Project Stats Panel
- Project Commit Activity Chart

When you click through to a Crucible Project from the Projects Tab, the ‘Project Summary’ screen opens.

Screenshot: The Crucible Project Summary Page
In the right hand pane, you can see an activity stream relating to this project. In the left hand pane, you can see various statistics charts relating to the project in context. These appear in a reduced size until you click them, when they will expand to show more information.

**Project Name Panel**

This contains a short message explaining which Crucible Project and FishEye repositories are being accessed to show the activity stream on the page.

**Project Line History Panel**

This panel contains a chart showing the lines of code added to the repository, graphed over time.

**Screenshot: The Project Line History Panel**

![Line History chart](image)

**Project Stats Panel**

This panel contains a chart showing numerical data for commits, files changed and lines change, graphed over time.
**Screenshot: The Project Stats Panel**

<table>
<thead>
<tr>
<th>Stats</th>
<th>Last Week</th>
<th>All Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commits:</td>
<td>137</td>
<td>16,033</td>
</tr>
<tr>
<td>Files changed:</td>
<td>543</td>
<td>69,896</td>
</tr>
<tr>
<td>Lines changed:</td>
<td>33,543</td>
<td>504,583</td>
</tr>
</tbody>
</table>

**Project Commit Activity Chart**

This panel contains a number of charts:

<table>
<thead>
<tr>
<th>Chart</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>52 week commits volume</td>
<td>This chart shows the amount of commits, shown by week over a one year period.</td>
</tr>
<tr>
<td>Commits by day</td>
<td>This chart shows the amount of commits, graphed by day over the past week.</td>
</tr>
<tr>
<td>Commits by hour</td>
<td>This chart shows the amount of commits, graphed by hours over the past day.</td>
</tr>
<tr>
<td>Commit calendar</td>
<td>This chart shows the amount of commits (shown as darker colours to indicate more commits) graphed by month, over years that the repository has been running.</td>
</tr>
</tbody>
</table>

**Viewing Reports**

This page contains instructions on how to use the Reports tab in Crucible to see lists of people whose action is required on open reviews. These are known as ‘blockers’. 
FishEye Reports (on this page):

- Viewing the 'Review Blockers' report
- Viewing the 'JIRA Blockers' report
- Viewing the Review Coverage Report

Viewing the 'Review Blockers' report

To view a list of people who have open reviews assigned to them,

To view the 'Review Blockers' report:

1. Click the dropdown arrow next to the 'Reviews' tab at the top of the page and select 'Reports' from the dropdown menu.
2. Click the 'Review Blockers' link under the 'Reports' sub-tab. The 'Review Blockers' report will be displayed.
   - Click a user's name to go to their Activity screen.
   - Click a number in the 'To Complete' or 'To Summarize' column to go to the list of reviews waiting to be completed/summarised by the user.

Screenshot: 'Review Blockers' Report
Viewing the 'JIRA Blockers' report

The 'JIRA Blockers' report shows you a list of users whose action is required on open reviews, for a particular set of JIRA issues. The reviews must be explicitly linked to a JIRA issue or mention a JIRA issue key in the summary or the objectives.

To view the 'JIRA Blockers' report:

1. Click the dropdown arrow next to the 'Reviews' tab at the top of the page and select 'Reports' from the dropdown menu.
2. Click the 'JIRA Blockers' link under the 'Reports' sub-tab.
3. Enter the details of your JIRA server and project, and click the 'Go' button. The 'JIRA Blockers' report will be displayed with the following information:
   - A list of JIRA issues for which one or more Crucible reviewers has not completed their review.
   - A list of users who have an incomplete Crucible review that relates to a JIRA issue.
   - A list of open JIRA issues for which a Crucible review is closed, and vice versa.
**Review Coverage Report**

Crucible has useful reports that show you detailed statistics on review activity. The 'review coverage' report allows you to see how much of the code in your repository has been reviewed, which files and when. You can also access the reviews.

ℹ️ This feature requires FishEye integrated with Crucible.

**On this page:**

- [Opening the Review Coverage Report](#)
- [Using the Summary Panel](#)
- [Using the Review Coverage Overview](#)
- [Using the Individual Committer Statistics Panel](#)
  - [Using the Changesets Coverage Panel](#)

**Screenshot: The Review Coverage Report**
Opening the Review Coverage Report

To open the Review Coverage Report:

1. Click the 'Source' tab.
2. Select your repository. The repository you've chosen will set the scope for the Coverage Coverage report.
3. If desired, navigate down the tree to the desired path you want to view coverage on.
4. Click the Reports tab in the secondary toolbar.
5. Select 'Review Coverage Report' from the list of reports in the upper panel.

You can view coverage of any path by navigate down the tree to the desired path you want to view coverage on, before clicking on the 'Reports' tab.

Using the Summary Panel

The summary panel shows some choice metrics from your Crucible instance. The following information from your repository is arrayed:

- Overall review coverage percentage.
- Change in review coverage percentage since the last reporting period.
- Total number of reviews.
- Total number of comments.
- Total number of reported defects.
- Total number of Lines of Code (LOC).
- Total number of commits.
- Total number of committers.
- Total number of unreviewed lines.
- Total number of lines under review.
- Total number of reviewed lines.
- A ratio of the number of lines unreviewed against reviewed Lines of Code (LOC).

Screenshot: Summary Panel in the Review Coverage Report
Using the Review Coverage Overview

The Review Coverage Overview shows a timeline of reviews, compared against their percentage of coverage. Hover your mouse cursor over the data points on the graph to see granular information and click through to a detailed weekly report. You can click the tabs to view the coverage expressed as a percentage of lines of code, changesets or revisions.

Screenshot: Overview Panel in the Review Coverage Report

Using the Individual Committer Statistics Panel

The Individual Committer Statistics window lets you choose a user from your Crucible instance and see all the changesets by that committer.

Screenshot: Individual Committer Statistics in the Review Coverage Report
Using the Changesets Coverage Panel

The Changesets Coverage Panel lets you see changesets from your Crucible instance (for the time period of the report), and their level of review coverage. This information can be sorted by the columns in this view and uses colour coding to denote review coverage (listed in the table below).

Colour Key

<table>
<thead>
<tr>
<th>Colour</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>dark green</td>
<td>reviewed</td>
</tr>
<tr>
<td>light green</td>
<td>in review</td>
</tr>
<tr>
<td>red</td>
<td>not reviewed</td>
</tr>
</tbody>
</table>

Screenshot: Changesets Coverage panel in the Review Coverage Report
Changeset Discussions

When using Crucible with FishEye, you can have threaded discussions with other users, on any changeset. To start a discussion, you simply start by adding a comment to a changeset.

You need to be logged in to create changeset comments.

Adding Comments to Changesets

To add a comment to a changeset:

1. Open the changeset view for the desired code commit.
2. Display comments by clicking 'Discuss' at the upper right corner, or the speech bubble icon beside the left navigation bar.
3. When the comment bar is visible, you can add a comment by clicking 'Add a Comment'. Type your content and click 'Post' to submit it.
4. You can tag your comment as a defect note by clicking the 'Defect' tick box.
5. Once submitted, others can respond to your comment by clicking 'Reply'. Replies are threaded as separate comment discussions. You can click on the link icon to save a permalink to that comment. The comment author can edit or delete their comments.
6. To hide the changeset comments, click the page icon. You can open the comments bar by clicking the speech bubble icon again.

As you compose a comment, it will auto-save periodically.

Screenshot: Opening Changeset Discussions
Turning Changeset Discussions On and Off

You can turn off changeset discussions in the Admin screens.

Go to the Admin screen, then choose 'Repository List' from the left navigation bar. Find your repository from the list that appears, and click 'View' beside it to see the repository settings page. Scroll to the bottom of the list
and find ‘Changeset Discussions’. Click ‘Edit’ to change the value to true or false. If set to false, changeset discussions are disabled.

By default, changeset discussions are on.

**Notifications**
- Comments show up in the activity stream,
- The author of the changeset will get email notifications when comments are added,
- Comment authors will get email notifications when someone replies to their comments.

### Changing your User Profile

See [Changing your User Profile](#) in the FishEye documentation for instructions on how to change your user profile.

### Roles and Status Classifications

This page explains the roles & status classifications in Crucible.

- **Roles in Crucible**
  - Author
  - Creator/Moderator
  - Reviewer
  - User

- **Status Classifications in Crucible**
  - Draft
  - Under Review
  - Summarized
  - Closed
  - Abandoned

#### Roles in Crucible

**Author**

The page author does not exist.

**Creator/Moderator**

The creator is the person who creates the review. In most cases this person will also act as moderator. The moderator is the person responsible for creating the review, approving the review, determining when reviewing is finished, summarising the outcomes and closing the review. By default, the moderator is the creator. See also author, the person whose changes to the code are to be reviewed.

**Reviewer**

A reviewer is a person assigned to review the change. Reviewers can make comments and indicate when they have completed their review. The moderator and author are implicitly considered to be participants of the review, but are not reviewers.
User

A user is a person using Crucible.

Status Classifications in Crucible

Draft

Draft Reviews are not yet completed or released to the reviewers.

Under Review

Reviews Under Review are either waiting for attention by reviewers or waiting to be summarized.

Summarized

Summarized reviews are past the reviewing phase. The moderator can still add conclusions or comments.

Closed

Closed reviews are complete.

Abandoned

Abandoned reviews are ‘in the trash’. Reviews must be Abandoned before they can be deleted.

See also the Glossary of terms used in Crucible.

Conducting a Review

This page contains links to instructions how to create a review and manage the workflow through its various states to completion. Click on the desired topic to see more information.

- Creating a Review
- Selecting the Files for the Review
- Adding Reviewers
- Issuing a Review
- Performing the Review
- Summarising and Closing the Review
- Moving a Review to Another Project
- Deleting a review

For an overview of how to apply a workflow to Crucible, see Defining your Workflow.

For an explanation of the different roles that people play in a review, see Roles and Status Classifications.

Creating a Review

This page explains how to create a Crucible review.

There are a number of ways to create a review. Choose from the list below:

- Creating a Patch Review
- Creating a Review within Crucible
- Creating a review from FishEye
- Creating a review from JIRA
- Creating a Review from a URL
- Creating a Snippet Review
Whichever way you choose, the overall process looks like this:

![Process Flowchart]

Note that only people with the 'Create' permission can create a review.

## Creating a Patch Review

This page includes instructions on uploading patch files from your repository, how to load them into Crucible to be reviewed and use Crucible's Patch Anchoring to retrieve more lines of context from the original file.

**On this page:**

- Using Crucible Patch Anchoring to Automatically Add Full Context
- Creating a Patch File From Your IDE
  - Creating a Patch File in IntelliJ IDEA 7.0
  - Creating a Patch File in Eclipse 3.3.1.1
- Creating a Basic Patch File From The Repository Command Line
  - CVS Patch Creation On The Command Line
  - Subversion Patch Creation Via The Command Line
  - Perforce Patch Creation Via The Command Line
  - Mercurial Patch Creation Via the Command Line
  - Git Patch Creation Via the Command Line
- Creating Patches That Include All Lines of Code
  - Creating a Patch in CVS With All Lines of code
  - Creating a Patch in Subversion With All Lines of Code
  - Creating a Patch in Perforce With All Lines of Code

Crucible allows you to review a change before it has been committed. To do this, you upload a patch file to the 'Patch' tab (or paste it in as text) when creating a review. You must first generate this patch file from your repository, using either commands built into your IDE, or via the repository command-line tools.

To create a patch in Perforce, you must ensure you have set P4DIFF to point to a GNU-compatible diff program.

By default, patch files will only show a few lines of code surrounding each change, rather than the entire file and its changes. Crucible's patch anchoring feature overcomes this limitation.

### Using Crucible Patch Anchoring to Automatically Add Full Context

Crucible's Patch Anchoring feature allows you to add a regular patch to a review (showing only a few lines of context). Then, Crucible will automatically search for the relevant file content in the connected repositories. When it finds the files, it will seamlessly add in more context from the file so that you can view all of the lines of code (greatly enhancing the review process).

To use patch anchoring:

1. Create a new review. From the 'Tools' menu in Crucible, select 'Create Review'.
2. Click 'Pre-Commit - Upload a patch file to be reviewed'. The 'patch upload' dialog appears. Click 'Browse', locate your file, then click 'Upload'. Crucible will now search for matches in the files in its database. Crucible will analyse all the paths in the patch, find the branches containing all those paths, then anchor the patch to the trunk or the branch with the most recent commit activity.

   Crucible makes a 'best guess' in its processing – you should check that it has anchored the patch to the correct location in your repository.

**Screenshot: Crucible Patch Anchoring**

![Add Content to Review TEST-56](image)

1. You can click 'Edit' to change the anchoring, by choosing a new match or removing the anchor. You can change the anchoring later, after the review is live.
2. Start the review. When viewing the diffs, you will be able to choose more than three lines of context from the 'View' menu.

**Screenshot: Editing Patch Anchoring Settings**
Creating a Patch File From Your IDE

Creating a Patch File in IntelliJ IDEA 7.0
To create a Patch File under IntelliJ IDEA, do the following:

Select a parent folder, sub-folder or file that you have altered in the Project tool window. Select 'Version Control' > 'Create Patch'. The following window appears:

Screenshot: The IDEA Create Patch window

Click 'Create Patch'. Choose a location to save the patch file and click 'Ok'.

If You Do Not Have the Create Patch Command Available Under IDEA

If you have not configured version control in IDEA, you may not have the 'Create Patch' option available. If so, use the following steps to create a patch file in IDEA:

1. Select a parent folder, sub-folder or file that you have altered in the Project tool window, right-click it and choose 'Local History' > 'Show History'.

Screenshot: The IDEA Show History dialog
2. In the Local History view, right-click the revision number, and choose ‘Create Patch’.

Screenshot: The IDEA Create Patch dialog

3. In the Create Patch dialog, choose a location for the patch file and a file name, then click ‘OK’.

Creating a Patch File in Eclipse 3.3.1.1

To create a patch file under Eclipse, do the following:

Find the parent folder, sub-folder or file that you have altered, right-click it and choose ‘Team’ > ‘Create Patch’.

Screenshot: Instigating a Patch in Eclipse
In the Create Patch window, choose a location on your computer and type an appropriate file name (the file format is plain text).

Screenshot: The Eclipse Create Patch dialog
Creating a Basic Patch File From The Repository Command Line

**CVS Patch Creation On The Command Line**

To create a patch in CVS, use the `cvs diff -Nu` command from your workspace. For example:

```
cvs diff -Nu > patch.txt
```

Note that patch files created with this command will only include around three lines of code, before and after each change. It will include revision information by default, so if you have FishEye, you may be able to anchor the patch to get full context.

**Subversion Patch Creation Via The Command Line**

To create a patch in Subversion, use the `svn diff` command from your workspace. For example:

```
svn diff > patch.txt
```

⚠️ `svn diff` does not print any information about files copied in the workspace.

Note that patch files created with this command will only include around three lines of code, before and after each change. It will include revision information by default, so if you have FishEye, you may be able to anchor
the patch to get full context.

**Perforce Patch Creation Via The Command Line**

Use `p4 diff -dcu` to generate a patch for changed files. For example:

```
p4 diff -dcu > patch.txt
```

The "-dcu" option provides a combination of "context format" and "unified format". It provides the diff in a standard unified diff format (which we need to parse the diff) as well as revision information (which we need to anchor to FishEye).

Since Perforce diffs do not include added and deleted files so you should then do a `p4 opened` to find such files. For added files, in UNIX you can append them individually to the patch using GNU `diff`:

```
diff -u /dev/null path_to_added_file >> patch.txt
```

In the example above, replace `path_to_added_file` with the actual path of your added file. You can follow a similar procedure with deleted files using `p4 print` to extract the previous version of the file.

Note that patch files created with this command will only include around three lines of code, before and after each change. It will include revision information by default, so if you have FishEye, you may be able to anchor the patch to get full context.

**Mercurial Patch Creation Via the Command Line**

Use `hg diff` to generate a patch for changed files. For example:

```
hg diff > patch.txt
```

This will include 3 lines of context. It will also include revision information, so if you have FishEye, you may be able to anchor the patch to get full context.

Note: if you use git style diffs (`--git`), the revision information will not be provided. This means that we cannot anchor the patch to a FishEye repository.

**Git Patch Creation Via the Command Line**

Use `git diff` to generate a patch for changed files. For example:

```
git diff > patch.txt
```

This will include 3 lines of context. It will also include revision information, so if you have FishEye, you may be able to anchor the patch to get full context.

**Creating Patches That Include All Lines of Code**

To create a patch file that shows all lines of code as well as the changes, use the following commands from your repository command-line tools.
Creating a Patch in CVS With All Lines of code

To create a CVS patch that shows all code (not just the changes and surrounding code), use this command:

```
cvs diff -N -U 10000 > patch.txt
```

The ‘10000’ number refers to the number of code lines included in the patch, before and after each change. 'Patch.txt' represents your desired name for the new patch file.

Creating a Patch in Subversion With All Lines of Code

To create a patch in Subversion that shows all code (not just the changes and surrounding code), use this command:

```
svn diff --diff-cmd diff -x "-U 10000" > patch.txt
```

- The in-built diff feature in `svn diff` does not support specifying lines of context, so you must tell Subversion to use an external diff command.
- The second "diff" in the command above needs to be the name of your external diff command. You might need to specify the full path to that command, such as `/usr/bin/diff`.
- On the Windows platform, you may need a Unix-like emulator such as Cygwin, and install the optional diff command for that.

Creating a Patch in Perforce With All Lines of Code

⚠️ Unfortunately, Perforce does not directly support creating patches that include all lines of code. A workaround is to checkout 'before' and 'after' versions of the file, and use GNU Diff to create a patch between the two files. That file could then be loaded into a Crucible review.

Creating a Review within Crucible

This page explains how to create a review from the Crucible interface.

On this page:

- Create a new review
- Choose where your review files will come from
- Add content to the review
- Edit the review details
- Adding an entire directory’s contents to a Crucible review

Create a new review

Within Crucible, create a review by opening the Tools menu at the top right of the Reviews screen, then clicking Create Review. You will be prompted to select the Project for the review (if you have multiple projects). Choose a project and click 'Create Review'.
Choose where your review files will come from

The 'Add Content to Review' screen appears, where you will now be prompted to choose where your review files will come from. Choose one of the options by clicking. See Selecting the Files for the Review for details.

Add content to the review

Once you select where your review files are coming from, you are prompted to select the files to be reviewed. Check the boxes next to any files you want to add.
Edit the review details

Once you have selected the files, click **Done** to go to the **Edit Review Details** screen, as shown below.

Screenshot: Editing review details

On the **Edit Review Details** screen, you can choose a title, reviewers, objectives, due date, linked reviews and issues. Once you’re finished, click ‘**Done**’.

Screenshot: Editing review details
The review will open in a preview form. Here, you can check all the details and click to edit any that aren’t correct. Once you click **Start Review**, the review is live.

**Adding an entire directory’s contents to a Crucible review**

To add an entire directory’s contents to a Crucible review, you will need to search to find all the files. For example, using "select revisions from dir /some/dir where is head", or similar logic.

* It is currently not possible in Crucible to add all the contents of a directory to a review with one click.

**Creating a review from FishEye**

This page explains how to create a Crucible review from FishEye.

---

**On this page:**

1. Open the FishEye Source view
2. Start the review
3. Choose a project

---

1. **Open the FishEye Source view**

To begin, the code author sets up the review. There are other ways to do this, but for this example, the author starts from the FishEye Source view of the file he or she wants to have reviewed.

**To navigate to the Source view for a file:**

1. On the **Source** tab, click the name for a repository.
2. Click the name of the file to be reviewed.
3. Click the **Source** sub-tab.
2. Start the review

To start the review:

- From the FishEye Source view, choose Reviews > Create Review.

Screenshot: Opening a review from the FishEye Source view

3. Choose a project

In the Edit Review Details dialog, choose the project for this review, and add reviewers. Now click Start Review.

Creating a review from JIRA

This page describes how to create a Crucible review directly from an issue in JIRA, the Atlassian issue-tracking application.

JIRA must be integrated with Crucible before you can do this. See JIRA Integration in Crucible for information on how to set this up.

To create a review from within JIRA:

1. Go to the issue in JIRA that relates to the work to be reviewed.
3. Either:
   a. Click the ‘+’ icon to the right of a changeset to create a new review for that changeset.
   b. Click Create crucible review to create a review for all the changesets in the list.
4. If a similar review already exists, you can add the changesets to that; otherwise click Create New Review.
5. In edit mode for a review:
• The content of the changeset becomes the content (i.e. files) to be reviewed.
• The author of the changeset becomes the **author** of the review, if Crucible is aware of this user. Otherwise the **creator** of the review becomes the author.
• The **creator** of the review becomes the **moderator**.
• The commit log message is used as both the Title and **Statement of Objective**.

6. Choose **Tools > Start Review** when you are ready.

The next step is to **add reviewers**.

**Screenshot: Adding a review from within JIRA**

---

**Screenshot: Review edit mode in Crucible**

---

**Creating a Review from a URL**

You can set up a URL which you can then click to create a Crucible review.
The format of your URL is as follows:

```
```

The parameters are as follows:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>csid</td>
<td>The changeset ID. You can specify one or more, of the form //repo/csid (where &quot;%2F&quot; is the URL-encoded form of is '/')</td>
<td>Yes</td>
</tr>
<tr>
<td>repo</td>
<td>The name of your repository.</td>
<td>Yes (unless supplied in the csid)</td>
</tr>
<tr>
<td>title</td>
<td>The title of your new Crucible review.</td>
<td>No</td>
</tr>
<tr>
<td>description</td>
<td>The description of your new Crucible review.</td>
<td>No</td>
</tr>
</tbody>
</table>

When you click the URL, you will be prompted to select the relevant projects, if more than one project exists) in which to create your review. A new draft review will then be created, including the following information:

- The content of the changeset becomes the content (i.e. files) to be reviewed.
- The author of the changeset becomes the author of the review, if Crucible is aware of this user. Otherwise the creator of the review becomes the author.
- The commit log message is used as both the Title (unless you have explicitly defined a title in your URL) and Statement of Objective.

All aspects of the review can be changed. To edit any of the above settings, click the title to see the 'Edit details' screen. Or you can click the Manage Files tab.

The next step is to add reviewers.

**Creating a Snippet Review**

This page explains how to create a simple code review using the Crucible Snippet Review feature. Snippet Reviews are designed to be lightweight ad-hoc code reviews.

To create a snippet review:

1. Copy the code to be reviewed from the source to your system clipboard.
2. Click **Create Snippet** in the Crucible toolbar.
3. Enter details for the snippet review:
   - Paste the code into the panel, where indicated.
   - Click on **Click to add title** near the top to enter a title for your review. If you don't specify a title, one will be automatically created for you.
   - Select a project from **Project**.
   - Select a programming language from **Syntax Highlighting**.
4. Click **Save** to create the snippet review.
5. Invite anyone that you want to participate in the snippet review by sending them the link to the review. The link is the review key, just above the review title. Anyone who is allowed to view the snippet is allowed to comment on it, and can close it.

6. Click **Reply** on any comments to respond.

7. Choose from the **Tools** menu to either close or delete the snippet review. Anyone can re-open, re-review or close snippet reviews. However, only the creator of a snippet review can delete it.

**Screenshots: Creating a Snippet Review**

Step 1

Step 2

Selecting the Files for the Review

This page explains how to select files/changesets that will be included in a Crucible review.

On this page:

- Selecting Changesets for Review
- Selecting Files for Review by Exploring Repositories
- Selecting Files for Review using the Crucible Search
- Using the Suggestions Feature When Adding Files to a Review
- Adding Pre-commit Patch Files to a Review
- Adding Attachments to a Review

To add content to a review:

1. Log in to FishEye/Crucible and either;
   - Create a review, as described on [Creating a Review](#), or
   - Open an existing review, which you are the **creator** or **moderator** of, and click the Add Content (➕) button.

2. The 'Add Content to Review' dialogue will be displayed (see screenshot below). Select the method you want to use, to find the content for your review:
   - **'Browse Changesets'** — Allows you to choose changesets from a Source Code Management (SCM) repository to add to a review. See [Selecting Changesets for Review](#).
   - **'Explore Repositories'** — Allows you to browse for files from a Source Code Management (SCM) repository to add to a review. This option only appears when FishEye is installed. See [Selecting Files for Review using by Exploring Repositories](#).
   - **'Search for Files'** — Allows you to search a Source Code Management (SCM) repository for files
or changesets to add to a review. This option only appears when FishEye is installed. See Selecting Files for Review using the Crucible Search.

- ‘Suggest Files’ — Analyses the list of files in the current review and makes suggestions based on certain logic (for example, suggesting a newer version of a file if one exists). This option only appears when FishEye is installed. See Using the Suggestions Feature When Adding Files to a Review.
- ‘Pre-commit’ — Allows you to upload patch files to a review. See Adding Pre-commit Patch Files to a Review.
- ‘Attachments’ — Allows you to upload any file to a review, including binary files and files outside of a Source Code Management (SCM) repository. See Adding Attachments to a Review.

3. Follow the appropriate instructions in the sections below to add content to your review.
4. Click the ‘Done’ button to finish adding content to the review.

Screenshot: The Add Content menu for Crucible reviews

Selecting Changesets for Review

Click the ‘Browse Changesets’ option on the ‘Add Content’ dialogue to add changesets to your review.

Screenshot: The Browse Changesets View in the Add Content dialogue
By default, Crucible presents a list of the author's changesets in reverse chronological order. You can see other changesets by changing the options at the top of the screen.

Click the checkbox next to a changeset ID to add the entire changeset. Note,

- You cannot add individual file revisions to a review, although you can remove them once the changeset is added. Click ‘Remove all revisions from review’ to remove all.
- You cannot add changesets that are entirely svnprops changes (i.e. it has no non-metadata changes). For details, see How do I force reviews to include SVN property changes?.

Options for adding changesets:

- ‘Repository’ — This is a list of the repositories that contain the files that can be reviewed. If the repository you require is not in the list then it has not been added to FishEye. Please contact your Crucible/FishEye administrator.
- ‘Author’ — This contains a list of all the authors who have made changes within the repository. When creating a review, this will default if possible to the username of the user authoring this review and will therefore show their changsets.
- ‘Branch’ — This will only show files and recent changes on that branch from the repository set above.
- ‘Tag’ — This will only show files and recent changes tagged.
- ‘Add to Review As’ — Choose the form of the review. See Choosing the way files are added to the review below.
- ‘Go to Changeset’ — Allows you to jump to a particular change set by entering its title and pressing Enter.

Choosing the way files are added to the review:

When adding files to a review, you can set the form of review taking place in the ‘Add to Review as’ drop-down menu:

- ‘Whole Files’ — Adds the entire file with all content, rather than just a diff with context.
- ‘Diffs’ — This is the default behaviour. This allows you to add multiple revisions of a file to one review and compare them in-review, in context with the change history.
- ‘Diffs to Last Reviewed Version’ — This adds files with a diff to the last reviewed changeset.
- ‘Diffs to... (a particular revision)’ — This allows you to specify the file to show the differences between two specific versions of a file.
- ‘Diffs to Last Branch Point’ — This adds files with a diff to the revision each file was last branched.
Click the 'Done' button once you have finished selecting the desired files. The files will be added to your review and your review will be displayed.

**Selecting Files for Review by Exploring Repositories**

Click the 'Explore Repositories' option on the 'Add Content' dialogue to browse repositories for files to add to your review.

**Screenshot: Browsing for files to add to a review**

- To find a file, browse the folders by clicking the relevant folder. The folders by default are sorted by path name but can sorted by last-commit or first-commit.
- To choose a file for reviewing, click the checkbox to the left of the filename and if required the revision number.
- To select a particular revision of a file, open the revision number list and select the option "Load Full History...". This will refresh the available options in the list.

Please note the following information when browsing files to add to a review:

- Empty folders will be greyed out.
- If the folders contain empty folders then a toggle option called 'Hide Empty' will appear under the 'Sort' options.
- To see or ignore deleted files, you can click the 'Hide' and 'Show' options located above the file names on the left.

**Choosing the way files are added to the review:**

When adding files to a review, you can set the form of review taking place in the 'Add to Review as' drop-down menu:

- 'Whole Files' — Adds the entire file with all content, rather than just a diff with context.
- 'Diffs' — This is the default behaviour. This allows you to add multiple revisions of a file to one review and compare them in-review, in context with the change history.
- 'Diffs to Last Reviewed Version' — This adds files with a diff to the last reviewed changeset.
- 'Diffs to... (a particular revision)' — This allows you to specify the file to show the differences between two specific versions of a file.
- 'Diffs to Last Branch Point' — This adds files with a diff to the revision each file was last branched.

Click the 'Done' button once you have finished selecting the desired files. The files will be added to your review and your review will be displayed.

Selecting Files for Review using the Crucible Search

Click the 'Search for Files' option on the 'Add Content' dialogue to use the Crucible search to find files to add to your review.⚠️ The 'Search' view is only available when using FishEye with Crucible.

Screenshot: Searching for files to add to a review

If you are not certain about which changesets/revisions/files to include in a review, use the Search view to find them. Adjust the search filters to find the files you need. If the simple filters are not enough, read about EyeQL queries in the FishEye documentation.

Read the FishEye documentation for more information about the searching your repositories.

Choosing the way files are added to the review:

When adding files to a review, you can set the form of review taking place in the 'Add to Review as' drop-down menu:

- 'Whole Files' — Adds the entire file with all content, rather than just a diff with context.
- 'Diffs' — This is the default behaviour. This allows you to add multiple revisions of a file to one review and compare them in-review, in context with the change history.
- 'Diffs to Last Reviewed Version' — This adds files with a diff to the last reviewed changeset.
- 'Diffs to... (a particular revision)' — This allows you to specify the file to show the differences between two specific versions of a file.
- 'Diffs to Last Branch Point' — This adds files with a diff to the revision each file was last branched.

Click the 'Done' button once you have finished selecting the desired files. The files will be added to your review and your review will be displayed.
Using the Suggestions Feature When Adding Files to a Review

Click the 'Suggest Files' option on the 'Add Content' dialogue to view and add files suggested by Crucible to your review. You need to have already added some file(s) to review for Crucible to suggest additional files.

Images: Viewing and adding file suggestions to a review

Crucible can make intelligent suggestions when you are creating a review. The Suggestions feature logic is based around the following:

- **Most recent versions**: If a newer version of a file exists, Crucible will suggest that you add it to the review.
- **Similar files**: Files with a similar filename may be of relevance to your review; Crucible will show them to you.

Choosing the way files are added to the review:

When adding files to a review, you can set the form of review taking place in the 'Add to Review as' drop-down menu:

- **'Whole Files'** — Adds the entire file with all content, rather than just a diff with context.
- **'Diffs'** — This is the default behaviour. This allows you to add multiple revisions of a file to one review and compare them in-review, in context with the change history.
- **'Diffs to Last Reviewed Version'** — This adds files with a diff to the last reviewed changeset.
- **'Diffs to... (a particular revision)'** — This allows you to specify the file to show the differences between two specific versions of a file.
- **'Diffs to Last Branch Point'** — This adds files with a diff to the revision each file was last branched.

Click the 'Done' button once you have finished selecting the desired files. The files will be added to your review and your review will be displayed.

Adding Pre-commit Patch Files to a Review

Click the 'Pre-commit' option on the 'Add Content' dialogue to add pre-commit patch files to your review.

Screenshot: Adding pre-commit patch files to a review
For a full explanation of the ‘Patch View’ functions, read about creating a patch review.

Click the ‘Done’ button once you have finished selecting the desired files. The files will be added to your review and your review will be displayed.

Adding Attachments to a Review

Click the ‘Attachments’ option on the ‘Add Content’ dialogue to add attachments to your review.

Screenshot: Adding attachments to a review

You can upload additional files to be used in the review, including binary files, images or code files that are not stored in a version control repository. The ‘Upload’ view contains various controls to help you do this. These are listed below.

Choose the ‘Upload Method’ as either ‘Select file from the file system’ or ‘Paste text from clipboard’:

- Displays if ‘Upload method’ is ‘Select file from the file system’:
  - ‘Character Set’ (if any) — Click the edit icon (📝) to choose the character set being used. ‘US-ASCII’ is the default.
  - ‘File’ — Click ‘Browse’ to find the file that you want to add to the review.
- Displays if ‘Upload method’ is ‘Paste text from clipboard’:
  - ‘Patch text’ — Paste your copied text in this text area.

Click the ‘Upload’ button, when you have made your selection. Once complete, a list of uploaded files is
displayed at the bottom of the screen. To add another iteration of a file, make changes to the file and upload it again with the same filename. It will be added as a new version.

Click the 'Done' button once you have finished selecting the desired files. The files will be added to your review and your review will be displayed.

**Iterative Reviews**

Crucible allows you to review several revisions of a file within one review, seamlessly switching between them. Comments are linked and relative to a specific revision. This allows you to review every change that has occurred on a code file within a given period of time. This lets you see the evolution of the file through various revisions (within one Crucible review).

**Screenshot: Iterative Reviews**

Adding Reviewers

This page explains how to add reviewers to a new review, after it has been created. See [Creating a Review](#) for information about creating reviews.

**On this page:**

- **Entering Basic Information**
- **Adding Reviewers**
  - Adding Users to a Review
  - Crucible Suggests Reviewers
  - Inviting Non-Registered Users to the Review
  - Checking the Draft and Starting the Review
Next Steps

Entering Basic Information

Once a review has been created, the Edit Review dialog opens.

Screenshot: The Edit Review dialog

In the Edit Review dialog, the author enters information needed for the review. This includes entering a title and description for the review, a due date and the key for a related JIRA issue (if any). The project, moderator and author are pre-selected (for this example, the author should select himself as a moderator.

You must also select reviewers.

Adding Reviewers

Before a review can be issued to reviewers, you must decide who can review it. When adding reviewers, you can add registered users immediately. The usernames will auto-complete, showing partial matches before you finish typing. You can quickly select one of the matches shown with the keyboard arrow keys, pressing Enter or Tab to add them to the review.

In addition, you can easily invite external users who do not yet have accounts in Crucible to take part by typing their email address into the **Reviewers** field.

Adding Users to a Review

Select users by typing names into the text field under **Reviewers**. Crucible will show a list of matches. Press Enter to select one after each entry.

Clicking the ‘Save’ button will save the review as a draft for later issue.

You can also decide to allow any registered user to add themselves as a reviewer in the review. To enable this option, put a check next to ‘Allow anyone to join’.

Crucible Suggests Reviewers

Crucible will automatically suggest reviewers, by analysing the users that have contributed to the files you’ve selected and also don’t have a lot of open reviews. You can easily pick reviewers from the list of suggestions by
Inviting Non-Registered Users to the Review

You can invite users who don't have a Crucible account to join a review.

There are two prerequisites:

1. FishEye’s SMTP server must be configured and capable of sending email.
2. The setting 'Built-in Public Sign-up' must be set to 'ON'. This setting can be accessed by opening the 'Admin Menu', then clicking 'Security' under 'Global Settings' on the left navigation bar.

To invite an external user to a review:

- a. Create a new review.
- b. On the 'Create New Review' screen, simply type the user's email address into the 'Reviewers' field, then press Enter to select.
- c. Click Save to save the draft review. The users are not sent any information at this time.
- d. When you click 'Start Review', this is when all email invites and notifications are sent out.
- e. The external user will receive an email address from the Crucible server, containing a special URL that they can visit.
- f. When the user loads the URL they received via email, they are taken to a special Crucible log in screen. On this screen, the user can create a new account that will be linked to the current email address. (If they already have a Crucible account under another address, they can simply sign-in with that username and password.)
- g. When the user has successfully created a Crucible account, they will be able to access the review(s) associated with their email address and take part.

You can enter multiple addresses separated by commas, allowing you to paste in a list of email addresses from your favourite email application.

When finished, the author clicks 'Save'. The review will now be created in a draft form.

Checking the Draft and Starting the Review

The draft review opens. In the draft stage, the author can check the contents of the review files to ensure they are correct and put in any notes for reviewers as comments. During the draft phase, no notification email is sent out to reviewers. Once the author is finished with the draft phase, he clicks 'Start Review'.

The review will now be started and notification email will go out to all participants. Crucible will now send out an email notification to all the participants. This lets them know that the review is under way and prompts them to take action, providing a URL for direct access to the review. (You can also subscribe to an RSS feed.)

Screenshot: A newly created Crucible review
Next Steps

You can now begin Performing the Review, If you have a moderator controlling your review process, you can move onto Issuing a Review.

Issuing a Review

This page contains information about starting a review in Crucible.

On this page:

- Starting a review
- Editing review details once started

Starting a review

Issuing a review simply means formally starting it and inviting people to take part.

Once you have selected the reviewers, the next stage is to notify the reviewers and the author (if different to the moderator) that they can start reviewing. The review has been in ‘Draft’ state until this point. Only the moderator has the permission to start a review.

To start the review:

- If you are the moderator of the review, click the ‘Start Review’ button. Or;
- If you are not the moderator of your review, click ‘Send to Moderator’. This changes the state to ‘Require Approval’ and notifies the moderator. The moderator can change any aspect of the review before starting it.
Once the review has been started, the review state becomes ‘Under Review’.

Screenshot: Starting a Review

Note that only people with the ‘Approve’ permission can start a review.

Editing review details once started

You can edit the details of a review at any time by simply clicking the ‘Edit Review’ button in the left navigation bar to launch ‘Edit Mode’. In Edit mode, you can quickly click red cross icons to remove files from the review. A single click returns you to regular Crucible functions, so you can more easily tune the content inside your reviews. Another button opens a dialog for rapidly adding more content to the review.

Screenshot: Launching Edit Mode

Screenshot: Crucible Edit Mode for Review Content
Performing the Review

This page describes how to find and manage the Crucible reviews that relate to you.

On this page:

- Browse Your Reviews Under the 'Dashboard' Tab
- Browse All Reviews Under the 'Reviews' Tab
- When Files Change During a Review
- Next Steps

Deciding what needs to be reviewed

The 'Statement of Objective' is a brief description of what the review is intended to achieve. Crucible does not dictate how or what to review. It simply provides a mechanism to record comments.

Browse Your Reviews Under the 'Dashboard' Tab

When you first load Crucible, the 'Dashboard' screen will load, which shows your current reviews and other activity related to you.

Use the Crucible 'Dashboard' to manage your reviews. Read the overview on filtering your view.

Active reviews are listed on each reviewer's dashboard under the default 'To Review' filter. Reviews are listed under 'Out for Review' until all reviewers indicate they are complete. Then the reviews move to the 'To Summarize' list.

Read more about using the Dashboard tab.

Browse All Reviews Under the 'Reviews' Tab

All reviews that involve you in any role are listed when you click 'Open' or 'Closed' in the left navigation bar. For instance, use the 'My Reviews' --> 'Open' filter to locate a review that doesn't require further action from you, but
is still under way.

If email notifications are enabled (see SMTP settings in the FishEye documentation), reviewers will receive an email with information about the review. Click the link within the email to go directly to the review.

**When Files Change During a Review**

If a file in the repository changes during a review, Crucible will visually alert you by showing the 'File Outdated' menu:

**Screenshot: Visual Cue for Updated Repository Files**

![Visual Cue for Updated Repository Files](image)

From the 'File Outdated' menu, you can choose to view the latest revision of the updated file, or add the latest revision to the review:

**Screenshot: 'File Outdated' menu**

From the 'File Outdated' menu, you can choose to view the latest revision of the updated file, or add the latest revision to the review:

**Next Steps**

- Adding Comments
- Flagging Defects
- Completing your Review
- Sending a Review's Comments via Email
- Using JIRA Integration in Crucible Reviews
- Using the Review History Dialog
- Tracking Crucible Review Metrics

**Adding Comments**

Comments can be added at the level of a review, revision, or line. You can also reply to a comment.

On this page:

- Locating existing comments
- Adding a Comment
- Draft Comments

Locating existing comments

The number shown next to a filename, in the left-hand column of the screen, indicates the number of comments
that apply to that file.

(The number of unread comments, if there are any, is shown in brackets.)

### Screenshot: Comments

#### Adding a Comment

- To add a comment that applies to the whole review, select the **Review Comments** line in the left-hand navigation panel, then click the following icon: 

- To add a comment that applies to a revision/change, select the filename in the left-hand navigation panel, then click the following icon: 

- To add a source-level comment, expand the source view then click a line of code. You can click and drag to select multiple lines from one revision or diff, or click individual lines to select/deselect them. The comment will appear in the source at the last line selected. Hover over the comment to highlight the selected lines.
  - To select text on the page without adding a comment, hold down the Alt button while dragging the cursor.
To reply to a comment, click the ‘Reply’ link at the bottom of the comment.

⚠ Only people with the ‘Comment’ permission can add comments.

💡 Read about flagging defects too.

**Screenshot: Adding a Comment**

---

**Draft Comments**

You can save your comment as a draft and then edit it later. When you complete the review, you will be prompted to post, discard or edit any remaining draft comments.

**Screenshot: Draft comments**

---

**Flagging Defects**

Comments in Crucible can be used to flag a defect in the code under review.

to do this, simply tick the ‘Defect’ box when adding a comment and select a category from the drop-down list.

**Screenshot: Defects**

---

You may want to mark comments as defects to associate defect classifications, or simply to highlight to the author or moderator that the issue you raised in your comment requires attention.
Crucible intentionally does not mandate how defects are to be used. The Crucible administrator can customize the defect classifications.

You can only use the defect classifications on comments that are not a reply to an existing comment.

**Completing your Review**

Once each reviewer has added comments to the review and has nothing further to add, the next step is to complete their individual review.

To complete your individual review, go to the review and click the 'Complete' button at the right of the screen, next to the 'Tools' menu:

*Screenshot: The Complete Button*

Only people with the 'Complete' permission can complete a review.

This notifies the moderator (via email if configured) that you have completed your review.

Reviewers can still continue to add comments until the moderator summarises the review. The moderator does not have to wait for all reviewers to complete their individual reviews before summarising.

If you have any draft comments, you will be prompted to post/discard/edit any comments before completing the review.

*Screenshot: Draft comments*

---

**Warning**

**You have draft comments**

Draft comments that aren't posted will be deleted.

*View drafts*
*Delete drafts*
*Post drafts*

*Screenshot: Review complete*
Sending a Review's Comments via Email

You can send all of the comments from a review to anyone you want via email. You may wish to do this to allow a person outside the review to quickly scan the content of the comments, or oversee the review activity. Alternatively, you may wish to send all participants this information to let them read the current status of the review and its comments in full.

**Before you begin:**

- Users that are not logged in cannot send email, but instead can view the text content of the review's comments by clicking the 'View Text' option which will appear instead of 'Email Review'.
- Emails of reviews are only sent in plain text, not HTML. HTML emails are only available via watches.

**To send all of a review's comments via email:**

1. In Crucible, navigate to the review in question.
2. From the 'Tools' menu, select 'Email Review' (see Screenshot 1 below).
3. The 'Recipients' page appears (see Screenshot 2 below). On that page:
   - In the 'To:' field you can enter multiple email addresses, separated by commas.
   - You can also tick the 'Send to Review Participants' check-box to include all of the review's reviewers.
4. When you have finished the distribution list, click the 'Next' button.
5. The 'Recipients' page appears (see Screenshot 3 below). On that page:
   - In the 'To:' field you can enter multiple email addresses, separated by commas.
6. When you have finished your message, click the 'Send' button.
7. The 'Status' page appears (see Screenshot 4 below), confirming that your email has been sent.

*Screenshot 1: The 'Email Review' option in Crucible*
Screenshot 2: The ‘Recipients’ Screen in Crucible

Screenshot 3: The ‘Message’ Screen in Crucible
Using JIRA Integration in Crucible Reviews

This page contains information on how to use JIRA integration with Crucible reviews.

On this page:

1. Create a JIRA Issue for your Review
2. Create Your Review and Link it to a JIRA Issue
3. Make a Comment or Defect on the Review
4. Click to create a JIRA Sub-Task
5. Resolve the JIRA issue through Crucible

Before you begin, both your Crucible and JIRA instances must be configured to use make use of these JIRA integration features. The Crucible project requires a linked JIRA project before issues can be linked to reviews.
1. Create a JIRA Issue for your Review

To use JIRA integration, you must begin with a JIRA issue that you will use as the parent issue for the review. Crucible will create and resolve sub-tasks belonging to this parent issue. Once your parent issue is created, make a note of its issue key, e.g. FE-1968.

2. Create Your Review and Link it to a JIRA Issue

When creating your review, you have an option called ‘Linked Issue’. Crucible may put a suggested JIRA issue key into this field automatically if a JIRA key is found in the review title. You can specify a different issue key and click ‘Link’ to save it. You can also click the ‘x’ to clear the field and load a different issue key.

3. Make a Comment or Defect on the Review

Once your review has a linked JIRA issue, create a comment or defect comment anywhere on the review. Once created, the comment actions will show a link titled ‘Create Issue’ (note that this link does not appear on replies — only on new comments). You can click that to instantly create a sub-task under the parent JIRA, which will take the content of the comment as its summary.
4. Click to create a JIRA Sub-Task

Clicking the 'Create Issue' link will allow you to create a JIRA sub-task under the parent JIRA issue, e.g.:

Screenshot: The JIRA 'Create Issue' dialog

- The list of possible assignees will include:
  - 'Automatic' (i.e. the default assignee for that JIRA project)
  - the assignee of the subtask's parent issue
  - the reporter of the subtask's parent issue
  - 'Unassigned' (if your JIRA administrator has enabled Allow Unassigned)
  - plus, if Trusted Applications have been configured between JIRA and Crucible,
    - the review author
    - the review moderator
    - the comment/defect author
    - yourself

Once created, the sub-task JIRA issue key, status and default action (i.e. 'Resolve') will be shown. If you hover
1. Users are mapped to their own accounts when using Trusted Applications. If you don't have the permissions to carry out the default action (‘Resolve’, in this case), an error will occur.

5. Resolve the JIRA issue through Crucible

Once the work required on your sub-task is completed, simply click the action link provided to signal that this has occurred (e.g. ‘Resolve’). The JIRA issue will be closed.

If you encounter problems or have trouble using JIRA integration, please read the FAQ page on this topic.

Using the Review History Dialog

The Review History dialog shows a chronological list of interactions within a review. You can see rich information about those interactions and control their display. You can sort the information by date, actor, or action.

To open the Review History dialog:

1. Open a review in Crucible.
2. Click ‘Tools’ > ‘Review History’ at the toolbar at the top right corner of the screen.
3. The Review History dialog opens.

This information can also be displayed in the new timeline mode, a graphical visualisation that shows events on a horizontal graph over time (click the ‘Timeline’ tab at the top of the dialog to switch from ‘Details’ to timeline view). Click and drag inside the timeline view to scroll the graph left and right. You can also click on the section showing months to scroll over a greater time scale.

Additionally, you can get access to the entire review history through the ‘CSV export’ link in the upper right hand
corner, allowing for easy data import into a spreadsheet or other application.

Screenshot: The Crucible Review History Dialog

Screenshot: The Crucible Review History Dialog, Timeline View

Tracking Crucible Review Metrics

Crucible tracks each participant’s percentage completion through each review and the total time they have spent.

To learn about these features, see the following pages:

- Using Progress Tracking
- Using Time Tracking

Using Progress Tracking

This page contains instruction on how to use progress tracking in Crucible.

On this page:

- How progress tracking works in Crucible
- Viewing the progress tracking totals
How to adjust progress tracking on a review

Adjusting settings for progress tracking

Further reading

How progress tracking works in Crucible

As you work your way through the files in a review, Crucible tracks the ones you have viewed. Whenever you open a file for review, Crucible will automatically mark it as read.

When participating in iterative reviews, progress tracking also takes lines of code and revisions into account.

Viewing the progress tracking totals

The 'Details' view shows a summary of the progress of each participant through the files in the review.

If there is only one file in the review, then the progress tracked will either show 0% or 100%.

Screenshot: Viewing the Progress Tracking Totals

How to adjust progress tracking on a review

You can mark a file as unread by clicking on its name to view the file's contents. In the source view, you have an option at the top left of the screen, 'Leave Unread'. If you select that, then the file you are looking at will not be added to your progress percentage.

Screenshot: Marking a File as Unread

Adjusting settings for progress tracking

Progress tracking is a configurable user preference that can be changed in the 'User Settings' menu, in the 'Reviews' sub-section. 'Auto-mark files as read' is on by default. When 'Auto-mark files as read' is set to 'No', marking files as read or unread is left to the user to manually manage.

Screenshot: Adjusting the Progress Tracking User Settings
Further reading

You may also want to learn about Crucible's Time Tracking feature.

Using Time Tracking

This page contains instruction on how to use time tracking in Crucible.

On this page:

- How time tracking works in Crucible
- How to adjust the time tracked on a review
- Viewing the time tracking totals
- JIRA integration
- Further reading

How time tracking works in Crucible

Crucible will automatically track the time you spend in a Crucible review. When you open a file for review, a counter in the Review Details panel starts. The time is added to your total when you leave the review screen.

Screenshot: Crucible Time Tracking

How to adjust the time tracked on a review

You can click and type in the time tracking control to adjust the time you have spent during the session.

Viewing the time tracking totals

The 'Details' view shows a summary of the progress and time tracked on each file.

Screenshot: Crucible Tracking Totals
JIRA integration

Using Crucible when integrated with JIRA, you can update time tracking from the following locations:

- The confirmation dialog for a reviewer completing a review,
- The confirmation dialog on closing a review,
- The regular toolbar location in Crucible.

Screenshot: JIRA Time Tracking Integration

Further reading

You may also want to learn about Crucible’s Progress Tracking feature.

Summarising and Closing the Review

Summarize is an optional step before closing a review.

To enable or disable the Summarize step, you will have to configure the permission in your Permission Scheme. Crucible ships with two permission schemes:

- 'Agile' - the summarize step is disabled for all users
- 'Default' - the summarize step is enabled for the moderator

You can choose to either summarize a review or close a review at any time, given that your Permission Scheme allows it. You can skip the summarize step by directly clicking the ‘Close’ button.

Normally, we recommend that you wait for all reviewers to complete their reviews. The reviews that the reviewers have completed will be in your Ready to Close menu on the Dashboard.

To summarize a review,

- Click Summarize at the right of the screen.
- Optionally enter a summary of the review.
- If you have no further comments to add, click Close Review; otherwise, click Continue Without Closing.

Screenshot: ‘Summarize’ button
The above review is not yet complete

We can see that Geoff Crain has still not finished reviewing, because there is no green tick next to his name.

On clicking Summarize, the user may be prompted to confirm the action if there are incomplete reviews or draft comments in the review.

The requests for confirmation are warnings only

The review can still be summarized and closed.

Once the review is in the ‘Summarize’ state, the moderator can optionally add a review summary, i.e. describe the outcomes/tasks/etc.

Screenshot: Summarize Review

Summarize Review

Summarize the review outcomes (optional)
Testing successful. Thanks for your time, everyone.

Screenshot: Review Closed
The summary is sent to all participants and displayed at the top of the closed review.

- Reviews in the 'Review' or 'Summarize' state can be closed.
- Reviews in the 'Summarize' or 'Closed' state can be re-opened. Re-opening changes the review's state back to 'Under Review', allowing all participants to add comments.

⚠️ Re-opening a review is not the recommended way to 're-review'. You should create a new review with the reworked changes and link it to its parent review (create a hyperlink back to the original review in the new Review's Objectives field).

Note that you need the 'Summarize', 'Close' or 'Re-Open' permission to summarize, close or re-open a review.

**Moving a Review to Another Project**

You can move reviews between projects once they have been created.

**To move a review between projects:**

1. Open the review. Click the 'Edit Review' button at the top of the screen.
2. The 'Edit Review' window will open, allowing you to change various aspects of the review.
3. Under 'Project' click the drop-down menu. This will allow you to select a new parent project for the review.
4. Click the 'Done' button at the bottom of the screen.

*Screenshot: Changing a Review's Parent Project*
Deleting a review

To delete a review you must first abandon the review. To do that, follow the instructions below.

⚠️ Deleted reviews cannot be retrieved.

**Related page:**
- [Deleting hung reviews manually](#)

To abandon and then delete a review:

1. Open the review.
2. Choose **Tools > Abandon**.
3. Now, on the Crucible dashboard, click **My Abandoned Reviews** in the left-hand navigation bar.
4. In the list of abandoned reviews, click the name of the review you wish to remove.
5. Once the review details are displaying, choose **Tools > Delete**. The review will be instantly deleted.

*Screenshot: Deleting a review in Crucible*
Defining your Workflow

This document describes several forms of Crucible Workflow in detail. Depending on the size of your team, there are four different ways that a development team could use Crucible for code reviews. Choose the workflow which suits your team.

- **Lightweight Code Commenting with Crucible (individual)**
- **One-to-One Reviews (Agile Pair)**
- **One-to-Many Reviews Without a Moderator (Agile Team)**
- **Formal Group Reviews (CMM Team)**

**Lightweight Code Commenting with Crucible (individual)**

1. **Author** commits new work.
2. Author creates the review, and adds comments using the easy web interface.
3. Author summarizes and closes the review, saving the code comments in Crucible's database, which is stored outside the repository.

*Diagram: Workflow for Lightweight Code Commenting*

**One-to-One Reviews (Agile Pair)**

1. **Author** creates the review.
2. Author invites reviewer to take part in the review.
3. **Reviewer** creates comments on the code.
4. Author responds to reviewer comments.
5. Follow-up comments are made if necessary.
6. Reviewer finishes own review process.
7. Author summarizes and closes the review.

*Diagram: Workflow for One-to-One Reviews*
For more information on one-to-one reviews, see The Crucible workflow. The workflow process in Crucible is covered in detail within this document.

One-to-Many Reviews Without a Moderator (Agile Team)

1. **Author** creates the review.
2. Author invites **reviewers** to take part in the review.
3. Reviewers make comments on the code.
4. Author responds to reviewer comments, follow-up comments are made if necessary.
5. Reviewers complete their reviews.
6. Author summarizes and closes the review.

*Diagram: Workflow for One-to-Many Reviews*

Formal Group Reviews (CMM Team)

1. **Author** creates the review.
2. **Moderator** invites **reviewers** to take part the review.
3. Reviewers make comments on the code.
4. Author responds to reviewer comments.
5. Follow-up comments are made if necessary.
6. Each discussion point is settled by the Moderator.
7. Moderator summarizes and closes the review.

*Diagram: Workflow for Formal Group Reviews*
To see a simple example of how to use Crucible with two people, see The Crucible workflow.

Viewing Your Favourites

This page contains instructions on how to view, rename and remove your existing favourites in Crucible. You can select code reviews, changesets, files, people and repositories as favourites in Crucible. This allows you to personalise the information that you see in your activity stream.

See Using Favourites for instructions on how to add new favourites.

On this page:

- Viewing your Existing Favourites
- Renaming a Favourite
- Removing a Favourite

Viewing your Existing Favourites

To view your favourites,

1. Click the favourites icon (⭐) in the header. The code reviews, changesets, files, people and repositories that you have selected will be displayed in a list on the dashboard (see screenshot).

2. Click the favourites icon (⭐) next to a favourite to show the dialogue for renaming (i.e. changing the 'Label') the favourite or removing the favourite. See Using Favourites for more information.

✔ Tip: You can add favourites wherever you see star icons next to code reviews, changesets, files, people and repositories, not just on this screen.

Screenshot: Viewing favourites
Renaming a Favourite

To rename a favourite,

1. Click the favourites icon (⭐) in the header. The code reviews, changesets, files, people and repositories that you have selected will be displayed in a list on the dashboard (see screenshot).
2. Click the favourites icon (⭐) next to a favourite to show the 'Update favourite' dialogue (see screenshot below).
3. Enter the new name for the favourite in the 'Name' field and click the 'Save label' button. The favourite will be renamed on the dashboard.

Screenshot: The 'Update favourite' dialogue

Removing a Favourite

To view your favourites,
1. Click the favourites icon (★) in the header.
   The code reviews, changesets, files, people and repositories that you have selected will be displayed in a list on the dashboard (see screenshot).

2. Click the favourites icon (★) next to a favourite to show the 'Update favourite' dialogue (see screenshot below).

3. Click the 'Remove' button. The code review, changeset, file, person or repository will be removed as your favourite (the icon will be greyed out) and will not display the next time you view your list of favourites.

   Tip: You can remove favourites wherever you see star icons next to code reviews, changesets, files, people and repositories, not just on this screen, not just on this screen.

Screenshot: The 'Update favourite' dialogue

Using Favourites

This page contains instructions on adding favourites in Crucible. You can select code reviews, changesets, files, people and repositories to be added to your favourites. This allows you to personalise the information that you see in your activity stream. We suggest you select items that you are currently working on as favourites, to create a more relevant personalised view.

See Viewing Your Favourites for instructions on how to view your existing list of favourites and how to rename and remove favourites.

On this page:

- Adding a Review to Your Favourites
- Adding a Review Comment Thread to Your Favourites
- Adding a Project to Your Favourites
- Adding a Person to Your Favourites
- Adding a Changeset to Your Favourites
- Adding a File or Folder to Your Favourites
- Adding a Repository to Your Favourites

Adding a Review to Your Favourites

To add a review to your favourites, hold the mouse cursor over the review name when it appears in a menu screen. The Review Hover menu appears. At the top right of the Review Hover menu, click the small grey cog icon that indicates the 'Tools' menu. The Tools menu opens. In the Tools menu, click 'Add Star'. This will add it to your favourites.

Screenshot: Adding a Review To Your Favourites
Adding a Review Comment Thread to Your Favourites

To add a review comment thread to your favourites, click the link 'Add Star' next to the grey star icon at the bottom of the first comment of the thread. From then on, new comments will be shown in your favourites activity stream.

Screenshot: Adding a Review Comment Thread to Your Favourites

Adding a Project to Your Favourites

To add a project to your favourites, click the 'Projects' tab. The Projects view appears. Here, simply click the grey star icon that appears next to the desired project name. The star icon will turn yellow, showing that it is selected as a favourite.
Adding a Person to Your Favourites

To add a person to your favourites, simply hold the mouse cursor over their username wherever it appears. The User Hover menu will appear. In the User Hover menu, click ‘Follow’. This will add the person to your favourites.

Adding a Changeset to Your Favourites

To add a changeset to your favourites, firstly open the changeset desired from the ‘Source’ tab. Once the changeset is open in Crucible, simply click the grey star icon that appears next to its name. The name appears in the breadcrumb links at the top of the screen.
Adding a File or Folder to Your Favourites

To add a file to your favourites, firstly open the file or folder desired, from the ‘Source’ tab. Once the file or folder is open in Crucible, simply click the grey star icon that appears next to its name. The name appears in the breadcrumb links at the top of the screen.

Screenshot: Adding a File or Folder to Your Favourites

Adding a Repository to Your Favourites

Adding a repository to your favourites (requires FishEye), click the ‘Source’ tab. The the ‘Source’ view appears. Here, simply click the grey star icon that appears next to the name of the desired repository. The star icon will turn yellow, showing that it is selected.

Screenshot: Adding a Repository to Your Favourites

Using Keyboard Shortcuts in Crucible

Crucible provides a number of keyboard shortcuts, allowing you to quickly carry out certain actions without the mouse. Keyboard shortcuts are available for most of the commonly-used functions in Crucible.

To see a list of available shortcuts, firstly navigate to a review in Crucible. Now open the ‘Tools’ drop-down menu at the top right corner of the screen, and select the ‘Keyboard Shortcuts’ option.

See the tables below for full details:

**General Shortcuts**

<table>
<thead>
<tr>
<th>Key</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>Opens reference list of keyboard shortcuts.</td>
</tr>
<tr>
<td>escape</td>
<td>Closes reference list of keyboard shortcuts.</td>
</tr>
<tr>
<td>alt</td>
<td>Hold down then click and drag to select source line contents.</td>
</tr>
<tr>
<td>shift + f</td>
<td>Toggle full screen review mode.</td>
</tr>
</tbody>
</table>

Custom Navigation
<table>
<thead>
<tr>
<th>Key</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>,</td>
<td>(Comma) Go to the previous element, which can be a file, comment or defect, depending on your current context.</td>
</tr>
<tr>
<td>.</td>
<td>(Period) Go to the next element, which can be a file, comment or defect, depending on your current context.</td>
</tr>
</tbody>
</table>

**Comment Navigation Shortcuts**

<table>
<thead>
<tr>
<th>Key</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Go to next review comment.</td>
</tr>
<tr>
<td>p</td>
<td>Go to previous review comment.</td>
</tr>
<tr>
<td>shift + p</td>
<td>Go to first review comment.</td>
</tr>
<tr>
<td>shift + n</td>
<td>Go to last review comment.</td>
</tr>
<tr>
<td>l</td>
<td>Go to next thread (skips replies).</td>
</tr>
<tr>
<td>h</td>
<td>Go to previous thread (skips replies).</td>
</tr>
<tr>
<td>]</td>
<td>Go to next unread comment.</td>
</tr>
<tr>
<td>[</td>
<td>Go to previous unread comment.</td>
</tr>
<tr>
<td>r</td>
<td>Reply to a comment.</td>
</tr>
<tr>
<td>m</td>
<td>Toggle comment read/unread status.</td>
</tr>
</tbody>
</table>

**File Navigation Shortcuts**

<table>
<thead>
<tr>
<th>Key</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>j</td>
<td>Go to next file.</td>
</tr>
<tr>
<td>k</td>
<td>Go to previous file.</td>
</tr>
<tr>
<td>shift + k</td>
<td>Go to first file.</td>
</tr>
<tr>
<td>shift + j</td>
<td>Go to last file.</td>
</tr>
<tr>
<td>u</td>
<td>Go to next unreviewed file.</td>
</tr>
<tr>
<td>i</td>
<td>Go to previous unreviewed file.</td>
</tr>
<tr>
<td>y</td>
<td>Set file reviewed and go to next unreviewed file.</td>
</tr>
</tbody>
</table>
### Using RSS Feeds in Crucible

**Subscribing to an RSS Feed**

In Crucible, all pages with an activity stream and any page which has a list of reviews will have an RSS option.

To access the RSS feed for a page, open the 'Tools' drop-down menu at the top right corner of the screen, then click the 'RSS' option.

This will open a page with the RSS feed displayed; you can also paste the URL from that page into your RSS reader of choice.

### Using Wiki Markup in Crucible

Crucible supports Wiki Markup text formatting in comments and review descriptions.

The text markup notation on this page is a reference showing the available formatting commands.

ℹ️ When using FishEye, you can also render [Wiki Markup in commit messages](#).

#### Headings

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>h1.Biggest heading</td>
<td>Turns text into a heading at size 1. <strong>Biggest Text</strong></td>
</tr>
<tr>
<td>h2.Bigger heading</td>
<td>Turns text into a heading at size 2. <strong>Bigger heading</strong></td>
</tr>
<tr>
<td>h3.Big heading</td>
<td>Turns text into a heading at size 3. <strong>Big heading</strong></td>
</tr>
<tr>
<td>h4.Normal heading</td>
<td>Turns text into a heading at size 4. <strong>Normal heading</strong></td>
</tr>
</tbody>
</table>
h5. Small heading

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>bold</em></td>
<td>Makes text appear <strong>bold</strong>.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Makes text appear in <em>italics</em>.</td>
</tr>
<tr>
<td>+underline+</td>
<td>Makes text appear <em>underlined</em>.</td>
</tr>
<tr>
<td>??citation??</td>
<td>Makes text appear in <strong>citation</strong> form.</td>
</tr>
<tr>
<td>-strikethrough-</td>
<td>Makes text appear <em>strikethrough</em>.</td>
</tr>
<tr>
<td>^superscript^</td>
<td>Makes text appear in <em>superscript</em>.</td>
</tr>
<tr>
<td><del>subscript</del></td>
<td>Makes text appear in <em>subscript</em>.</td>
</tr>
<tr>
<td>{{monospaced}}</td>
<td>Placing double curly-brackets around text makes it appear <em>monospaced</em>.</td>
</tr>
</tbody>
</table>

bq. Block Quote

To make an entire paragraph into a block quotation, place "bq. " before it.

Example:

```
| Some block quoted text |
```

{quote}

here is quoteable content to be quoted

{quote}

Quote a block of text that's longer than one paragraph.

Example:

```
| here is quoteable content to be quoted |
```

{color:red}

look ma, red text!

{color}

Changes the color of a block of text.

Example: look ma, red text!
Text Breaks

Wiki Markup allows you to insert breaks or different kinds of hyphens and dashes.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(empty line)</td>
<td>Produces a new paragraph</td>
</tr>
<tr>
<td><code>\</code></td>
<td>Creates a line break.</td>
</tr>
<tr>
<td><code>----</code></td>
<td>Creates a horizontal ruler.</td>
</tr>
<tr>
<td><code>---</code></td>
<td>Produces em dash — symbol.</td>
</tr>
<tr>
<td><code>--</code></td>
<td>Produces en dash – symbol.</td>
</tr>
</tbody>
</table>

Links

Creating links is easy with Wiki Markup.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Crucible Review CR-FE-100</td>
<td>CR-FE-100]</td>
</tr>
<tr>
<td>[Atlassian Crucible</td>
<td><a href="http://atlassian.com">http://atlassian.com</a>]</td>
</tr>
</tbody>
</table>

Examples:
- [http://www.atlassian.com/crucible](http://www.atlassian.com/crucible)
- [Atlassian Crucible](http://atlassian.com/crucible)

Note: The square brackets [], around external links are optional in the case you do not want to use any alternate text for the link (i.e. just display the raw URL).
| [mailto:mail@example.com] | Creates a link to an email address.  
Example: mail@example.com |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>[file:///c:/temp/foo.txt] [file:///z:/file/on/network/share.txt]</td>
<td>Creates a download link to a file on your computer or on a network share that you have mapped to a drive. To access the file, you must right click on the link and choose &quot;Save Target As&quot;.</td>
</tr>
<tr>
<td>(anchor:anchorname)</td>
<td>Creates a bookmark anchor inside the page. You can then create links directly to that anchor. So a link like this: [My Page#here] will link to wherever in &quot;My Page&quot; there is an (anchor:here) macro, and the link [#there] will link to wherever in the current page there is an (anchor:there) macro.</td>
</tr>
</tbody>
</table>

### Lists

Lists allow you to present information as a series of ordered items. Use asterisks * for bulleted lists and hash symbols # for numbered lists.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
</table>
| * A bulleted list  
* Second item  
** indented item 1  
** indented item 2 | Examples:  
• A bulleted list  
• Second item  
  • indented item 1  
  • indented item 2  
1. A numbered list  
2. Second item  
  a. indented item 1  
  b. indented item 2 |
| # A numbered list  
# Second item  
## indented item 1  
## indented item 2 | |

### Images

Images can be referenced from remote sources only.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![<a href="http://www.host.com/image.gif">http://www.host.com/image.gif</a>]</td>
<td>The image will be displayed from the remote source.</td>
</tr>
<tr>
<td>![<a href="http://www.host.com/image.gif">http://www.host.com/image.gif</a></td>
<td>align=right, vspace=4]</td>
</tr>
</tbody>
</table>

### Tables

Tables allow you to organise content in a rows and columns, with a header row if required.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
</table>
The code above produces a table that looks like this:

<table>
<thead>
<tr>
<th>heading 1</th>
<th>heading 2</th>
<th>heading 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>col A1</td>
<td>col A2</td>
<td>col A3</td>
</tr>
<tr>
<td>col B1</td>
<td>col B2</td>
<td>col B3</td>
</tr>
</tbody>
</table>

### Advanced Formatting

This section covers panels, code windows and showing plain text with no formatting.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{noformat}</td>
<td>Makes a preformatted block of text with no syntax highlighting. All the optional parameters of the {noformat} macro are valid for the {panel} macro as well. Example:</td>
</tr>
<tr>
<td></td>
<td>This is a no-formatted piece of text, so <em>no</em> <em>formatting</em> is done here.</td>
</tr>
<tr>
<td>{panel}</td>
<td>Embraces a block of text within a fully customizable panel. The optional parameters you can define are as follows.</td>
</tr>
<tr>
<td></td>
<td>- title: Title of the panel</td>
</tr>
<tr>
<td></td>
<td>- borderStyle: The style of the border this panel uses (solid, dashed and other valid CSS border styles)</td>
</tr>
<tr>
<td></td>
<td>- borderColor: The color of the border this panel uses</td>
</tr>
<tr>
<td></td>
<td>- borderWidth: The width of the border this panel uses</td>
</tr>
<tr>
<td></td>
<td>- bgColor: The background color of this panel</td>
</tr>
<tr>
<td></td>
<td>- titleBGColor: The background color of the title section of this panel</td>
</tr>
<tr>
<td></td>
<td>Examples:</td>
</tr>
<tr>
<td></td>
<td>Some text in a basic panel</td>
</tr>
<tr>
<td></td>
<td>My Title</td>
</tr>
<tr>
<td></td>
<td>Some text with a title</td>
</tr>
</tbody>
</table>
The code macro displays a preformatted block for showing code with syntax highlighting. All the optional parameters of the \{panel\} macro are valid for \{code\}. The default language is Java but you can specify JavaScript, ActionScript, XML or SQL.

Examples:

**Java with a title bar:**

```java
// Some comments here
public String getFoo()
{
    return foo;
}
```

**A basic display with XML code:**

```xml
<test>
    <another tag="attribute"/>
</test>
```

### Miscellaneous Markup Features

Emoticons and often-used images can be easily embedded with the following Wiki Markup Syntax:

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\X</td>
<td>Escape special character X (i.e. {)</td>
</tr>
<tr>
<td>:), :{</td>
<td>Graphical emoticons (smileys): 😊, 😞</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notation</th>
<th>Image</th>
</tr>
</thead>
<tbody>
<tr>
<td>:)</td>
<td>😊</td>
</tr>
<tr>
<td>:(</td>
<td>😞</td>
</tr>
<tr>
<td>:P</td>
<td>😝</td>
</tr>
<tr>
<td>:D</td>
<td>😄</td>
</tr>
<tr>
<td>;)</td>
<td>😊</td>
</tr>
<tr>
<td>(y)</td>
<td>😃</td>
</tr>
<tr>
<td>(n)</td>
<td>😞</td>
</tr>
<tr>
<td>(i)</td>
<td>😞</td>
</tr>
<tr>
<td>(/)</td>
<td>😞</td>
</tr>
<tr>
<td>(x)</td>
<td>😞</td>
</tr>
<tr>
<td>(l)</td>
<td>😞</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notation</th>
<th>Image</th>
</tr>
</thead>
<tbody>
<tr>
<td>(+)</td>
<td>🟢</td>
</tr>
<tr>
<td>(-)</td>
<td>🟥</td>
</tr>
<tr>
<td>(?)</td>
<td>🎧</td>
</tr>
<tr>
<td>(on)</td>
<td>🟢</td>
</tr>
<tr>
<td>(off)</td>
<td>🟥</td>
</tr>
<tr>
<td>(*)</td>
<td>🟢</td>
</tr>
<tr>
<td>(*)r</td>
<td>🟢</td>
</tr>
<tr>
<td>(*g)</td>
<td>🟢</td>
</tr>
<tr>
<td>(*b)</td>
<td>🟢</td>
</tr>
<tr>
<td>(*)y</td>
<td>🟢</td>
</tr>
</tbody>
</table>

### Using Gadgets in Crucible

This page explains how to use the bundled gadgets in Crucible.
Overview of Crucible Gadgets

As of the release of Crucible 2.3, you can show Crucible data in other Atlassian applications such as JIRA and Confluence by way of special gadgets.

Crucible has three gadgets bundled with it by default:

<table>
<thead>
<tr>
<th>Gadget Name</th>
<th>Description and Gadget URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>'To Do Gadget'</td>
<td>This gadget is a list of Crucible to-do items including reviews to do, comments to read or reviews to summarise. The URL for this gadget is as follows:</td>
</tr>
<tr>
<td></td>
<td><a href="http://HOSTNAME:8060/rest/gadgets/1.0/g/com.atlassian.fecru.fecru-gadgets-plugin:overdueReviews/gadgets/todo.xml">http://HOSTNAME:8060/rest/gadgets/1.0/g/com.atlassian.fecru.fecru-gadgets-plugin:overdueReviews/gadgets/todo.xml</a></td>
</tr>
<tr>
<td></td>
<td>In this example, HOSTNAME:8060 is the hostname of your Crucible instance.</td>
</tr>
<tr>
<td>'Hassle Gadget'</td>
<td>This gadget shows you who you are still waiting on, in other words which reviewers haven't completed your reviews. The URL for this gadget is as follows:</td>
</tr>
<tr>
<td></td>
<td><a href="http://HOSTNAME:8060/rest/gadgets/1.0/g/com.atlassian.fecru.fecru-gadgets-plugin:overdueReviews/gadgets/hassle.xml">http://HOSTNAME:8060/rest/gadgets/1.0/g/com.atlassian.fecru.fecru-gadgets-plugin:overdueReviews/gadgets/hassle.xml</a></td>
</tr>
<tr>
<td></td>
<td>In this example, HOSTNAME:8060 is the hostname of your Crucible instance.</td>
</tr>
</tbody>
</table>
Gadget Configuration 1: Add JIRA to FishEye as an OAuth consumer

Firstly, you need to add JIRA to FishEye as an OAuth consumer. To do this, open the Admin Screen, then click 'Open Authentication (OAuth)' under 'Global Settings' in the left navigation bar. The OAuth configuration screen opens. Click 'Add OAuth Consumer'.

Screenshot: The OAuth Administration Screen
Now, copy the URL for your JIRA instance into the field labelled 'Consumer Base URL' then click 'Add'. The application in use (JIRA or Confluence) will be auto-detected.

**Screenshot: Adding Consumers to OAuth**

Once the instance is added correctly, it will appear in the list of consumers. From here, you’re ready to move onto step two.

**Screenshot: List of OAuth Consumers**

**Gadget Configuration 2: Add Gadget to your Application's Gadget Directory**

As a JIRA administrator you allow the use of these gadgets by adding them to the Gadget Directory. For each gadget, you will need to complete and enter the URL listed in the table above.

See the [JIRA documentation](#) for details on this process.

**Gadget Configuration 3: Add Gadget to the Application's Dashboard**

Finally, as a JIRA user, you need to add the gadget to your dashboard:

See the [JIRA documentation](#) for details on this process.

Once complete, the gadget will appear on your JIRA dashboard and display information drawn from Crucible and FishEye.

Confluence also allows gadgets to be added to its dashboard. See the [General Gadgets Documentation](#) for more information.

**Screenshot: The 'To Do' Gadget**
### Crucible: To Do

<table>
<thead>
<tr>
<th>state</th>
<th>ID</th>
<th>Name</th>
<th>Due</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summarze</td>
<td>CR-3</td>
<td>FE-3456: Make stuff nicer</td>
<td>35 minutes ago</td>
</tr>
<tr>
<td>Respond</td>
<td>CR-1</td>
<td>FE-1234: fix Widget stamping</td>
<td>(1 unread comment) in 6 days</td>
</tr>
<tr>
<td>Review</td>
<td>CR-2</td>
<td>FE-2345: Repair Things</td>
<td>in 6 days</td>
</tr>
</tbody>
</table>

### Screenshot: The Hassle Gadget

<table>
<thead>
<tr>
<th>reviewer</th>
<th>ID</th>
<th>Name</th>
<th>Due</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CR-1</td>
<td>FE-1234: fix Widget stamping</td>
<td>(1 unread comment) in 6 days</td>
</tr>
<tr>
<td></td>
<td>CR-4</td>
<td>FE-4567: Disentangle states</td>
<td>in 10 hours</td>
</tr>
</tbody>
</table>

### Screenshot: The Overdue Reviews Gadget

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Due</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR-3</td>
<td>FE-3456: Make stuff nicer</td>
<td>49 minutes ago</td>
</tr>
<tr>
<td>CR-4</td>
<td>FE-4557: Disentangle states</td>
<td>in 10 hours</td>
</tr>
</tbody>
</table>

### Screenshot: The Changeset Review Coverage Gadget
Using Review Reminders

Crucible will automatically send reviewers a reminder email one working day before the deadline.

Review authors and moderators can also do the following:

- Send manual reminders to reviewers whose work is still pending.
- Configure preset reminders for reviews that have a deadline.

Reminders are only sent if Crucible's SMTP server is configured. Please see Configuring the SMTP server.

Preset Reminders

When a review has a deadline (due date), Crucible (by default) will send a preset reminder to all of the pending reviewers, one working day before the deadline.

To edit the timeframe for the reminder, firstly edit the review, then click Remove next to Send Reminder. Now, set the day for the reminder email to be sent (a number of working days before the deadline).

The Send Reminder setting is only available if the review has a due date set.

Manual Reminders

Crucible can send manually-initiated reminders to all the reviewers that have not yet completed their reviews.

To do this, click Tools > Notify Pending Reviewers. The reminder message will be sent immediately.

This capability is only open to those participants who are authorised to summarize the review.

Transitioning JIRA issues

For Crucible reviews that are linked to JIRA issues, you can now advance their JIRA workflow from within
In the **Close Review** dialog, Crucible will now display any available workflow transitions for the linked issue.

### Transitioning a linked issue

To transition a linked issue:

- In the Review Summary screen, choose **Access: Review > (Summarize or Close) > Close Review**. The possible transitions appear as buttons below the transition.

![Screenshot above: Transitioning a JIRA issue in Crucible.](image)

**CR-FE-6163 Closed**

**Linked issue**

- **Issue**: [FECRU-1401 – Add a review action in the UI to send manual reminder...](image)
- **Status**: [To be reviewed](image)
- **Transition Issue**: Close
- **Log work**: Submit 43m

**You're done!**

You have no outstanding reviews to attend to right now.

**Notes**

- Administrators can activate or deactivate JIRA Issue Transitions at a global level via the Administration interface.
- Only the transitions accessible by the user are displayed.
- The list of available transitions only appears if the user has visibility to any available workflow transitions.

ℹ️ This feature does not support editing fields on the issue, only setting the `resolution` field if required by the transition.

### Related Links

- [Using JIRA Integration in Crucible Reviews](#)
- [JIRA Integration in Crucible](#)

### Glossary

Code review terminology can be confusing as there are many different words for the concepts, roles and process. Crucible has adopted the following terms (click for definitions):
approve

authors in Crucible

code review

comment

creator

defect

moderator

participant

permission scheme

permissions in Crucible

projects in Crucible

review duration

reviewer

role

state

statement of objective

users in Crucible

approve

Issuing a review to the reviewers is known as approving the review.

authors in Crucible

The author is the person primarily responsible for acting on the outcomes of the review. In the vast majority of cases the author will be the person who made the code change under review. Note: to map your repository username to your FishEye/Crucible username, see Changing your User Profile.

code review

Without prejudice to 'code inspection', 'peer review' or a myriad of other terms, Crucible uses the phrase code review for simplicity.

See Getting Started.

comment

A comment is a short textual note that is linked to a review, revision/diff, source line, or to another comment.
Adding Comments

creator

The creator is the person who creates the review. In most cases this person will also act as moderator.

defect

A defect is a comment flagged as something that requires addressing and includes optional defect classifications.

See Flagging Defects and Customising the defect classifications.

moderator

The moderator is the person responsible for creating the review, approving the review, determining when reviewing is finished, summarising the outcomes and closing the review. By default, the moderator is the creator. See also author, the person whose changes to the code are to be reviewed.

participant

Crucible uses the terms creator, author, moderator, and reviewer to describe the roles of review participants.

permission scheme

A permission scheme assigns particular permissions to any or all of the following:

- Particular Users
- Particular Groups
- All logged-in users
- Anonymous Users
- People in particular Review Roles, such as:
  - Author
  - reviewer
  - creator
  - moderator

The scheme's permissions will apply to all reviews belonging to the project(s) with which the scheme is associated.

You can create as many permission schemes as you wish. Each permission scheme can be associated with many projects or just one project, allowing you to tailor appropriate permissions for individual projects as required.

See Creating a permission scheme.

permissions in Crucible

A permission is the ability to perform a particular action in Crucible, e.g. 'Create Review'. Permissions are assigned to particular users, groups or review roles by means of permission schemes.

The following permissions are available:
<table>
<thead>
<tr>
<th>Permission</th>
<th>Description</th>
<th>Default Assignees</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit</td>
<td>Ability to edit a review's details and change the set of revisions being reviewed.</td>
<td>Creator, Moderator</td>
</tr>
<tr>
<td>View</td>
<td>Ability to view a review. (People without this permission will not know that the review exists.)</td>
<td>Anonymous users, All logged-in users, Creator, Author, Reviewer, Moderator</td>
</tr>
<tr>
<td>Abandon</td>
<td>Ability to abandon (i.e. cancel) a review.</td>
<td>Moderator, Creator</td>
</tr>
<tr>
<td>Re-Open</td>
<td>Ability to re-open a closed or abandoned review.</td>
<td>Creator, Moderator</td>
</tr>
<tr>
<td>Uncomplete</td>
<td>Ability of a reviewer to change their individual review status from Complete to Uncomplete.</td>
<td>Reviewer</td>
</tr>
<tr>
<td>Reject</td>
<td>Ability to reject a review submitted for approval (i.e. prevent it from being issued to reviewers).</td>
<td>Moderator</td>
</tr>
<tr>
<td>Complete</td>
<td>Ability of a reviewer to change their individual review status to Complete.</td>
<td>Reviewer</td>
</tr>
<tr>
<td>Comment</td>
<td>Ability to add or remove a comment to or from a review.</td>
<td>Creator, Author, Reviewer, Moderator</td>
</tr>
<tr>
<td>Approve</td>
<td>Ability to approve a review (i.e. issue it to the reviewers).</td>
<td>Moderator</td>
</tr>
<tr>
<td>Submit</td>
<td>Ability to submit a review for approval (i.e. request that the review be issued to the reviewers).</td>
<td>Creator, Author</td>
</tr>
<tr>
<td>Close</td>
<td>Ability to close a review once it has been summarised.</td>
<td>Moderator</td>
</tr>
<tr>
<td>Delete</td>
<td>Ability to delete a review.</td>
<td>Creator, Moderator</td>
</tr>
<tr>
<td>Ability to summarise a review. (Normally this would be done after all reviewers have completed their review.)</td>
<td>Moderator</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Ability to create a review.</td>
<td>All logged-in users</td>
<td></td>
</tr>
<tr>
<td>Ability to resurrect an abandoned (i.e. cancelled) review.</td>
<td>Creator, Moderator</td>
<td></td>
</tr>
</tbody>
</table>

**Projects in Crucible**

A Crucible project is a collection of reviews, typically reviews that all relate to the same application. In addition to providing a logical way of grouping reviews together, a project allows you to:

- define default moderators, authors and reviewers for the reviews in that project.
- define which people are eligible to be reviewers for the reviews in that project.
- use permission schemes to restrict who can perform particular actions (e.g. 'Create Review') in that project.

Every Crucible review belongs to a project. Each project has a name (e.g. ACME Development) and a key (e.g. ACME). The project key becomes the first part of that project's review keys, e.g. ACME-101, ACME-102, etc:

By default, Crucible contains one project. This default project has the key 'CR' and the name 'Default Project'.

See [Creating a Project](#).

**Review Duration**

The review duration is the period of time for which a review will run.

See [Setting the Default Review Duration for a Project](#).

**Reviewer**

A reviewer is a person assigned to review the change. Reviewers can make comments and indicate when they have completed their review. The moderator and author are implicitly considered to be participants of the review, but are not reviewers.

See [Participant](#).

**State**

A Crucible review moves through the following states in the following sequence:
### Draft
See [Creating a Review](#).

### Require Approval
Relevant only when the [moderator](#) is not the [creator](#). See [Issuing a Review](#).

### Under Review
See [Issuing a Review](#) and [Reviewing the Code](#).

### Summarize
See [Summarising and Closing the Review](#).

### Closed
See [Summarising and Closing the Review](#).

⚠ Reviews can be re-opened, i.e. moved from [Summarize](#) or [Closed](#) back to [Under Review](#).

A review may also be in the following states:

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abandoned</td>
<td>This happens when a review is deleted.</td>
</tr>
<tr>
<td>Rejected</td>
<td>Any reviews that a <a href="#">moderator</a> has rejected.</td>
</tr>
</tbody>
</table>

**statement of objective**

A *statement of objective* is an optional text description of the review and any specific areas the [reviewers](#) should focus on.

**users in Crucible**

A *user* is a person using Crucible.